Study and Design of Deep Learning Computer-Aided Diagnosis Systems Based on Biomedical Images and Signals

Department of Electrical and Information Engineering
ELECTRICAL AND INFORMATION ENGINEERING
Ph.D. Program

SSD: ING-INF/06 - ELECTRONIC AND INFORMATION BIOENGINEERING

Final Dissertation

Study and Design of Deep Learning Computer-Aided Diagnosis Systems Based on Biomedical Images and Signals

---

by

Giacomo Donato Cascarano

Supervisor:
Prof. Vitoantonio Bevilacqua, Ph.D.

Coordinator of Ph.D. Program:
Prof. Luigi Alfredo Grieco, Ph.D.

Course n°33, 01/11/2017 - 31/12/2020

Department of Electrical and Information Engineering
ELECTRICAL AND INFORMATION ENGINEERING
Ph.D. Program

SSD: ING-INF/06 - ELECTRONIC AND INFORMATION BIOENGINEERING

Final Dissertation

Study and Design of Deep Learning Computer-Aided Diagnosis Systems Based on Biomedical Images and Signals

---

by

Giacomo Donato Cascarano

Referees:
Prof. Leandro Pecchia, Ph.D.
Prof. Salvatore Vitabile, Ph.D.

Supervisor:
Prof. Vitoantonio Bevilacqua, Ph.D.

Coordinator of Ph.D. Program:
Prof. Luigi Alfredo Grieco, Ph.D.

Course n°33, 01/11/2017 - 31/12/2020

Abstract

This Ph.D. thesis aims to describe the research works conducted for the design, the development and the evaluation of innovative Computer-Aided Diagnosis (CAD) systems based on machine learning and deep learning techniques. Several CAD solutions were developed in different medical applications trying to ensure, when possible, three main CAD requirements: improve clinicians performance, reduce or at least not increase clinicians time and integrate the CAD solution in standard procedures. The proposed applications involved images and signals processing; the firsts required the use of different deep learning models to face classification, detection and segmentation problems, while the latter allowed to investigate machine learning as signal processing technique for movement disorder analysis and for a more speculative research in the rehabilitation field. In order to properly validate the proposed algorithms, all the methodologies were applied on real data provided by clinicians, public datasets or specific acquisitions.

Potentialities, challenges and drawbacks about deep learning for medical imaging analysis are discussed in two medical fields, digital pathology and radiology, and complete pipelines are proposed to accomplish three clinical practices: global glomerulosclerosis analysis for Chronic Kidney Disease evaluation, kidneys volume analysis for Autosomal Dominant Polycystic Kidney Disease evaluation and organs segmentation for generic volume quantification. Each study case aims to identify and overcome the limitation of classical image processing techniques, and paves the way towards the clinical use of CAD systems based on deep learning. A second part of this thesis focuses on machine learning and deep learning for signals processing; deep neural networks were investigated for movement disorders analysis and a particular neural model for surface electromyography analysis has been proposed for the evaluation of complex muscle activation patterns, useful in the rehabilitation field. The developed solutions for signals and images processing, were compared with literature standards and, if possible, a personalised classical pipelines has been proposed and customised to face each clinical challenge.

The thesis is divided into six chapters. The first chapter provides an introduction about the reference context. The following chapter two describes the state of the art about traditional

CAD systems based on conventional machine learning algorithms, and the novelty that deep learning techniques bring to CADs and medical practices; description of the main convolutional neural network models and autoencoders, and literature about the application of deep learning and machine learning to the concerned medical fields are reported. Chapters three, four and five report the original contribution about the application of deep learning and machine learning techniques to the two types of medical data: images and signals; in detail, chapter three reports the applications in the clinical areas of digital pathology and radiology, focusing on the development of full pipelines based on image analysis; chapter four shows a more speculative research work for signal processing, focusing on the application of undercomplete autoencoders for surface electromyography analysis; chapter five reports the applications of deep neural networks for diseases assessment and grading in subjects affected by movement disorders. The analysed study cases and the contributions reported in this thesis were compared with standard processing techniques ad-hoc developed. Finally, the conclusions about the research works and proposals for future researches are reported in chapter six.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Computer-aided diagnosis (CAD) has been an important research field in the last decades. In medicine, CAD systems combine processing techniques proper of the application domain and machine learning methods to analyse images, signals and patient data. The goal of CADs is the comprehensive evaluation of input information to ease and assist clinicians in their decision-making process, and its outcome is strictly linked to the application domain. In general, CAD are designed to model the ratio behind traditional clinician practice, and are used to predict or to follow-up patients conditions. A well designed and validated CAD is a valuable support system for the diagnostic process, providing an objective double reading, thus avoiding medical malpractices. The objectivity of CAD systems is their breakthrough advantage; the abilities to quantitatively characterise disease information coming from multiple biological scales and dimensions have the potential to enable the development of preventive strategies and medical treatments precisely targeted to each class of patients.

Machine learning is a wide field of computer science accounting applications for many research areas based on images and signals analysis. Machine learning are mathematical and statistical algorithms applicable in multidisciplinary fields, and are able to process heterogeneous data to extract relevant information; the correct interpretation of these information lead to the desired outcome prediction for a given task. Formerly, machine learning techniques were indirectly applied on input data such as images and mono- or multi-dimensional signals, because heterogeneous and non standardised data make their application more difficult to validate. Then, input data were commonly pre-processed to extract meaningful domain-dependent features that were used as inputs of machine learning algorithms. The advent of deep learning among machine learning methods reversed this problem thanks to its ability to automatically extract those features, limiting the necessity of hand-engineered features. The advantages are magnified on inputs with grid-like topology, such as images or time

series, but it introduces some drawbacks. Deep learning models are commonly more complex than traditional counterparts requiring a more tedious learning process and a well designed input knowledge-base; furthermore, from a clinician point of view, automatic extracted features lose meaning and leads these models to act as black-boxes. The meaning-less is only apparent, but the complexity of the models makes their understanding really difficult. However, recent research about radiomic using deep learning are going in this direction, linking specific features to clinical outcomes.

The emerging of deep learning as the state-of-the-art machine learning method, its ability to face any kind of decisional problem and the improvements in the hardware technology attracted new researchers, allowing the development of more performing CAD systems able to tackle new challenging and complex clinical tasks.

1.1 Objective and Research Question

The success of deep learning in many machine learning tasks brings back interests in research and development of CAD in various scenarios. Important examples are the recent competitions[1] of developing CAD work-flow for different classification tasks, where all winning teams used deep learning approaches [1]. Although the impressive performance improvements brought by the adoption of this new paradigm in the CAD field, high optimism and expectations should be viewed with caution. Many benchmarks shown the improvement and the robustness achievable in CAD applications with the adoption of deep learning instead of conventional machine learning, but these new CAD pipelines have not been sufficiently tested in clinical practise, where real scenarios pose the problem of high data variability and heterogeneity. Furthermore, some proposed solutions claim to achieve perfect performance or even better than expert clinicians, thus showing that solution validated on restricted knowledge-base are worth to be investigated but they are not enough for daily clinical applications [1].

Taking into account the precedent considerations and the compelling opportunity of deep learning, the objective of this research work has been to investigate and develop innovative CAD pipelines for further improve the state of the art in such systems. Several CAD solutions were developed in different medical applications trying to ensure, when possible, three main CAD requirements: improve clinicians performance, reduce or at least not increase clinicians time and integrate the CAD solution in standard procedures. The proposed applications involved images and signals processing; the firsts required the use of image processing and

[1] Examples: https://www.kaggle.com/competitions and http://dreamchallenges.org/

different deep learning models to face classification, detection and segmentation problems, while the latter allowed to investigate machine learning as signal processing technique for diseases assessment and grading and for a more speculative research in the rehabilitation field. In order to properly validate the proposed algorithms, all the methodologies were applied on real data provided by clinicians, public datasets or specific acquisitions (for the study cases involving signals).

1.2 Contribution

The aim of this thesis is to present a set of full work-flows for the development of innovative CAD systems based on deep learning and machine learning techniques. Potentialities, challenges and drawbacks about deep learning for medical imaging analysis are discussed in two medical fields, digital pathology and radiology, and complete pipelines are proposed to accomplish three clinical practices: global glomerulosclerosis analysis for Chronic Kidney Disease evaluation, kidneys volume analysis for Autosomal Dominant Polycystic Kidney Disease evaluation and organs segmentation for generic volume quantification. Each study case aims to identify and overcome the limitation of classical image processing techniques, and paves the way towards the clinical use of CAD systems based on deep learning. A second part of this thesis focuses on machine learning and deep learning for signals processing; deep neural networks were investigated for movement disorders analysis and a particular neural model for surface electromyography analysis has been proposed for the evaluation of complex muscle activation patterns, useful in the rehabilitation field. The developed solutions for signals and images processing, were compared with literature standards and, if possible, a personalised classical pipelines has been proposed and customised to face each clinical challenge.

1.3 Part Outline

The thesis is divided into six chapters. The first and current Chapter 1 provides an introduction about the reference context. The following Chapter 2 describes the state of the art about traditional CAD systems based on conventional machine learning algorithms, and the novelty that deep learning techniques bring to CADs and medical practices; description of the main convolutional neural network models and autoencoders, and literature about the application of deep learning and machine learning to the concerned medical fields are reported. Chapters 3, 4 and 5 report the original contribution about the application of deep learning and machine

learning techniques to two types of medical data: images and signals; in detail, Chapter 3 reports the applications in the clinical areas of digital pathology and radiology, focusing on the development of full pipelines based on image analysis; Chapter 4 shows a more speculative research work for signal processing, focusing on the application of undercomplete autoencoders for surface electromyography analysis; Chapter 5 reports the applications of deep neural networks for diseases assessment and grading in subjects affected by movement disorders. The analysed study cases and the contributions reported in this thesis were compared with standard processing techniques ad-hoc developed. Finally, the conclusions about the research works and future research proposes are reported in Chapter 6.

Chapter 2

State of the art

2.1 Traditional Approaches for Computer-Aided Diagnosis

Computer-Aided Diagnosis (CAD) systems are powerful tools supporting physicians in management and interpretation of biomedical signals, helping them in clinical decisions. Mono or multi-dimensional biomedical signals hide valuable information that have to be analysed and correlated, with data coming from other domains, to give a comprehensive evaluation, possibly, in a short time. CAD systems are able to process digital signals in order to discover interesting hidden information, and to following suggest diseases or clinical outcomes; the output of a CAD system will be the input for medical professionals to support their decision.

Over time, CAD systems have been addressed with several different terms, mainly based on the CAD specific purpose; examples are expert systems (ES), computer-aided evaluation or diagnosis, computerised sound analysis (e.g., Computerised Lung Sound Analysis (CLSA) and Computerised Heart Sound Analysis (CHSA)), computerised biomedical signal analysis [2–6]. Also, there are many ways in which CADs may operate. Simpler automatic systems usually emulate the decision work-flow based on diagnostic rules used by a human expert for making diagnoses. More evolute and sophisticate systems are based on intelligent algorithms that can give them learning capabilities, that is the ability to analyse clinical data and infer new knowledge [7]; this new knowledge can enhance initial diagnostic rules and enable the ability to further improve performance over time. However, the new knowledge is ever validated by human domain experts.

The level of sophistication of a CAD systems followed the increase of data amount and complexity produced in the modern clinical practice, and led to the application of approaches previously not used in the clinical fields: Artificial Intelligence (AI), Data Mining algorithms and Machine Learning (ML). Important examples are the technological improvements of the human body examination tools, including X-rays devices, ultrasounds, Magnetic Resonance Imaging (MRI) and Computed Tomography (CT), that require customised application to process the generated data. All these improvements allowed the development of new CAD systems that widen the support in making and supporting diagnostics decisions to diseases and clinical conditions [8–10].

2.1.1 A Brief History of Decision Support Systems in Medicine

Researchers use of computers to investigate problems started in the late 1950s, and involved several fields including medicine and biology. Literature of that period reports the first attempts to deal with medical diagnosis by using computerised systems [11–13]. The firsts automatic diagnostic systems, called expert systems in medicine, were base on the application of rules and knowledge bases to recreate the association between symptoms and laboratory test outcomes [14, 15]. During the 1960s several radiologists started working on an early form of CAD system for the automatic detection of abnormalities in medical images [16–18]. However, these systems demonstrated to be too simple to handle complex problems such as clinical diagnosis, and could not be used in the daily clinical practice. The main limitation was the poor automatisms in clinical diagnosis of the algorithms [19], but this entices researches to investigate new ones based on artificial intelligence and specific fields, such as pattern recognition and classification algorithms [20]. Today, CAD systems are considered as an important part of a diagnostic process which also actively involves human experts; diagnostic radiology and medical image analysis are some of the most active research and application examples of CAD development [21]. Furthermore, the use of CAD as support systems, implied others advantages, such as the avoid of medical malpractice with benefit on medical healthcare costs, and they constitute cheap and suitable alternatives to double reading as a mean for reducing diagnostic errors [22].

In the last years, CAD systems become very popular in the academic literature (Fig. 2.1). Research activities allowed to develop intelligent systems based on signal processing algorithms, new techniques for extracting features from input data, and machine learning algorithms for developing intelligent classifiers for the different clinical objectives; then

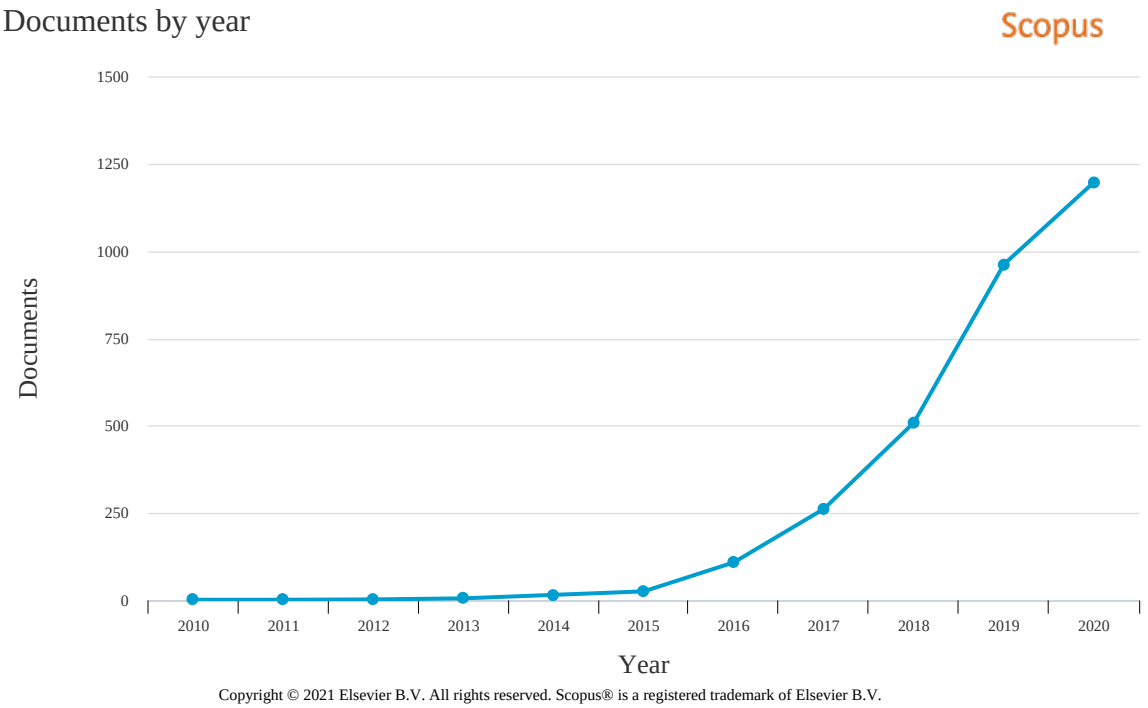


Fig. 2.1 Number of publications per year. Scopus search: *TITLE-ABS-KEY (((deep AND learning) AND ((cad) OR (computer AND aided AND diagnosis))))*; results filtered from 2010 to 2020. Narrowing the research topic on *medical imaging* drops the overall document count from 3094 to 1282 with the same trend.

nowadays, the design of CAD systems require several professionals, from domain experts to system developers.

2.1.2 The Traditional Pipeline of CAD Systems

The development of a complete CAD system requires the design of several modules that operate together to form a full work-flow, considered by the end-user as black-box that takes data as input and provides the desired output; a simplified pipeline is depicted in Fig. 2.2. Based on the specific clinical domain interested by the decision support system, each module specialises its specific working methodology and procedures to accomplish the required tasks. From the perspective of the clinical objective, CAD systems may be divided between those aimed at classifying, i.e. diagnosis detection or suggestion, and system for segmentation purposes, usually useful for quantification or objectification of measurements. Finally, in order to assess the performance of the designed CAD work-flow, it is necessary to define validation procedures aimed at verifying the correct functioning of the automatic system

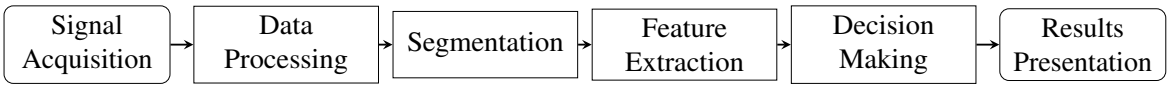


Fig. 2.2 Simplified pipeline with the main steps composing a CAD system.

during normal working conditions [21, 23, 24]. The main pipeline steps will be discussed in the following paragraphs.

Data Processing. For each clinical domain, developing CAD systems requires a campaign for collecting and organising data, based on which decisions can be taken. Since data can be collected from different sources with different modalities of acquisition, or from different clinical facilities with different protocols, they are usually heterogeneous and need to be pre-processed in order to clean and standardise the information content [25]. This is also a crucial step in order to facilitate and optimise the following phases, since its output affects the performance of the whole work-flow. Based on the defects affecting data respect to the desired standard, there are a lot of strategies and algorithms to apply in order to clean and uniform them [26, 27]. Common examples are data re-sampling, noise reduction algorithms, or other filtering procedures. In the case of CAD systems working on images, a pre-processing phase is needed after the acquisition of the input data; this step is fundamental for improving the quality of the images and removing possible artefacts [28]; literature accounts a huge number of useful algorithms for medical imaging pre-processing [29–33].

Segmentation. In the case of images or signals, a further step allowing to segment the input data may be required. In fact, the following feature extraction step may require or may be able to operate only if a precise Regions of Interest (ROI) is provided. Segmentation is a very important step in processing medical imaging, aiming at separating images into regions that are meaningful for a specific task, such as the detection of organs; further details will be given in Section 2.4.2.2.

Feature Extraction. After data pre-processing, the candidate regions of the input signals are used to extract meaningful and representative features. Traditionally, the feature are extracted to represent a particular aspect of the input space that is usually suggested by the domain expert; as example, feature related to the shape can be useful for tumours evaluation (Fig. 2.3), whilst features describing the texture can be useful for the evaluation of kidney components (Section 2.4.1.2).

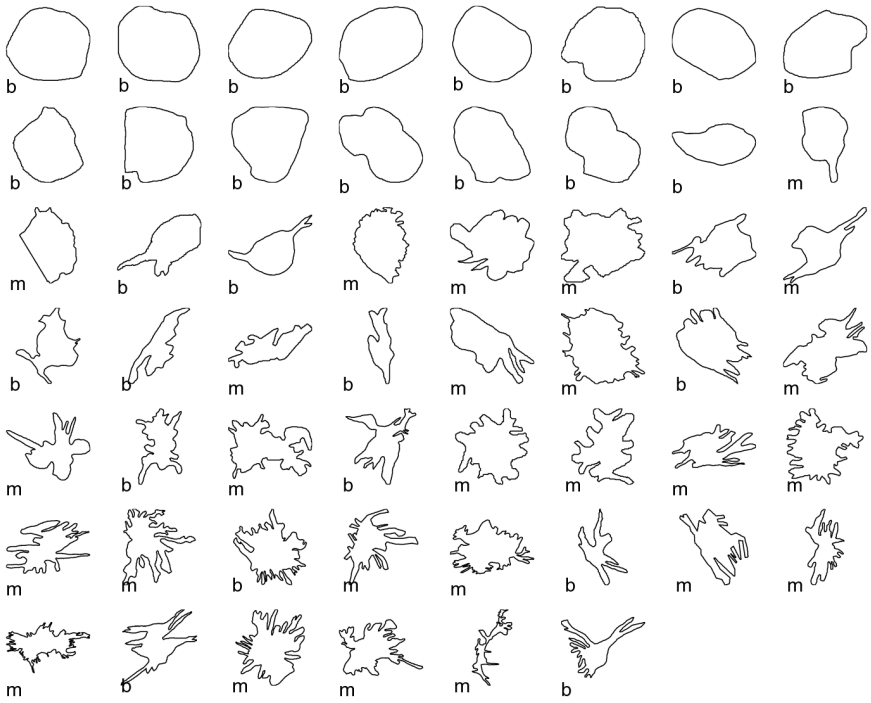


Fig. 2.3 Example of tumour diagnosis from contours of breast masses: benign masses (b), malignant tumours (m). It is possible to observe how the shape and the smoothness of a tumour mass contour can be used to help in its malignity evaluation.

According to literature, starting from '70s there are several sets of features that could be used to characterize regions of interest [34]. But, due to the different image sources, processing procedures and analysis algorithms available for medical imaging, recent research works raised the problem of reproducibility; a standardisation requirement is essential for innovative field, such as radiomics [35], but it becomes useful also for general CAD systems (note that a CAD system could involve radiomics). Furthermore, a standardised features set is a useful comparison benchmark when innovative techniques for feature processing are introduced, i.e. deep learning.

In the tentative to introduce standardisation, many researchers started to group features based on their nature (i.e. intensity or morphological), or on the method used for their calculation (i.e. Gray Level Co-occurrence Matrix (GLCM) [36] or neighbourhood analysis [37]). Before extracting features, as stated before, ROIs should be generated; two kinds of ROIs could be defined: one describing morphological information and the other about intensity information. The two ROIs may be identical, but not necessarily due to pre-processing steps (e.g. segmentation procedure could be performed on the intensity mask to remove unwanted or out of range intensity value). Table 2.1 reports the full list of the features families, the number of features belonging to, and the required ROI mask [38], built upon the feature sets proposed by Aerts *et al.* [39], Hatt *et al.* [40] and derived works.

Morphological features, GLMC and intensity-based statistical features, revealed to be very frequently used in research works, together with higher-order features [45]. They are following detailed:

- *morphological features* - Morphological features describe bi- or three-dimensional geometric aspects of a ROI, such as area or volume. For this purpose, shapes are approximated to circumferences (for bi-dimensional features) or triangle meshes (for three-dimensional features). From the generated meshes, the marching cubes algorithm is generally used to generate the complex ROI mesh of the analysed area or volume where features will be computed [46]. Table 2.2 reports the most employed morphological features; such features allow describing areas of interest from the volumetric or surface point of view, making evident structural characteristics;

- *intensity-based statistical features* - These features describe the distribution of voxel intensities within the ROI. Table 2.3 reports only the most used features of such kind;

- *grey level co-occurrence based features* - The last set of analysed features are based on the Grey-Level Co-occurrence Matrix. Considering a grey level image, with N grey levels, GLCM is an $N \times N$ matrix describing the second-order joint probability function of a ROI

Table 2.1 List of features families and required ROI mask. Table from Brunetti *et al.* [41].

Feature Family	**Feature Count**	**Required Mask**	
		Morphological	**Intensity**
Morphology	*29*	✓	✓
Local Intensity	*2*		✓
Intensity-based Statistics	*18*		✓
Intensity Histogram	*23*		✓
Intensity-Volume Histogram	*5*		✓
Gray Level Co-occurrence Matrix (GLCM) [36]	*25*		✓
Gray Level Run Length Matrix (GLRLM)[42]	*16*		✓
Gray Level Size Zone Matrix (GLSZM) [43]	*16*		✓
Gray Level Distance Zone Matrix (GLDZM) [43]	*16*	✓	✓
Neighbourhood Grey Tone Difference Matrix (NGTDM) [44]	*5*		✓
Neighbourhood Grey Level Dependence Matrix (NGLDM) [37]	*17*		✓

and it is defined as $P(i,j|\delta,\theta)$. The $(i,j)th$ element of this matrix represents the number of times the combination of levels i and j occur in two pixels in the image that are separated by a distance of $\delta pixels$ along angle θ. Let's consider:

- $\mathbf{P}(i,j)$ be the co-occurence matrix for an arbitrary δ and θ;
- $p(i,j)$ be the normalized co-occurence matrix and equal to $\frac{\mathbf{P}(i,j)}{\sum\mathbf{P}(i,j)}$;
- N_g be the number of discrete intensity levels in the image;
- $p_x(i) = \sum_{j=1}^{N_g} P(i,j)$ be the marginal row probabilities;
- $p_y(j) = \sum_{i=1}^{N_g} P(i,j)$ be the marginal column probabilities;
- μ_x be the mean gray level intensity of p_x and defined as $\mu_x = \sum_{i=1}^{N_g} p_x(i)i$;
- μ_y be the mean gray level intensity of p_y and defined as $\mu_y = \sum_{j=1}^{N_g} p_y(j)j$;
- σ_x be the standard deviation of p_x;
- σ_y be the standard deviation of p_y;
- $p_{x+y}(k) = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j),$ where $i+j=k,$ and $k = 2,3,\ldots,2N_g$;
- $p_{x-y}(k) = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j),$ where $|i-j|=k,$ and $k = 0,1,\ldots,N_g-1$;
- $HX = -\sum_{i=1}^{N_g} p_x(i)\log_2\left(p_x(i)+\varepsilon\right)$ be the entropy of p_x;
- $HY = -\sum_{j=1}^{N_g} p_y(j)\log_2\left(p_y(j)+\varepsilon\right)$ be the entropy of p_y;
- $HXY = -\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\log_2\left(p(i,j)+\varepsilon\right)$ be the entropy of $p(i,j)$;
- $HXY1 = -\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\log_2\left(p_x(i)p_y(j)+\varepsilon\right)$;
- $HXY2 = -\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p_x(i)p_y(j)\log_2\left(p_x(i)p_y(j)+\varepsilon\right)$.

Based on these considerations, the features computed by GLCM and reported in Table 2.4 could be defined. More details about GLCM are in Haralick *et al.* paper [36].

Theoretically, to better represent the input images, a huge number of feature should be preferred allowing a comprehensive description of the input space; but in practical application the number of features have to be compared to the size of input data and it is usually suggested to consider fewer features than dataset samples. Furthermore, most of the times the extracted features are needles and correlated each other. In general, the high dimensionality of input space could lead to relevant problems in the subsequent decision phase, seriously affecting performance, increasing time and computational resources. For these reasons, it is possible to further process the dataset with a feature selection or a space transformation that converts the input space to a new one that ensures specific characteristic (e.g. feature orthogonality),

and preserves all or a defined part of the input information content. Several techniques allow to properly process a dataset in order to transform and reduce its dimensionality, leading to a general increase of performance for the whole pipeline; these include feature selection methods, e.g. Information Gain [48], or feature transformation and reduction methods, e.g. Principal Component Analysis or Independent Component Analysis [49–57]. Some of the features described above and PCA were widely used in the study cases reported in Chapters 3 and 4.

Devision Making After the preparation of a good feature dataset, the decision step has the role to find and apply a model (a set of rule is the simpler one) that is able to give the desired output according to the input (e.g. discriminate between benign and malignant tumour or determine its grade). The model can be of any type (supervised or unsupervised) and complexity (from simple rule to artificial neural network models).

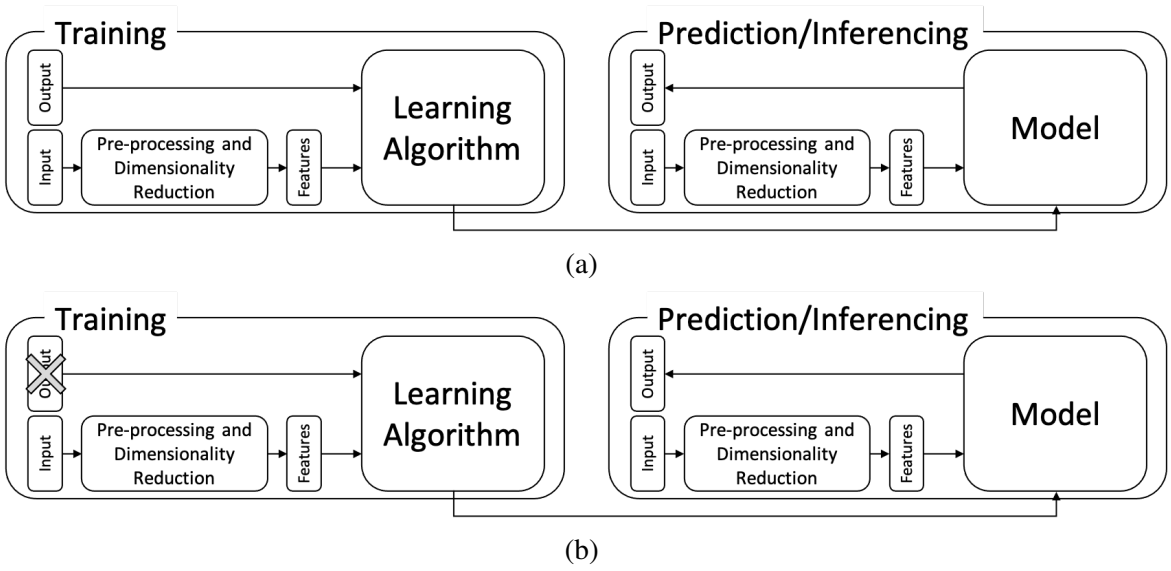


Fig. 2.4 Simplified scheme of supervised (a) and unsupervised (b) approaches.

The goal of a decision model is to learn how to map the inner knowledge of a dataset to satisfy the purpose of the model itself; there are two main typology of learning process: supervised and unsupervised. The first one looks for the relation between input and output, trying to replicate the real "model" that link them; the latter, try to find common aspects among data allowing to aggregate them in base of some rule (e.g. minimum distance) and giving an insight on the dataset structure. For supervised learning, in detail, there is usually a learning algorithm that try to manipulate the model parameters to optimise an error function; in some cases the model training is not deterministic (really common for non

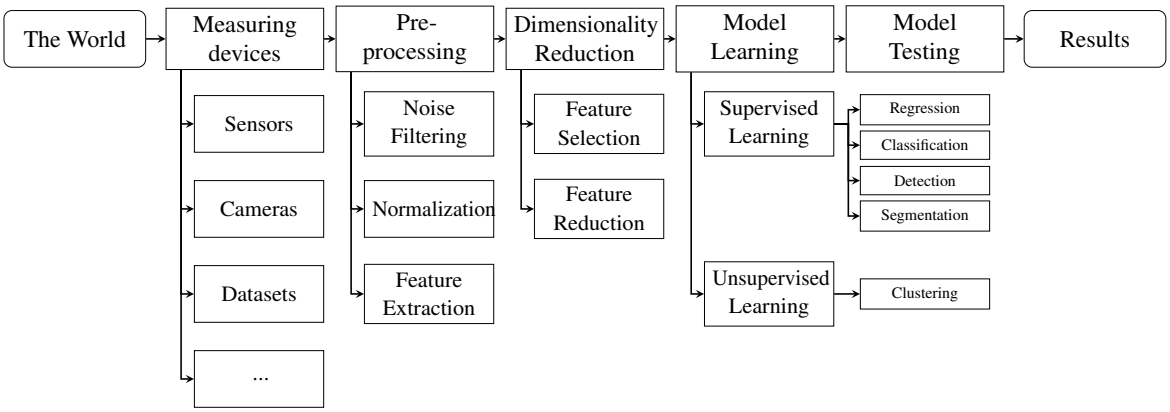


Fig. 2.5 Example of a full CAD pipeline.

convex cost function) and there is a plenty of hyper-parameters of the model or of the whole work-flow that have to be chosen. A good practise is to split the dataset in three subset called training, validation and test set. The first subset is used to optimise the model, the second to take decision about the trained models (i.e. it is possible to chose the best one among several trained with different hyper-parameters), and the last one is usually reserved for final results only. The data used in each subset should be different, and are commonly randomly sampled from the whole dataset There are no mathematical rules for the determination of the correct sizes of training, validation and test sets. However, there are some rules of thumb derived from the designer experience [58]. Another popular strategy for model design and selection is Cross-Validation; the main idea is to sample the input dataset, once or several times, to create more training-validation subset, and train the model considering each subset combination. The popularity of cross-validation mostly comes from the universality of the data splitting heuristics. Nevertheless, some procedures have been proved to fail for some model selection problems, depending on the goal of model selection, estimation or identification. Furthermore, many theoretical questions remain widely open [59–65]. Both, dataset split and cross validation were widely used in the study case presented in Chapters 3 and 4.

A more detailed representation of a full CAD work-flow and reporting all the different variables explained before, is depicted in Fig. 2.5.

Table 2.2 List of the most employed morphological features. N_f and N_v are the number of faces defining the mesh and the number of voxels included in the ROI, respectively. Table from Brunetti *et al.* [41] and pyradiomics documentation [47].

Feature Name	**Equation**
Mesh Volume (V)	$V = \sum_{i=1}^{N_f} V_i,\ where\ V_i = \frac{Oa_i \cdot (Ob_i \times Oc_i)}{6}$
Voxel Volume (V_{voxel})	$V_{voxel} = \sum_{k=1}^{N_v} V_k$
Surface Area (A)	$A = \sum_{i=1}^{N_f} A_i,\ where\ A_i = \frac{1}{2}\lvert \mathrm{a}_i\mathrm{b}_i \times \mathrm{a}_i\mathrm{c}_i \rvert$
Surface Area to Volume ratio	$surface\ to\ volume\ ratio = \frac{A}{V}$
Sphericity 3D	$sphericity = \frac{\sqrt[3]{36\pi V^2}}{A}$
Maximum 3D and 2D diameters	Largest pairwise Euclidean distance between ROI surface mesh vertices, eventually computed in a 2D plane
Major (MA), Minor (mA) and Least (LA) Axis Length	$MA = 4\sqrt{\lambda_{major}}, mA = 4\sqrt{\lambda_{minor}}, LA = 4\sqrt{\lambda_{least}}$
Elongation	$elongation = \sqrt{\frac{\lambda_{minor}}{\lambda_{major}}}$
Flatness	$flatness = \sqrt{\frac{\lambda_{least}}{\lambda_{major}}}$
Mesh Surface (A)	$A = \sum_{i=1}^{N_f} A_i,\ where\ A_i = \frac{1}{2}\mathrm{Oa}_i \times \mathrm{Ob}_i$
Pixel Surface (A_pixel)	$A_{pixel} = \sum_{k=1}^{N_v} A_k$
Perimeter (P)	$P = \sum_{i=1}^{N_f} P_i,\ where\ P_i = \sqrt{(\mathrm{a}_i - \mathrm{b}_i)^2}$
Perimeter to Surface ratio	$perimeter\ to\ surface\ ratio = \frac{P}{A}$
Sphericity 2D	$sphericity = \frac{2\pi R}{P} = \frac{2\sqrt{\pi A}}{P}$

Table 2.3 List of the most employed intensity-based statistical features. $\mathbf{X}$ is a set of N_p voxels; $P(i)$ is the first-order histogram with N non-zero bins; $p(i)$ is the normalised first-order histogram ($\frac{P(i)}{N_p}$). Table from Brunetti *et al.* [41] and pyradiomics documentation [47].

Feature Name	**Equation**
Energy	$energy = \sum_{i=1}^{N_p} (\mathbf{X}(i)+c)^2$
Total Energy	$totalenergy = V_{voxel} \sum_{i=1}^{N_p} (\mathbf{X}(i)+c)^2$
Entropy	$entropy = -\sum_{i=1}^{N_g} p(i) \log_2 \big(p(i)+\varepsilon\big)$
Minimum	$minimum = \min(\mathbf{X})$
Maximum	$maximum = \max(\mathbf{X})$
Mean	$mean = \frac{1}{N_p} \sum_{i=1}^{N_p} \mathbf{X}(i)$
Median	the median grey level intensity in the considered ROI
Mean Absolute Deviation (*MAD*)	$MAD = \frac{1}{N_p} \sum_{i=1}^{N_p} \lvert \mathbf{X}(i) - \bar{X} \rvert$
Robust Mean Absolute Deviation (*rMAD*)	$rMAD = \frac{1}{N_{10-90}} \sum_{i=1}^{N_{10-90}} \lvert \mathbf{X}_{10-90}(i) - \bar{X}_{10-90} \rvert$
Root Mean Squared (*RMS*)	$RMS = \sqrt{\frac{1}{N_p} \sum_{i=1}^{N_p} (\mathbf{X}(i)+c)^2}$
Variance	$variance = \frac{1}{N_p} \sum_{i=1}^{N_p} (\mathbf{X}(i) - \bar{X})^2$
Standard Deviation	$std = \sqrt{\frac{1}{N_p} \sum_{i=1}^{N_p} (\mathbf{X}(i) - \bar{X})^2}$
Skewness	$skewness = \frac{\mu_3}{\sigma^3} = \frac{\frac{1}{N_p} \sum_{i=1}^{N_p} (\mathbf{X}(i)-\bar{X})^3}{\left(\sqrt{\frac{1}{N_p} \sum_{i=1}^{N_p} (\mathbf{X}(i)-\bar{X})^2}\right)^3}$
Kurtosis	$kurtosis = \frac{\mu_4}{\sigma^4} = \frac{\frac{1}{N_p} \sum_{i=1}^{N_p} (\mathbf{X}(i)-\bar{X})^4}{\left(\frac{1}{N_p} \sum_{i=1}^{N_p} (\mathbf{X}(i)-\bar{X})^2\right)^2}$
Uniformity	$uniformity = \sum_{i=1}^{N_g} p(i)^2$

Table 2.4 List of the most employed GLCM-based features. Table from Brunetti *et al.* [41] and pyradiomics documentation [47].

Feature Name	**Equation**
Autocorrelation	$autocorrelation = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)ij$
Cluster Prominence	$cluster\ prominence = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}$
Cluster Shade	$cluster\ shade = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\left(i+j-\mu_x-\mu_y\right)^3 p(i,j)$
Cluster Tendency	$cluster\ tendency = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\left(i+j-\mu_x-\mu_y\right)^2 p(i,j)$
Contrast	$contrast = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(i-j)^2 p(i,j)$
Correlation	$correlation = \frac{\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)ij-\mu_x\mu_y}{\sigma_x(i)\sigma_y(j)}$
Difference Average (*DA*)	$difference\ average = \sum_{k=0}^{N_g-1} k p_{x-y}(k)$
Difference Entropy	$difference\ entropy = \sum_{k=0}^{N_g-1} p_{x-y}(k)\log_2\left(p_{x-y}(k)+\varepsilon\right)$
Difference Variance	$difference\ variance = \sum_{k=0}^{N_g-1}(k-DA)^2 p_{x-y}(k)$
Joint Energy	$joint\ energy = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\left(p(i,j)\right)^2$
Joint Entropy	$joint\ entropy = -\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\log_2\left(p(i,j)+\varepsilon\right)$
Informational Measure of Correlation (*IMC*)	$IMC_1 = \frac{HXY-HXY1}{\max\{HX,HY\}}$, $IMC_2 = \sqrt{1-e^{-2(HXY2-HXY)}}$
Maximal Correlation Coefficient (*MCC*)	$MCC = \sqrt{2nd\ largest\ Eig(Q)}$, *where* $Q(i,j) = \sum_{k=0}^{N_g}\frac{p(i,k)p(j,k)}{p_x(i)p_y(k)}$
Inverse Difference Moment Normalised (*IDMN*)	$IDMN = \sum_{k=0}^{N_g-1}\frac{p_{x-y}(k)}{1+\left(\frac{k^2}{N_g^2}\right)}$
Inverse Variance	$inverse\ variance = \sum_{k=1}^{N_g-1}\frac{p_{x-y}(k)}{k^2}$
Sum Average	$sum\ average = \sum_{k=2}^{2N_g} p_{x+y}(k)k$
Sum Entropy	$sum\ entropy = \sum_{k=2}^{2N_g} p_{x+y}(k)\log_2\left(p_{x+y}(k)+\varepsilon\right)$
Sum of Squares	$sum\ squares = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(i-\mu_x)^2 p(i,j)$

2.1.3 Traditional Machine Learning Algorithms for CAD Systems

Based on the consideration discussed in the previous section, the decision model is the step where the semantic relation between input and output is created, allowing the CAD systems to make decisions about a pathology or compute descriptive scores for a quantitative evaluation. Based on the model designed for making the decision and on the employed training techniques, different kinds of CAD systems can be defined. Among the several learning techniques available in literature, supervised learning is the most interesting for the aim of this thesis.

Decision Support Systems (DSS) based on supervised learning, are automatic architectures producing a model that is able to discover and describe some relation already present in a dataset but not known to the user. To allow the learning process, a good dataset or a knowledge base is needed, and in general the information about the output given a particular input have to be provided; as example, to design a model able to classify a disease, to detect the presence of a tumour or to evaluate its stage (e.g. regression), the preliminary association between the input feature set and the output have to be known.

In the last years, a relevant number of studies in the clinical area proposed approaches based on supervised learning algorithms; the most known and used algorithms are Artificial Neural Networks (ANN) and Support Vector Machines (SVM), as well as Swarm Intelligence or Linear Discriminant Analysis (LDA) [66–82].

Since the importance of Artificial Neural Networks and the link between them and deep learning (they can be considered as the deep learning ancestors), and since Artificial Neural Networks are used in some of the study case reported in Chapter 3, a detailed description is reported above. Furthermore, the concepts that will be reported about the learning process are usable for deep learning model too.

Artificial Neural Networks. Artificial Neural Networks (ANN) are biologically inspired computing systems that mimic the physiological neural networks of the human brains [83]. ANN can be considered as a set of numerical techniques, and were initially used in several fields from solving ordinary and partial differential equations to the modelling and control of non-linear systems [84]. ANNs can be generally defined as non-linear, multi-layered, parallel regression techniques and can be used for signal processing, forecasting and clustering. As for others supervised techniques, the training process of ANN is based on the learning of distributions from examples. For example, in medical image analysis, it may learn to detect cancer from images to support oncologists during the phases of diagnosis [10], while for signal processing it is able to evaluate and objectify the follow-up of neurological diseases

[85–89]. The networks topology can be described with a graph-based representation, where the nodes, namely artificial neurons, model the physiological neurons of a biological brain, while the connections model the synapses. The first mathematical model of a neuron, called *perceptron* , was introduced by McCulloch and Pitts in 1943 [90]. The perceptron processes the inputs by multiplying the inputs x_i with corresponding weights w_i, then the weighted inputs are summed up (a bias b is usually added), and lastly the summed weighted inputs are evaluated with an activation function $f(\cdot)$ producing the output y (Eq. 2.1).

$$y(\widehat{X},\widehat{W}) = f\left(\sum_i x_i w_i + b\right) \tag{2.1}$$

The activation function is chosen depending on the problem the artificial neuron should solve, and can be a linear or a non-linear function [91].

During the learning phase, specific algorithms are used to train an ANN model finding the optimal $\widehat{W}$ and $\widehat{B}$, that are those which minimize the error on the training set.

The limitation of the perceptron is its simplicity, since it works only as linear or binary classifier [92]. Minsky *et al.* [92] showed how a relatively simple task, such as the XOR function, can not be performed with a perceptron. Afterwards, new models topology and training algorithms were introduced: associative memories [93], multi-layer perceptron (MLP) [94], back-propagation learning algorithm [94]. These allowed to overcome the perceptron limitations by increasing the complexity of the neural network, introducing intermediate neural layer and modelling more complex functions. Nowadays, ANN generally refers to MLP networks or feed-forward neural network (FFNN), composed by at least one hidden layer fully connected to the others, and propagating information from input to output in only one direction (Fig. 2.6). Equation 2.1 can be then reformulated as Equation 2.2, where l stands for the generic hidden layer.

$$y_k(\widehat{X},\widehat{W}) = f_{l+1}\left(\sum_j w_{l+1,kj} f_l\left(\sum_i w_{l,ji} x_i + b_l\right) + b_{l+1}\right) \tag{2.2}$$

Even if the increase of the network complexity enables the learning of complex tasks, it introduces the drawback about the choice of the optimal number of intermediate layers and nodes. A useful rule of thumb is the Occam's razor [95], for which the simplest model should be chosen among the best performing ones.

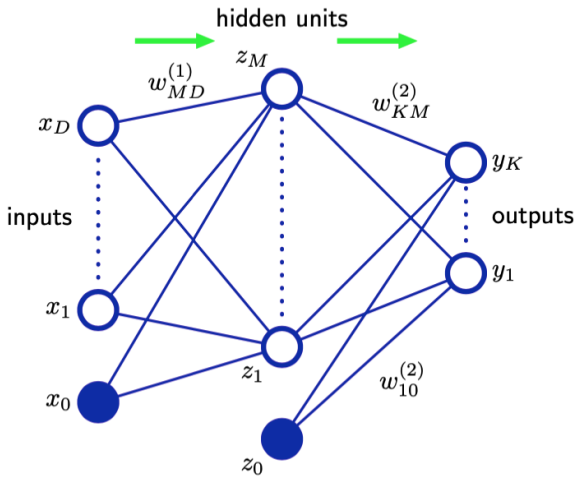


Fig. 2.6 Topology of a multilayer perceptron; it has at least one hidden layer and all layers are fully connected. Arrows denote the direction of information flow through the network during forward propagation. Image from Bishop *et al.* [96].

Artificial Neural Networks: The Learning Process. As stated before, an ANN model can be defined by its neurons and their inter-connections, and the learning process "manipulate" the parameters (i.e. weights and biases) to find the optimal ones. This process is usually achieved looking for the optimum (usually minimum) of an error or cost function. The function is generally defined as the discrepancy between the desired output and the actual output of the model, but its exact form depends on the problem being solved. Most of the real tasks resolvable by supervised learning can be defined as regression or classification problem. For regression, the mean squared error is commonly used as error function:

$$E(\widehat{W}) = \frac{1}{2}\sum_i (y_i - t_i)^2. \tag{2.3}$$

For classification, the cross-entropy error function is used instead:

$$E(\widehat{W}) = -\sum_i (t_i \ln(y_i)(1+t_i)\ln(1-y_i)), \tag{2.4}$$

that can be generalised to multi-class problems with K classes and N samples as [96]:

$$E(\widehat{W}) = -\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ki} \ln y_{ki}. \tag{2.5}$$

The error function can be viewed as the surface defined by the parameter space; it is smooth and continuous, but also non-linear and non-convex, and its optimal parameter set can not be

analytically computed due to the presence of local minima besides the global minimum [97]. Therefore, the process is set up as an iterative procedure and compute a new and adjusted parameters vector $\tau+1$ in each step updating the previous one τ:

$$\widehat{W}_{\tau+1} = \widehat{W}_{\tau} + \Delta\widehat{W}_{\tau}, \tag{2.6}$$

The most common way to calculate the update $\Delta\widehat{W}_{\tau}$ is by using gradient information as:

$$\widehat{W}_{\tau+1} = \widehat{W}_{\tau} - \alpha\nabla E(\widehat{W}_{\tau}) \tag{2.7}$$

where α is called learning rate. This procedure is called gradient descent algorithm and it is one of the most used base algorithm also in most complex model, such as the deep learning ones. As suggested by the name, the error based update component always have a descent direction, due to the gradient of the error function $\nabla E(\widehat{W})$ that always points into the direction of greatest increase of the function. The learning rate α is used to control the length of each step taken in the given direction, preventing variables over-correction, which could lead to non-convergence of the algorithm. For this reason, the learning rate can be used to manipulate how fast a network moves toward an optimal value. As good practise, a decay function can be used to manipulate the learning rate over time (or over the training steps), so that big steps are taken in the beginning of the algorithm and smaller afterwards, stabilising the convergence.

Two different timing approaches can be used to update the parameters vectors: Gradient Descent (GD) and Stochastic Gradient Descent (SGD). With gradient descent all the samples of the training set are processed before a single parameters update of each iteration. This requires to compute and store in memory all the derivative terms (Eq. 2.8), and to sum over the gradients all the samples' contributes to compute a single shift in the model parameters space. When the complexity of the network or the dimension of the dataset make this not feasible, it is possible to update the parameters each sample of after a bunch of samples (Minibatch Stochastic gradient Descent), but ensuring that the model can fit better the data seen in the batch and not the whole dataset. In general it is possible to observe that SGD version converges much faster compared to GD because it does not have to process the whole dataset before an update, but the convergence could reach a less optimal results and oscillate around the best one. However, since the suboptimal result is close enough the desired one, the procedure can be considered as converged. SGD is widely used when the model or the

data make the problem non tractable with reasonable time and computational resources, as for deep network and convolutional neural networks [98].

$$\nabla E(\widehat{W}) = \left[\frac{\partial E}{\partial \widehat{W}_{l_0}}, \frac{\partial E}{\partial \widehat{W}_{l_1}}, \cdots \frac{\partial E}{\partial \widehat{W}_{l_L}} \right], \tag{2.8}$$

Since the gradient needs to be calculated in every update step, error back-propagation is usually used for efficiency. The idea behind error back-propagation is to propagate the resulting error from the output layer back to the input [99]. The error' contribute δ_l to the output error is calculated for every layer. Since the states and the outputs of hidden layers are not known, error terms can not be directly calculated, but it can be estimated by backward propagating the error contribute through the network. Then, layer l receives the error contribute δ_{l+1} from layer $l+1$ and updates it according to the following equation:

$$\delta_l = f'(\widehat{z}_l) \cdot \left[(\widehat{W}_{l+1})^\intercal \delta_{l+1} \right], \tag{2.9}$$

where $\widehat{z}_l$ is the input vector of layer l and $f'(\cdot)$ is the inverse of the activation function; the δ_l contribute is then used for layer $l-1$. Given the layer contribute and the activation a_l, it is possible to compute the gradient of the error function with respect to the parameters of the current layer, as:

$$\frac{\partial E(\widehat{W})}{\partial \widehat{W}_l} = \delta_l a_l. \tag{2.10}$$

This is repeated for every layer [97]. The basic concepts of error back-propagation are illustrated in Fig. 2.7.

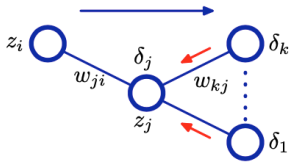


Fig. 2.7 Illustration of the calculation of δ_j for hidden unit j by back-propagation of the δ's from those units k to which unit j sends connections. The blue arrow denotes the direction of information flow during forward propagation, and the red arrows indicate the backward propagation of error information. The error message δ is propagated from output to input and updated in every layer. In each layer, the partial derivatives with respect to the parameters $\widehat{W}_l$ are calculated using the error message. Together, these partial derivatives make up the gradient of the error function ∇E. Image from Bishop *et al.* [96].

Important is to note that the core idea behind error back propagation and gradient descent has been used in a wide range of new algorithms; it can be also applied to deep

learning network, such as convolutional neural network, where the main difference, from an algorithmic point of view, is the different weights and parameters arrangement.

2.2 Deep Learning for Computer-Aided Diagnosis

Deep learning represents a subset of machine learning approaches capable to process data in the raw format [100]. As stated in Section 2.1.2, classical pipeline based on machine learning algorithms requires a pre-processing step to extract meaningful features from data; deep learning approaches, instead, can extract feature automatically. Although first deep learning applications were already suggested and applied at the end of the last century [101, 102], they become popular in the last years thanks to the new parallel computing capabilities provided by the recent breakthroughs in graphics processing units (GPU) [100]. The explosion of deep learning applications started with the CNN architecture AlexNet [103] winning the ImageNet challenge[1] in 2012. From that moment on, they have received an increasing attention in different communities, including image processing and medical image analysis [104], where convolutional neural networks (CNN) became a new processing standard. Their dominant performance involved many other fields based on generic signal processing, leading to the development of several network topologies, and as a consequence, nowadays, any ANN with more than one hidden layer is considered a deep neural network.

The huge growing interest about deep learning techniques is due to their inherent capability of overcoming the drawbacks of traditional machine learning algorithms based on hand-crafted features [50, 55, 105, 106]. Deep learning techniques have also been found to be suitable for the analysis of big data with successful applications to computer vision, speech recognition, pattern recognition, natural language processing, and recommendation systems [80, 105, 107]. The quickness with whom deep learning applications diffused was not immediately followed by a formal theory about convergence, time and topology strategies; in fact, it is possible to use a deep learning models "out-of-the-box" without particular contrivances, other than following best practices, and achieving acceptable performance.

The theory behind the training of neural networks is limited to topologies with one or few hidden layers; whereas the theory behind multilayer networks remains largely without formalisation [108]. Allen-Zhu *et al.* attempted to formalise a convergence theory behind deep learning, and to explain the empirical findings by Goodfellow *et al.* [109], proving that stochastic gradient descent algorithm is able to find a global minima in polynomial time;

[1] https://www.image-net.org

the authors assessed the applicability of their theory to fully connected neural networks, convolutional neural networks, and residual neural networks [108].

The problem of network convergence was faced also by Ioffe *et al.* The authors focused on the difficulties of deep neural networks training due to the changing of layers distribution over the training phase, as the parameters of the previous layers change; this phenomenon, referred by the authors as internal covariate shift, slows down the training by requiring a lower learning rate. Ioffe *et al.* successfully addressed the problem by normalizing layers inputs and performing the normalization for each training batch; this new Batch Normalization layer allows to increase the learning rates and to reduce the effects of the initialization [110]. Nowadays, the Batch Normalization layer is widely used and considered as a standard layer in deep and convolutional neural networks.

Regarding the training time, despite the popularity of deep learning in wide application fields, there is not a well know methodology to predict the timing for a specific problem. Common applications try to infer the training time as a linear extension of the single epoch timing or from the number of operations. These approaches are usually an over-simplification because they ignore secondary aspects of the training such as data loading or non-optimal parallel execution. Starting from the definition of the training time as the product of the training time per epoch and the number of epochs which need to be performed to reach the desired level of accuracy, Justus *et al.* proposed an alternative approach in which a deep learning network is trained to predict the execution time for parts of another deep learning network. The combination of the individual parts provide the prediction of the whole execution time [111].

Despite the deep learning became widely used in many artificial intelligence problems, due to its ability to outperform alternative techniques and even humans, it is not general purpose. The main drawback is the requirement of large amount of data (and annotated output data in most applications); this sometimes biases the researchers to work on specific data that lack of generalization, such as benchmarked datasets, or on tasks where annotation is easy to obtain instead of on the tasks itself. Fortunately, the research community continuously produces and shares new public dataset, and there are techniques that allow to reduce the need of supervision (i.e. transfer learning, unsupervised learning, and weakly supervised learning) [112].

2.2.1 Main Deep Learning Architectures

According to recent surveys on deep learning methods [105, 113], it can be stated that there are five main deep learning architectures: deep neural networks (DNNs), deep recurrent neural networks (RNN), convolutional neural networks (CNNs), autoencoders (AEs) and deep belief networks (DBNs). CNN and AE architectures have been widely used in this thesis, then a brief description will be reported in the next two sections. Brief information about the others architectures can be found in Buongiorno *et al.* [114, 115], while for a detailed discussion it is suggested the book by Goodfellow *et al.* [116].

2.2.1.1 Convolutional Neural Networks

Convolutional neural networks were defined by Goodfellow *et al.* as artificial neural networks that use convolution in place of a general matrix multiplication in at least one layer [113, 116–131]. A CNN is based on weights, biases and non-linear activation functions as ANN but, in addition, considers a mathematical convolution operation in at least one layer. As stated in paragraph *Artificial Neural Networks* of Section 2.1.3, regular fully connected networks compute a transformation of the input by using the weights of the fully connected hidden layers. Each neuron of each layer is connected to all the neurons of the previous one, being independent from the other neurons of the same layer (i.e. there are not connection among neurons belonging to the same layer). Due to their architecture, they are not suitable to process data with grid-like topology (i.e. time series, image data); in detail, due to the huge amount of involved weights, regular ANNs do not scale well in image processing applications; the number of parameters to be learned, indeed, increases rapidly as the image resolution grows up. This because each pixel in the input image counts as one input dimension; as example, if each of these inputs were connected to a hidden layer with a "few 100 hidden units", an image of size 200×200 would already result in $4 \cdot 10^4$ weights per neuron and over $4 \cdot 10^6$ weights for the whole layer. Training all these weights would require a large amount of memory. Furthermore, even overlooking the memory constrain, serialising an image or a grid-like input is not a good idea, because the topology of the input may preserve useful information such as local correlations among neighbouring pixels; this information is destroyed when vectorizing grid-like data.

The convolutional layer, on which CNNs are based, carries out the computational reduction, and made these architecture particularly suitable when facing domains involving multidimensional data, such as images or volumetric data. These layers force the network to extract local features by restricting the receptive fields of hidden units to a certain neighbour-

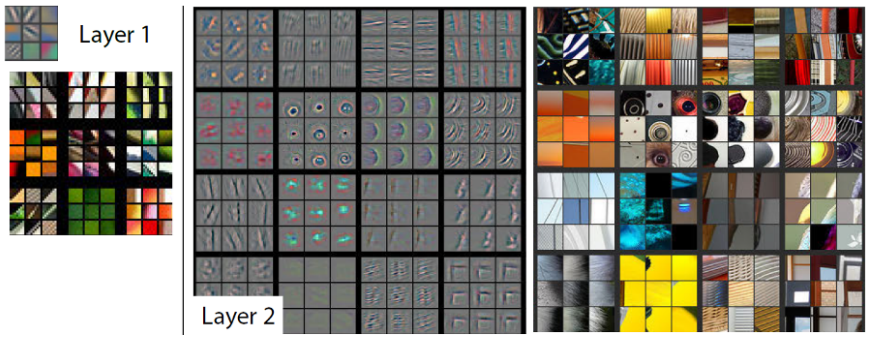


Fig. 2.8 Visualisation of the output of convolutional layers. It is possible to observe how firsts layers allow the extraction of features related to classic ones such as oriented edges, end-points or corners. Image from Lee *et al.* [139].

hood of the input. Such features extraction capability is automatically learned and, mainly in the firsts layers, could be associated to classic ones, such as oriented edges, end-points or corners (Fig. 2.8) [132, 133]. The increase of the network depth leads to more abstract features with higher semantic meaning, but, although such features may be used in the same way of those described in Section 2.1, they would lost significance, especially for the medical context, as the network depth grows [134]. Thus, it would not be possible to link specific features to clinical outcomes, but recent research about radiomic using deep learning are going in this direction [39, 135–138].

Advancing with the analysis of convolutional layers, they can be considered as a set of filters, commonly named kernels, whose weights are learned during the training phase. The filter shape is properly selected and, for a bi-dimensional convolution, the filters slide on two dimension only: heigh and width, whilst the depth size is fixed to the same value of the input space; as example, for RGB images the filters shape of the first convolution layer is $N \times N \times 3$, where N is an odd number lower than the input width and height (usually 3, 5 or 7) and 3 because of the images depth (i.e. color channels). The forward computation consists of a filter convolution across width and height, that is, a filter sliding with dot product among corresponding inputs and filter weights (Fig. 2.9).

To bring back the concept of convolution to the one of neurons and weights, it is possible to consider each unit of a convolutional layer as a neuron that receives inputs from a small neighbourhood of units of the previous layer. The neighbourhood, called local receptive field, is defined by the filter size and indicates the extent of the scope of input data a neuron within a layer is responsible for. All the neurons whose receptive fields comes from the same input plane defined by the convolution operation, are grouped together and share the same weights vector (convolutional kernel weights); the advantage of this is that it maintains

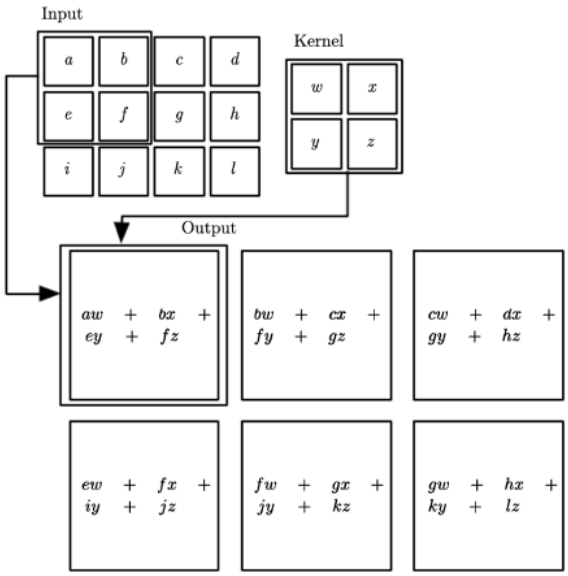


Fig. 2.9 Example of application of the convolution operation with a 2-D kernel. Image from Goodfellow *et al.* [116].

the same "feature extractor" used in one part of the input, across other sections of the input data, leading to a great reduction of the number of parameters to train. The local receptive fields and the weights sharing, are the main breakthrough introduced by CNNs, and with the spatial sub-sampling provided by pooling layers, allow these networks to slightly face affine transformations of the input, such as shift, rotation and scale [96, 132].

The output of a convolutional layers is the so-called feature maps, and since several kernels can be applied for each convolutional layer, a feature maps is a three dimensional volume where width and height are dependent on width and height of the input, and depth is equivalent to the chosen number of filters for the layer. This process can be also understood as convolving the input with several different filters, each one contributing to one feature map of the output.

Besides, the number of filters, there are two more parameters introduced by convolution that influence the size of the feature map: stride and padding. The first is the number of pixels by which the kernel slides over the input image; a larger stride results in reduced width and height of the feature map and a stride of one leave width and height unchanged. The padding can be applied to the borders of the image allowing the application of the filter to bordering pixels; zero- and mirror-padding are the most used.

Since the convolution is a linear operation, they are arranged with non-linear layers provided by suitable activation function, and allowing CNNs to learn non linearities. Although

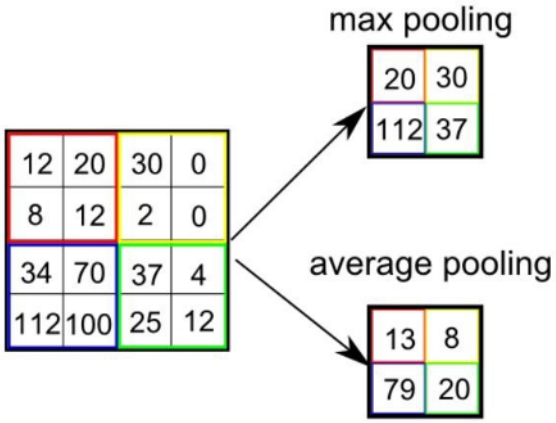


Fig. 2.10 Example of application of the max and average pooling operations with a 2-D kernel.

ANNs employ sigmoidal functions which emulate the neural biological behaviour, their use is not efficient with CNN models, then, simpler functions became popular: Rectified Linear Units (ReLU) (Eq. 2.11) or its derived leaky ReLU, Gaussian Error Linear Unit (GELU), or Exponential Linear Unit (ELU).

$$f(x) = ReLU(x) = max(0,x) \tag{2.11}$$

CNNs introduced also a new type of layer: the pooling layer. Its purpose is to down-sample an input feature map by reducing its width and height, but leaving the depth unchanged. The concept of receptive field is used for pooling layers too. There are several down-sampling implementation (Fig. 2.10), such as max pooling where the maximal value inside a receptive field is taken, or average or mean pooling, where the average or mean value is calculated. Pooling goal is to reduce the feature dimensions and therefore, further reduce the number of parameters (weights, biases) of the network to be learnt. Furthermore, it makes the network invariant to small scaling.

In a convolutional neural networks, several convolutional, non linear and pooling layers can be applied successively, and the deeper the network becomes, the better the ability of the network to extract useful features. The outputs of such a network are high level, low dimensional features of the input that can be finally used for classification, object detection or segmentation purposes; for classification, one or more fully connected layers, with appropriate activation functions, are usually used to compute the class scores.

The higher ability of the network to extract useful features when the depth increases should not mislead, because it clearly leads to an increase of the trained parameters, and,

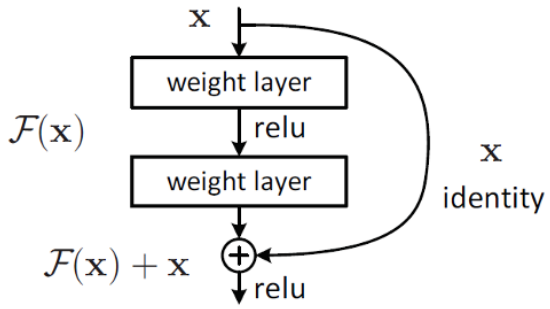


Fig. 2.11 Residual block schema used in Residual Layers.

the depth of CNNs and the complexity of their topology are usually linked to performance outcomes. In detail, accuracy degradation may occur while increasing the network depth [130], but some trick were devised; Residual Layers introduced by Kaiming *et al.* , for example, allow to "skip connections", that is, connect the output of one layer with the input of an earlier one (Fig. 2.11).

About the learning process, based on the complexity of the problem including inputs and outputs dimensionality and the computational resources available, training deep CNNs from scratch may be difficult or, in some cases, impossible. In fact, the training process would require learning a huge number of parameters, including those linked to convolution and those related to the fully connected layers. In such cases, a technique called transfer learning can be used [140]. Transfer learning consists in using a CNN network pre-trained on a very large dataset as initialisation, for learning discriminating new classes, or as features extractors. The first approach is specifically used to fine-tune the parameters of the pre-trained network continuing the training on a new dataset. It is possible to fine-tune the whole network or just a specific section (i.e. the first convolution layers that learn to extract generic features could have fixed weights). The latter approach, instead, considers the pre-trained network as a feature extractor; the fully connected layers (including also some convolutional layers) are removed, and the feature map of the last considered layer are then used as input to new fully connected layers or different classifiers, such as support vector machines.

Literature and applications of CNN topologies to several medical fields are reported in Sections 2.4.1, 2.4.2 and Chapter 3.

2.2.1.2 Autoencoders

Autoencoders (AE) are particular types of neural networks trained with the aim to copy input data to the corresponding output layer [116]; hence, the size of the input layer is the same as

the output one. Autoencoders are trained with an "unsupervised" learning technique able to train neural networks for a representation learning task, e.g. de-noising, feature reduction, clustering, image processing [141–153]. Autencoders topology can implement all the element describe above for classical ANNs and CNNs; AEs that makes use of convolutional layers are called Convolutional Autoencoders. The peculiarity of AEs, with respect to ANNs or CNNs, is the encoder-decoder architecture, and the identification of the output of the encoder as *latent code h*, that is a particular feature set able to represent the input. The encoder part has the role to extract (or codify) each input x into the code $h = e(x)$, while the decoder has to produce the reconstruction, rebuilding the input $r = d(h)$; the learning process tries to minimise the error between the output of the decoder, that is the output of the network, and the input; since the input is also used as output target, the learning process is usually considered unsupervised. A valid autoencoder is not trained to perfectly copy inputs, but to produce outputs that resemble the training data. According to both the internal structure of the networks and their training modalities, there are five main AE families [116]: under-complete AE, regularized AE, sparse AE, denoising AE and variational AE.

Under-complete AE has a specific structure that is able to extract the most representative features contained in the input data. Such property is achieved by imposing the size of code h to a value that is smaller than the dimension of input x. By introducing such a bottleneck the AE should be forced to learn an internal structure that exists in the input data (e.g., correlation among input signals).

Literature and applications of under-complete AE topologies to signals processing are reported in Section 2.4.3 and Chapter 4, respectively.

2.2.2 Deep Learning for Detection Problems

Object detection is the capability to localise and classify objects belonging to different classes inside an image. Among the CNN-based methods for object detection, a particular mention is devoted to the Region-Based Convolutional Neural Networks (R-CNN) family of models. The original R-CNN (Fig. 2.12) was proposed in 2014 by Girshick *et al.* from UC Berkeley [123]. The method fuses region proposal algorithms with CNNs at the purpose of performing object detection. The first part of an R-CNN pipeline is devoted to the categories-agnostic generation of region proposals, that will be subsequently processed looking for desired objects. Several methods are available for the regions generation [154–156]; Girshick *et al.* used selective search [156] for their model, combining the benefit of an exhaustive search and the ability to understand the image structure to guide the sampling process; regions

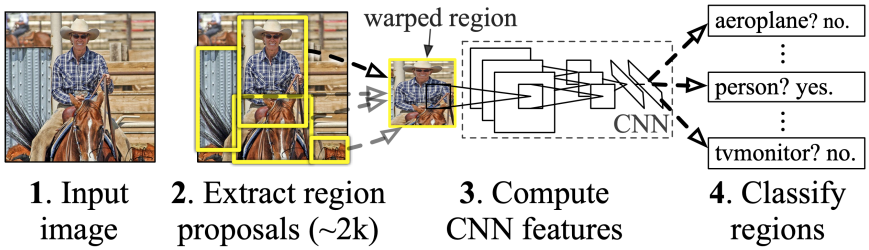


Fig. 2.12 R-CNN object detection system overview. The system takes an input image (1), extracts around 2000 bottom-up region proposals (2), computes features for each proposal using a large CNN (3), and then classifies each region using class-specific linear SVMs (4). Image from Girshick *et al.* [123].

number is fixed to 2000. The extracted regions are warped into a $227 \times 227 \times 3$ volume to be fed into a CNN dedicated to the feature extraction and composed by five convolutional layers and two fully connected layers (these force the input volume dimension to be fixed). The resulting features vector is given as input to a set of class-specific linear support vector machines (SVMs). To improve the localization and narrowing the bounding boxes to the looked object, limiting the background introduced by the region proposal step, Girshick *et al.* used the bounding box regression method proposed by Felzenswalb *et al.* [157]; authors improved Felzenswalb *et al.* solution by applying regression on features computed by the CNN instead of on geometric features computed on the inferred deformable part model locations. The main drawback of using a general purpose region proposal algorithm is that it is not able to learn specific domain patterns, and the fixed number of regions increase training and inferencing time. When proposed, R-CNN outperformed comparable architectures, achieving a mean average precision (mAP) of 53.7% on PASCAL VOC 2010.

In 2015, Girshick *et al.* improved the R-CNN method creating a new object detection network named Fast R-CNN [124]. In Fast R-CNN (Fig. 2.13), the whole input image is fed to the CNN to generate convolutional feature maps. Then, region proposal step is applied on the convolutional feature maps, and the extracted regions are warped into squares. A RoI-pooling layer [158], applying max-pooling on a fixed ROI grid, is adopted to reshape the proposals to a fixed size, so that they can be forwarded to fully connected layers. The RoI feature vector is fed into a softmax layer to predict the class of the proposed region, and into a four outputs regressor predicting the bounding box informations. Fast R-CNN improved performance and speed both for training and inferencing phases, with an mAP of 66% in PASCAL VOC 2012 (compared to 62% of R-CNN), but the use of the slow and time-consuming selective search for region proposal remains the main drawback.

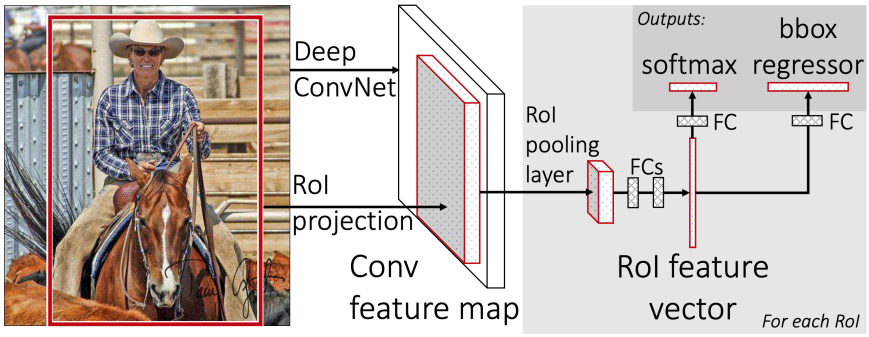


Fig. 2.13 Fast R-CNN architecture. An input image and multiple regions of interest (RoIs) are input into a fully convolutional network. Each RoI is pooled into a fixed-size feature map and then mapped to a feature vector by fully connected layers (FCs). The network has two output vectors per RoI: softmax probabilities and per-class bounding-box regression offsets. Image from Girshick *et al.* [124].

This concern was solved in 2016 with a further evolution of the R-CNN architecture proposed by Ren *et al.* and named Faster R-CNN [125] (Fig. 2.14). The team of Microsoft Research discovered that feature maps computed in the first part of Fast R-CNN can be used to generate region proposals by using a Region Proposal Network (RPN) instead of the slower and not-learnable selective search algorithm. The approach based on RPN differs from the ones used in previous architectures because it exploits anchor boxes, instead of pyramids of images or filters. Anchor boxes are a collection of rectangular bounding boxes proposals and scores obtained by sliding a spatial window over the whole feature map. The sliding window is realized via a small fully convolutional network having as input an $n \times n$ spatial window of the feature map. The scale and aspect ratio of anchor boxes are hyper-parameters and, in order to identify objects at different resolutions, it is required to employ anchor boxes of different shapes. The use of anchor boxes introduce the further advantages of translation invariance and multi-scale feature maps, which led also to a lower model size. Faster R-CNN improves both the speed and the accuracy respect to its predecessors with an mAP of 73.2% on Pascal VOC 2007 and of 70.4% on Pascal VOC 2012.

Non-Maximum Suppression Object detection pipeline usually require a post-processing step, called Non-Maximum Suppression (NMS), to clean results from overlapped bounding boxes; this problem is common in work-flow that operate in a sliding window fashion with overlap between adjacent windows. Starting from the detection box B with the maximum score, NMS suppress all other detection boxes that overlap more than a predefined threshold (i.e. that has an IoU greater than a predefined threshold). The procedure is recursively

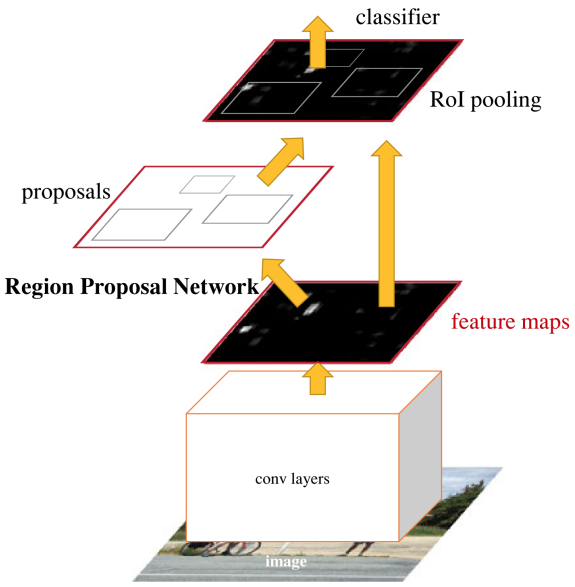


Fig. 2.14 Faster R-CNN network for object detection. Image from Ren *et al.* [125].

reiterated on the remaining boxes until all have been processed. The NMS algorithm is designed so that objects lying within the predefined overlap threshold lead to misses, so the threshold have to be carefully chosen. Soft-NMS by Bodla *et al.* [159] attempts to solve this problem by decaying the detection scores of all other objects as a continuous function of their overlap with B. Authors assess that Soft-NMS improved Faster R-CNN mAP on PASCAL VOC 2006 and MS-COCO by 1.7% and 1.1%, respectively. Anyway, both NMS and Soft-NMS suffer from the problem of not considering the area of the detected objects, then if an object have two detected bounding boxes, one inside the other, the algorithm can choose the smaller box even if it has only a very slightly higher confidence score. A possible NMS modification will be proposed and investigated in Section 3.1.6. NMS procedure is reported in Algorithm 1.

2.2.3 Deep Learning for Segmentation Problems

Semantic segmentation is the capability to classify pixels belonging to an input image in a image-to-image fashion, where the model output are images of class pixel confidence. In order to accomplish this task, most CNN semantic segmentation architectures are based on encoder-decoder networks. The encoder is devoted to the feature extraction process, shrinking the spatial dimensions whilst increasing the depth. The decoder has the task to recover the spatial information from the output of the encoder, and several technique, such as transposed convolution, up-convolution or atrous convolution, can be implemented.

Algorithm 1: Non-Maximum Suppression (NMS) [159]

input : $B_i = b_1, ..., b_{N_i}$, the N_i initial detections
$b_j = (x_j, y_j, w_j, h_j), j = 1, ..., N_i$
$S_i = s_{i_1}, ..., s_{i_{N_i}}$, the N_i initial scores
T_{iou}, the NMS threshold on IoU

output : $B_o = b_1, ..., b_{N_o}$, the $N_o \leq N_i$ final detections
$S_o = s_{o_1}, ..., s_{o_{N_o}}$, the $N_o \leq N_i$ final scores

```
B_o = {};
S_o = S_i;
while B_i is not empty do
    m = argmax(S_o);
    B_o = B_o ∪ {b_m};
    B_i = B_i \ {b_m};
    while b_j ∈ B_i do
        if iou(b_m, b_j) ≥ T_iou then
            B_i = B_i \ {b_j};
            S_o = S_o \ {s_j};
        end
    end
end
```

Activation function and up-sampling (by using the indexes information coming from the corresponding encoder) can be used too. For the pixel classification task, it is used a convolutional layer performing a 1×1 convolution and predicting the class confidence score, instead of the common fully connected layers; then the model output is an $n \times m \times classes$ volume. The lack of fully connected layers allow these models to process input of arbitrary size producing consistently-sized output.

Due to the several application in the medical imaging field, three main approaches based on SegNet, DeepLab v3+ and U-Net architectures are considered in this thesis.

SegNet is a CNN architecture for semantic segmentation proposed by researchers at University of Cambridge [160]. As other semantic segmentation architectures, SegNet is composed of an encoder network and a corresponding decoder, followed by a final pixel-wise classification layer (Fig. 2.15). The encoder is inspired by VGG-16 topology and it is composed by convolutions and max pooling, but discarding the fully connected layers. One clever point of SegNet is that it removes the necessity of learning the up-sampling process, by storing indices used in max-pooling step in encoder and then applying them when up-sampling in the corresponding layers of the decoder. The decoder applies

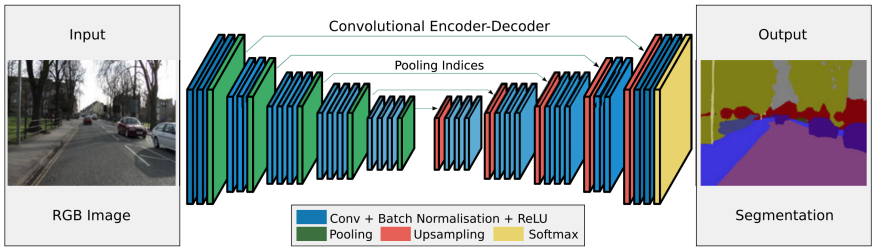


Fig. 2.15 SegNet architecture. Image from Badrinarayanan *et al.* [160].

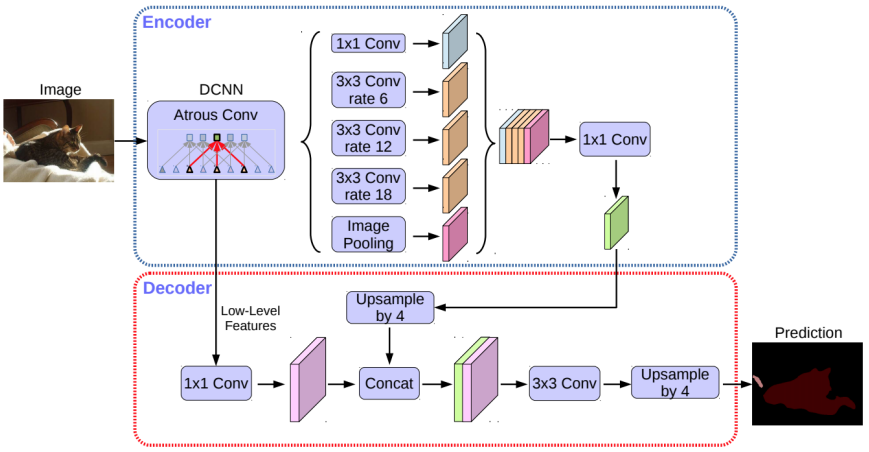


Fig. 2.16 DeepLab v3+ model. Image from Chen *et al.* [165].

convolutions and upsampling by using indexes from the corresponding encoder layer; finally, a K-class softmax is used to predict the output class for each pixel. In literature, SegNet was applied to segmentation tasks such as semantic segmentation of glomeruli for kidney glomerulosclerosis evaluation [161], prostate cancer [162], gland segmentation from colon cancer histology images [163] and brain tumor segmentation from multi-modal magnetic resonance images [164].

DeepLab v3+ architecture has been proposed by Chen *et al.* [166] (Fig. 2.16). One of the interesting novelties introduced, is the atrous convolution (from the French à trous, holes), also known as dilated convolution (Fig. 2.17). The idea has been commonly used in wavelet transform before being adapted to convolutions for deep learning. For one-dimensional signals, the output $y[i]$ generated applying atrous convolution to a signal $x[i]$ with filter $w[k]$ of length k is computed as:

$$y[i] = \sum_{k=1}^{K} x[i + r \cdot k] w[k], \tag{2.12}$$

with r equal to the stride with which the input signal is sampled; $r = 1$ lead Equation 2.12 to the standard convolution. Atrous convolution consents to broaden the receptive field of

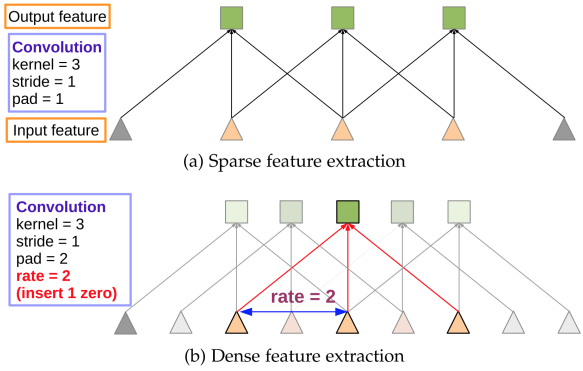


Fig. 2.17 Example of atrous convolution in 1-D. (a) Sparse feature extraction with standard convolution on a low resolution input feature map. (b) Dense feature extraction with atrous convolution with rate $r = 2$, applied on a high resolution input feature map. Image from Chen *et al.* [166].

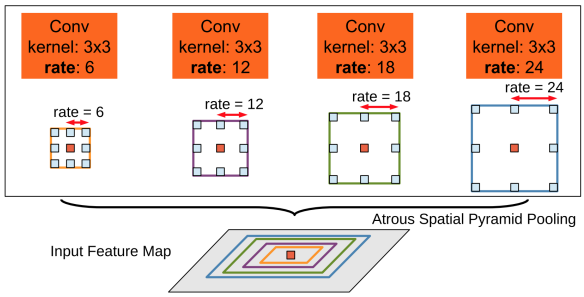


Fig. 2.18 Atrous Spatial Pyramid Pooling. To classify the center pixel (orange), ASPP exploits multi-scale features by employing multiple parallel filters with different rates. The effective receptive fields are shown in different colors. Image from Chen *et al.* [166].

filters to incorporate larger context. Manipulating r is, therefore, a valuable techniques to tune the field-of-view, permitting to identify the right trade-off between context assimilation (large field-of-view) and fine localization (small field-of-view). Chen *et al.* proposed also a modified version of Spatial Pyramid Pooling [167], called Atrous Spatial Pyramid Pooling (Fig. 2.18), in which multiple parallel atrous convolutions with different sampling rates are applied to the input feature map, and then merged together. This help to deal with objects of the same class occurring with different scales in the image. DeepLab v3+ has been used in different medical imaging tasks, such as semantic segmentation of glomeruli for kidney glomerulosclerosis evaluation [161], colorectal polyps [168] and automatic liver segmentation [169, 170].

In 2015, Ronneberger *et al.* introduced U-Net [171], a CNN architecture for supervised learning, originally conceived for the task of 2D biomedical image segmentation (cell and membrane segmentation). As depicted in Fig. 2.19, the model is based on the encoder-

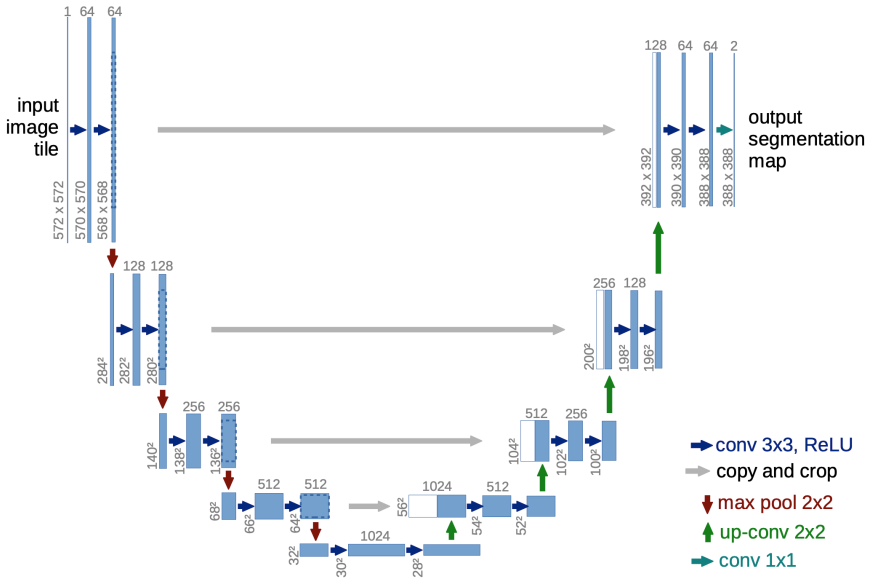


Fig. 2.19 U-net architecture example for 32×32 pixels in the lowest resolution. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations. Image from Ronneberger *et al.* [171].

decoder topology. In detail, each encoder block implements a double stack of convolutions (3×3 kernels) and ReLU, interleaved by max pooling (2×2, stride 2); pooling layers reduce block-by-block the spatial dimensionality of the feature maps, but also increasing their semantic complexity. The corresponding decoders implement the same convolution-ReLU block, but they are interleaved by transposed convolution (2×2) to restore features size. To preserve the spatial localization of the features, the output feature maps of each encoder are also cropped and concatenated to the corresponding ones coming from the transposed convolution. A final 1×1 convolution is used to convert the final features map to the desired number of classes.

U-Net-like networks have been widely applied to automate the task of segmentation; a 3D implementation of U-Net has been proposed by Çiçek *et al.* [172], while Milletari *et al.* proposed a variation of the standard U-Net, called V-Net, for the 3D medical image segmentation [173], presenting the following peculiarities: use of down-convolutions with stride 2 and kernel size 2, instead of 2 max-pooling, use of PReLu [174] non-linearities and adoption of residual connections. All the U-Net-like networks have been easily adapted to suit the domains of interest in several studies, such as segmentation of organs [175], aortic dissections [176] or proximal femur [177].

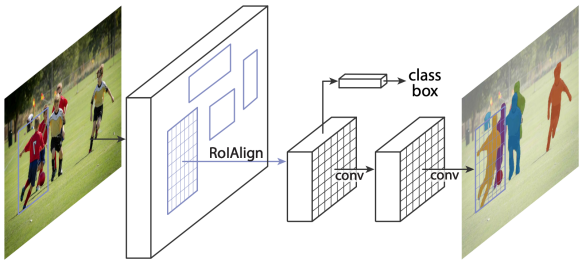


Fig. 2.20 Mask R-CNN framework for instance segmentation. Image from He *et al.* [178].

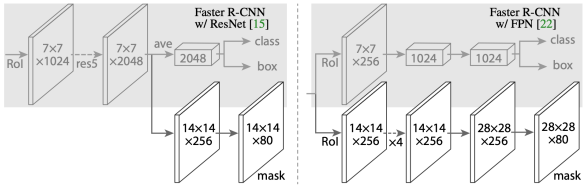


Fig. 2.21 Mask R-CNN head architecture. Image from He *et al.* [178].

2.2.4 Deep Learning for Instance Segmentation Problems

Instance segmentation is the capability to detect objects and classify pixels. These models implement a pipeline able to both accomplish the task of object detection and semantic segmentation, leading to bounding boxes and segmentation masks. The most popular instance segmentation model is Mask R-CNN (Fig. 2.20); it can be considered as an improvement of the R-CNN family detectors, and has been developed by a team of Facebook AI Research (FAIR) in 2017 [178]. Authors discovered that with slight modification of the Faster R-CNN model they can build a model able to perform also mask segmentation with good results.

The overall Mask R-CNN architecture is composed by two parts: a backbone architecture performing features extraction, and a head architecture performing classification, bounding box regression and mask prediction [178]. The backbone architecture could implement any CNN topology such as ResNet [130]; authors exploited a modified ResNet version implementing the Feature Pyramid Network (FPN) proposed by Lin *et al.* [179]. FPN uses a top-down architecture with lateral connections (skip connection) to extract high-level feature maps from different levels of the feature pyramid according to their scale. The authors extended the Faster R-CNN heads starting from the ResNet and FPN papers, by adding a fully convolutional mask prediction branch extracting masks independently for each RoI. Authors proposed two implementations of the head architecture (Fig. 2.21); the first (image on the left), is based on ResNet-C4 (4th stage ResNet) including the compute-intensive fifth stage, while the latter (image on the right) is based on FPN that already includes the fifth *res5* stage allowing a more efficient head using fewer filters. As reported by the authors the addition of

the third branch predicting object mask was a natural and intuitive idea, but since it is distinct from class and bounding-box outputs, it requires the extraction of much finer spatial layout of an object. Then, the main novelty was the introduction of pixel-to-pixel alignment, naturally missing in Fast and Faster R-CNN (RoI-pooling suffers of spatial inaccuracy, as it was not designed for pixel level evaluations). Authors introduced a RoIAlign step that overcome the misalignments between the ROI and the extracted features caused by quantisation/rounding of RoI-pooling; RoIAlign uses bilinear interpolation to compute the exact values of the input features at fixed regularly sampled locations (4), and aggregate the result by means of max or average values. He *et al.* claimed that RoIAlign can improve Mask R-CNN masks accuracy from 10% to 50%, and that results are not sensitive to the exact sampling locations, or to how many points are sampled, as long as no quantization is performed. Further detail on the Mask R-CNN implementation can be found in the original paper [178].

Instance segmentation, as object detection architectures from the R-CNN family discussed before, can adopt NMS to reduce the number of proposals, since many of them can be overlapped especially if the pipeline operates in a sliding window fashion with overlap between adjacent windows.

Mask R-CNN networks have been widely applied for glomeruli segmentation [180] and to implement full pipelines for kidney analysis by means of kidney glomerulosclerosis evaluation [181].

2.3 Performance Evaluation

In order to evaluate the performance of machine learning algorithms, and following, a CAD framework based on them, literature accounts several metrics and performance indexes. The correct evaluation of the system is crucial to assess global performance and the ability of the model to generalise; that is the ability to process unknown new data assuming that they preserve the same informative content or statistical property of the dataset used to create the model. Then, the aim of performance indexes is to evaluate if the learning algorithm is able to model the data. Common procedures are based on the division of the dataset in three subset: training, validation and test; these are used for learning the model parameters from data, evaluate possibles hyperparameters and calculate independent results, respectively. Validation dataset can be also used during the learning phase to manipulate the training (e.g. early stop) and for cross validation, but since it becomes part of the learning process an evaluation on it will be biased; for this reason an independent test dataset is strongly advised.

The performance evaluation is mainly task-dependent and indexes differ among classification, object detection and semantic segmentation; instance segmentation results, instead, can be considered as the joining between object detection and semantic segmentation, then no new indexes are needed. In the following sections, for each category the performance evaluation indexes will be discussed.

2.3.1 Classification Metrics

The classification performance can be defined starting from the output of a binary classifier with Positive (P) and Negative (N) classes. The use of positive and negative terms is common in clinical practice, where a positive result is commonly associated to the presence of a particular condition, and conversely, negative for the absence of it (usually healthy condition). Starting from the number of positive and negative output of a generic classifier, a Confusion Matrix can be defined as in Table 2.5.

Table 2.5 Base Confusion Matrix for metrics computation.

		True Condition	
		Positive	Negative
Predicted Condition	Positive	True Positive TP	False Positive FP
	Negative	False Negative FN	True Negative TN

The confusion matrix allows the definition of four cases: True Positive (TP) indicates the number of instances labelled as Positive, and correctly classified as Positive; True Negative (TN) refers to the number of instances labelled as Negative and correctly classified as Negative; False Positive (FP) refers the number of instances labelled as Negative but misclassified as Positive; False Negative (FN) indicates the number of instances labelled as Positive but classified as Negative. The most used metrics for classification are computed starting from the confusion matrix table and are reported in Equations from 2.13 to 2.20.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.13}$$

$$Precision = \frac{TP}{TP+FP} \tag{2.14}$$

$$Recall\ (Sensitivity) = \frac{TP}{P} = \frac{TP}{TP+FN} \tag{2.15}$$

$$Specificity = \frac{TN}{N} = \frac{TN}{FP+TN} \tag{2.16}$$

$$Jaccard\ Similarity\ Index = \frac{TP}{TP+FP+FN} \tag{2.17}$$

$$Miss\ Rate = \frac{FN}{TP+FN} \tag{2.18}$$

$$F-Measure = \frac{2*Precision*Recall}{Precision+Recall} \tag{2.19}$$

$$MCC = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)*(TP+FN)*(TN+FP)*(TN+FN)}} \tag{2.20}$$

Matthews Correlation Coefficient (MCC) [182], in particular, takes into account false negative and false positive and computes a correlation coefficient between predicted and target classes. It assumes values in the range $[-1;1]$, where 1 indicate perfect prediction, -1 complete disagreement and 0 is equivalent to the random predictor. As stated by Chicco [27], among the usual performance scores, MCC is the only one that takes into account the ratio of the confusion matrix sizes, and it revealed to be a better index of performance than accuracy (Eq. 2.13) or F-measure (Eq. 2.19) on unbalanced datasets.

The use of different indexes is required in real application in which it is necessary to reach a compromise between optimal performance and domain requirements. Higher TPs and TNs than FPs and FNs is the ideal condition, but real applications are ever subject to prediction error due to dataset variability (e.g. noise). When a model wrongly predict a false positive (i.e. type I error), it indicates that a given input imply the existence of a particular output condition, when it does not; a false negative error, dually, wrongly indicates that the output condition is absent (i.e. type II error) [183]. For decision support system, the decision of which error should be minimised is where the compromise relies on. When type I and type II errors can not be both avoided, the chose depends on the application domain; for example, a test for screening the population to identify pathologies in asymptomatic people, should avoid type II errors, increasing as much as possible the sensitivity (or recall) of the system, and can tolerate an higher number of false positives, particularly if the screening test is not harmful or expensive. Other cases, such as diagnostic tests providing a definitive diagnosis,

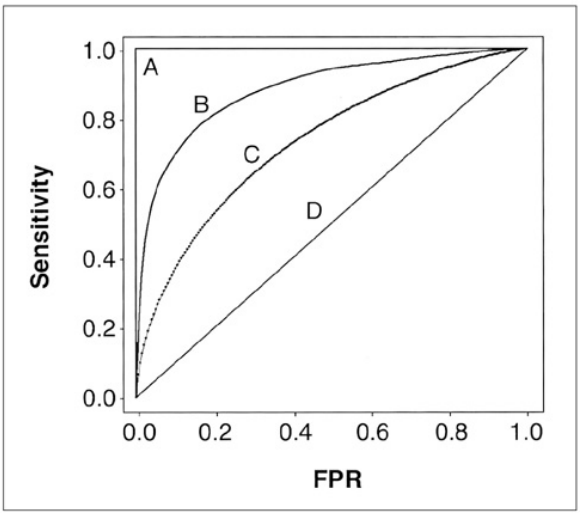


Fig. 2.22 Example of four ROC curves which correspond different values of the area under the curve. A classifier performance may range from chance of curve D ($AUC = 0.5$) to optimal of curve A ($AUC = 1$). Test B, in this example, shows higher AUC and then has a better overall diagnostic performance than test C. Image from Park *et al.* [185]

should avoid both type I and II errors, increasing the specificity of the test and aiming to improve diagnostic precision and accuracy [184].

Another important tool for performance evaluation are the Receiver Operating Characteristic (ROC) curves. The use of ROC curve is strongly encouraged when several classification techniques and hyper-parameters are tuned, and, in general, when it is necessary to evaluate the discrimination ability of different statistical methods (e.g. it illustrates the diagnostic ability of a binary classifier as its discrimination threshold is varied) [185–189]. The ROC curve is a curve created by plotting the True Positive Rate (TPR), or Sensitivity, against the False Positive Rate (FPR), obtained as in Equation 2.21, at various model configuration. Starting from the ROC curves, the Area Under the ROC Curve (AUC) can be calculated; it ranges from 0 to 1, and it allows to assess the better performing models or model configurations. A perfect test has AUC equal to 1, while the chance has an AUC equal to 0.5. Examples are reported in Fig. 2.22.

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{N} = \frac{FP}{FP+TN} = 1 - Specificity \tag{2.21}$$

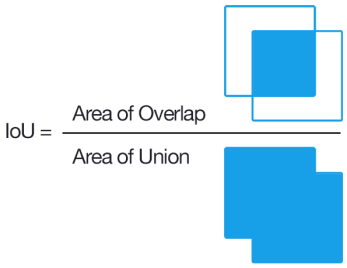


Fig. 2.23 Definition of IoU.

2.3.2 Object Detection Metrics

Object detection problems have to be evaluate by means of different metrics than classification ones. To build a confusion matrix starting from detected objects and their ground truth, it is necessary to define what is a positive or negative detection, and the new background class has to be implicitly included. A common ratio behind the definition of detection, is the extraction of an index about an overlap information between two bounding boxes. The analysis of literature suggests two indexes to be used in detection task: Intersection over Union (IoU) and Intersection over Minimum (IoM). Given two bounding boxes A and B (or any finite sample sets), IoU can be defined as the ratio between the intersection and the union of their areas (Equation 2.22, Fig. 2.23), while IoM as the ratio of the intersection area over the minimum bounding box area (Equation 2.23). Both indexes range from 0 to 1, where 1 indicates a perfect match. In Equation 2.22 and 2.23, $|\cdot|$ denotes the set cardinality operator.

$$IoU(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{2.22}$$

$$IoM(A,B) = \frac{|A \cap B|}{min(|A|,|B|)} \tag{2.23}$$

The two equation allow to define a quantitative index of boxes overlap, then a threshold is necessary to assess the detection success or fail; to avoid problem linked with the presence of adjacent object, common ratio is to set the threshold to 0.5.

There are other two metrics widely used in object detection challenges as PASCAL VOC[2], Google Open Images[3] and COCO[4]: Average Precision (AP) and mean Average Precision (mAP). These indexes offer a global level evaluation, estimating the performances of a model on the whole dataset, and allowing simpler benchmarking; on the contrary they

[2]http://host.robots.ox.ac.uk/pascal/VOC/
[3]https://opensource.google/projects/open-images-dataset
[4]https://cocodataset.org/

did not provide insights on model errors. In detail, AP is defined as the area under the precision-recall curve and mAP is the AP averaged over all classes. A naive implementation of AP is described by the following equation:

$$AP = \int_0^1 p(r)dr \tag{2.24}$$

In literature (e.g. PASCAL VOC) AP is usually calculated by adopting the average interpolated precision value of the positive examples [190], and precision and recall can be defined with respect to a confidence value c, as $p = P(c)$ and $r = R(c)$, respectively; in detail, recall $R(c)$ is the fraction of objects detected with confidence of at least c and precision $P(c)$ is the fraction of correct detections, with:

$$P(c) = \frac{R(c) \cdot N_j}{R(c) \cdot N_j + F(c)} \tag{2.25}$$

In Equation 2.25, N_j is the number of objects in class j and $F(c)$ is the number of incorrect detections with at least confidence c. Note that due to the dependence from N_j, a strong unbalanced dataset can bias the AP comparisons among different classes toward the most represented; Hoiem *et al.* proposed a normalised version of AP applicable if a sufficient dataset rebalancing is not possible [190].

Mean Average Precision (mAP) for K classes can be calculated as following:

$$mAP = \frac{1}{K} \sum_{j=1}^{K} AP_j \tag{2.26}$$

With the definition of what is a detection, a confusion matrix can be built; object detection problems have to implicitly assume that background is a class to be considered in the detection task. When there is only one class of object to be detected, a simple confusion matrix can be defined considering as positive the objects of the class (or all the object classes if there are more classes to detect) and background as negative. Then, a false positive will be the detection of a non-object (e.g. full background or object not satisfying the overlap threshold), and false negative will be an object miss (e.g. object not satisfying the overlap threshold). True negatives can non be considered since the correct background detection is meaningless. All the definition can be extended to the multi-class case considering the mismatch of a class as false positive for another and false negative for itself.

An example of the extension of the classification confusion matrix reported in Table 2.5 to a multi class object detection is reported in Table 2.6

Table 2.6 Example of multi class object detection confusion matrix for metrics computation.

		True Condition		
		Positive_A	Positive_B	Negative
Predicted Condition	Positive_A	TP_A	FN_B FP_A	FP_A
	Positive_B	FN_A FP_B	TP_B	FP_B
	Negative	FN_A	FN_B	-

With the definition of the confusion matrix, all the metrics defined for a classification problem can be extended for object detection (Equations from 2.27 to 2.31). All the metrics have to be defined with a dependence from the threshold confidence c, and indexes involving true negative lose importance.

$$Precision = P_j(c) = \frac{TP_j(c)}{TP_j(c)+FP_j(c)}, \tag{2.27}$$

$$Recall\ (Sensitivity) = R_j(c) = \frac{TP_j(c)}{P_j(c)} = \frac{TP_j(c)}{TP_j(c)+FN_j(c)}, \tag{2.28}$$

$$Jaccard\ Similarity\ Index = \frac{TP_j(c)}{TP_j(c)+FP_j(c)+FN_j(c)}, \tag{2.29}$$

$$Miss\ Rate = \frac{FN_j(c)}{TP_j(c)+FN_j(c)}, \tag{2.30}$$

$$F-Measure = \frac{2*P_j(c)*R_j(c)}{P_j(c)+R_j(c)}. \tag{2.31}$$

All the object detection indexes can be also applied on instance segmentation.

2.3.3 Semantic Segmentation Metrics

Regarding the segmentation tasks, the confusion matrix can be considered, but it has to be defined at pixel-level. In this case, each pixel can be classified as positive or negative (or class_i in a multi-class segmentation). The pixel-level classification, commonly leads to confusion matrices with high numbers that reduce its readability; common practice is to consider its normalised version where each matrix position is divided by the total number of occurrences of the corresponding class. The resulting normalised confusion matrix values will be in the range $[0,1]$.

A new consideration can be done for the metrics. When facing a pixels classification problem the common metrics described before can be used, but they have to be adapted according to the analysed level of detail. In particular, three level of metrics can be defined: class, image and dataset metrics. Class metrics commonly used are:

- accuracy - defined as the number of correctly classified pixels in each class divided by the number of ground truth pixels belonging to that class. Equation 2.13 can be used;
- IoU - calculated as the number of correctly classified pixels divided by the number of all the pixels assigned to that class (both prediction and ground truth). Equation 2.22 can be used considering A and B as predicted and ground truth pixel sets, respectively;
- mean F-measure - defined as the class F-measure averaged over all images.

Image metrics are calculated for each image and are:

- global accuracy - defined as the total correctly classified pixels over the total number pixels;
- mean accuracy - defined as the class accuracy averaged among the classes;
- mean IoU - defined as class IoU averaged among the classes;
- weighted IoU - define as the weighted average of class IoU;
- F-measure - defined as the F-measure of each class of the image.

Dataset metrics refers to global evaluation and are:

- global accuracy - calculated as the number of correctly classified pixels of the dataset divided by the total number of dataset pixels;
- mean accuracy - defined as the mean of class accuracy;
- mean IoU - defined as the mean of class IoU;
- weighted IoU - defined as the weighted mean of class IoU;
- mean F-measure - defined as the mean of image F-measure.

All the semantic segmentation indexes can be applied on instance segmentation. Furthermore, starting from the pixel segmentation it is possible to extract the class bounding boxes, moving the problem from a semantic segmentation task to an object detection one; then, the appropriate metrics can be used. This problem and the required image processing steps will be discussed in the study case presented in Section 3.1.5.

Others semantic segmentation indexes focus on the evaluation of the shape or the surface of the segmented volume of interest. These indexes became popular thanks to SLIVER07

and LiTS challenges [191, 192]; in detail, It is possible to make a distinction between quality measures based on the volumetric overlap and those based on surface distances.

Relevant quality measures based on volumetric overlap are the Volumetric Overlap Error (VOE), defined in Equation 2.33 and the Sørensen–Dice Coefficient (DSC), defined in Equation 2.34. The VOE definition depends on the ratio between intersection and union, namely the Jaccard Index J defined over a set. In all the definitions involved in the quality measures, B denotes the binarised predicted segmented volume P (obtained by thresholding P) and G the ground truth volume; the cardinality operator for a set is denoted as $|\cdot|$.

$$J(B,G) = \frac{|B \cap G|}{|B \cup G|} \tag{2.32}$$

$$VOE(B,G) = 1 - J(B,G) \tag{2.33}$$

$$DSC(B,G) = \frac{|B \cap G|}{|B| + |G|} \tag{2.34}$$

A more general formulation of both DSC and Jaccard Index is the Tversky index $T_{\alpha,\beta}(B,G)$, defined as:

$$T_{\alpha,\beta}(B,G) = \frac{|B \cap G|}{|B \cup G| + \alpha|B - G| + \beta|G - B|} \tag{2.35}$$

Note that $T_{0.5,0.5}(B,G)$ corresponds to DSC(B,G), while $T_{1,1}(B,G)$ corresponds to J(B,G). Besides calculating the overlap error, it is also possible to quantify the Relative Volume Difference (RVD), defined as:

$$RVD(B,G) = \frac{|B| - |G|}{|G|} \tag{2.36}$$

Interesting quality measures based on the external surface distances are the Maximum Symmetric Surface Distance (MSSD, or Symmetric Hausdorff Distance) and the Average Symmetric Surface Distance (ASSD) [192]. These measures are particularly useful for applications like surgical planning, where make a suitable prediction of the mesh of the organs is vital. In order to properly define these distances, let's define a metric space (X,d) where X is a 3D euclidean space and d is the euclidean distance over the 3D euclidean space. Then, let $S(P),\ S(G) \subseteq X$ be the external surfaces of P and G volumes, respectively; it is possible to define a distance function between any two non-empty sets $S(P)$ and $S(G)$ of X, also known as one-sided Hausdorff distance, as in Equation 2.37.

$$h(S(P),(S(G)) = \sup_{s_p \in S(P)} \{ \inf_{s_G \in S(G)} d(s_P, s_G) \} \tag{2.37}$$

Then, MSSD and ASSD, can be defined as Equation 2.38 and 2.39, respectively.

$$MSSD(S(P),S(G)) = \max\{h(S(P),S(G)),h(S(G),S(P))\} \tag{2.38}$$

$$ASSD(P,G) = \frac{1}{|S(P)|+|S(G)|}\left(\sum_{s_P \in S(G)} d(s_P,S(G)) + \sum_{s_G \in S(G)} d(s_G,S(P))\right) \tag{2.39}$$

2.4 Clinical Domains

In this section will be presented the medical domains of the clinical problems faced in the study cases of Chapters 3 and 4. The deeply comprehension of the problem, from the physician point of view, is the first requirement to design a complete CAD system. The experience, the best practice or simply the mind work-flow of domain experts is useful to develop optimal pipeline especially in the field of medical imaging. For each clinical problem will be reported and discussed also the literature novelties with a particular focus on deep learning based approaches.

2.4.1 Digital Pathology

Pathology is the branch of medical science that involves the study and diagnosis of disease through the examination of surgically removed samples, namely biopsy samples, coming from organs, tissues and bodily fluids [193]. Pathologists specialise in a wide range of diseases including cancer, and the most of the cancer diagnoses are made by them; they can also employ genetic studies and gene markers in the assessment of various diseases. A general pathological examination is based on the microscope observation of biopsy samples to help determine if it is cancerous or non-cancerous (benign).

Pathology accounts three main branches based on different source of information and focusing on diverse goal: Surgical Pathology, Cytopathology and Molecular Pathology [193]. Surgical Pathology is the most significant and time consuming branch of pathology and involves the analysis of macroscopic (gross) and microscopic (histologic) tissues; it focuses on the examination with the naked eye or under a microscope for definitive diagnosis of disease and to determine a possible treatment plan. Surgically removed histological sections are processed for microscopic viewing, using either chemical fixation or frozen sections. In particular, the latter case involves freezing the tissue and generating thin frozen slices of the specimen, which are mounted onto glass slides; these are following either stained with

chemicals or antibodies to reveal cellular components for the microscope viewing. Surgical Pathology could be also applied in autopsy examination to determine the cause and manner of death, to evaluate any disease, injury, the state of health, and the appropriateness of any medical diagnosis and treatment before death. Cytopathology, instead, focuses on the diagnose of diseases on a cellular level. It is usually used to aid in the diagnosis of cancer, but also helps in the diagnosis of certain infectious diseases and other inflammatory conditions. In contrast to histopathology, which studies whole tissues, cytopathology is generally used on samples of free cells or tissue fragments that are spontaneously exfoliated or manually removed from tissues by abrasion or fine needle aspiration;. Molecular Pathology, finally, is a relatively recent discipline that has achieved remarkable progresses over the past decade. It is focused on the study and the diagnosis of disease through the examination of molecules within organs, tissues or bodily fluids, and starts from the assumption that many diseases, such as cancer, are caused by mutations or alterations in the genetic code of a subject. The identification of specific mutations allows clinicians to classify a disease and also to choose the appropriate treatment based on the patient genetic. As a result, molecular analysis is leading the way towards personalized medicine allowing the precise prediction of patient's response to therapies, and, thanks to its high levels of sensitivity, it allows the detection of very small tumours resulting in earlier diagnosis and improved patient care and survival.

The main field of interest for this thesis is the analysis of microscopic tissues of surgical pathology, and in particular histopatologic tissues.

Over the last decade, the advent and the proliferation of digital slide scanners lead to the development of a new sub-field of pathology called Digital Pathology. In particular, digital pathology focuses on the data management of Whole Slide Images (WSIs) generated by digitised specimen slides [194]. Digitisation of tissue glass slides facilitates and improves several medical clinical workflows, reducing the needs for storing glass slides on-site and reducing the risk of physical slides to get broken or lost over time. Furthermore, it allows the transfer and the remote consultation (i.e. telepathology) of pathological data, and the following creation and availability of large digital repositories of tissue slides constitute a huge-potential for research, diagnosis, and education [195]. With digital slide archives, such as *The Cancer Genome Atlas*[5] (TCGA), thousands of slides are freely available on-line, leading to new publications about digital pathology image analysis; the large amount of new publications confirms the growing interest in this research area. Finally, the availability of public digital information allows the application of well-known and standardised computational imaging researches for the development of computer-aided diagnosis systems, and

[5]https://www.cancer.gov/tcga

brings the opportunity for new research fields based on innovative techniques, such as deep learning. Also, computer-based image analysis is already available in commercial diagnostic systems, but further advances in image analysis algorithms are required in order to fully realise the benefits of computer-aided diagnosis systems for digital pathology in medical discovery and patient care [196].

The abilities to quantitatively characterise disease information coming from multiple biological scales and dimensions have the potential to enable the development of preventive strategies and medical treatments precisely targeted to each class of patients. Classically, pathologists tissues classification was based on manually-recognised patterns, and recent researches demonstrated that, in some cases, image analysis algorithms could reproduce the pathologist-rendered diagnoses [197]. The advances in pathology imaging technologies, along with comparably advances in the "-omics" and radiology domains, are revolutionising the medical ability to rapidly capture and exploit vast amounts of multi-scale and multi-dimensional data in combination with genetic background.

Since the use, the analysis and the application of innovative techniques on whole slide images is a core aspect for the aim of this thesis, the following sections will delineate the main challenges in the WSIs management and processing, and will report the main literature about these applications on the chronic kidney disease study case.

2.4.1.1 Main Challenges in Digital Pathology

The analysis of digitalised histological tissues introduces new problems compared to standard image processing tasks. The main challenges when dealing with digitalised slides dataset are: images resolution, dataset unbalancing and stain color variation.

Image resolution problem. Digitalised histopathological images, and generally WSIs, are high resolution acquisitions of specimen slides; they could reach dimensions higher than $50000 \times 50000 pixels$ with usually three colour channels. This huge dimension make the direct processing of the whole image an infeasible task and most of the time it is impossible to process the entire image with deep learning or even common image processing algorithms. A simpler solution could be to reduce the original image size by undersampling, but a good trade-off between final image size and acceptable information lost should be found [198] (consider that exists a precise correspondence between pixel and spatial resolution of the real tissue sample).

Common WSIs are usually composed by one or more histological tissue sections surrounded by background; as observable in Fig. 2.24 detecting these sections and reducing the

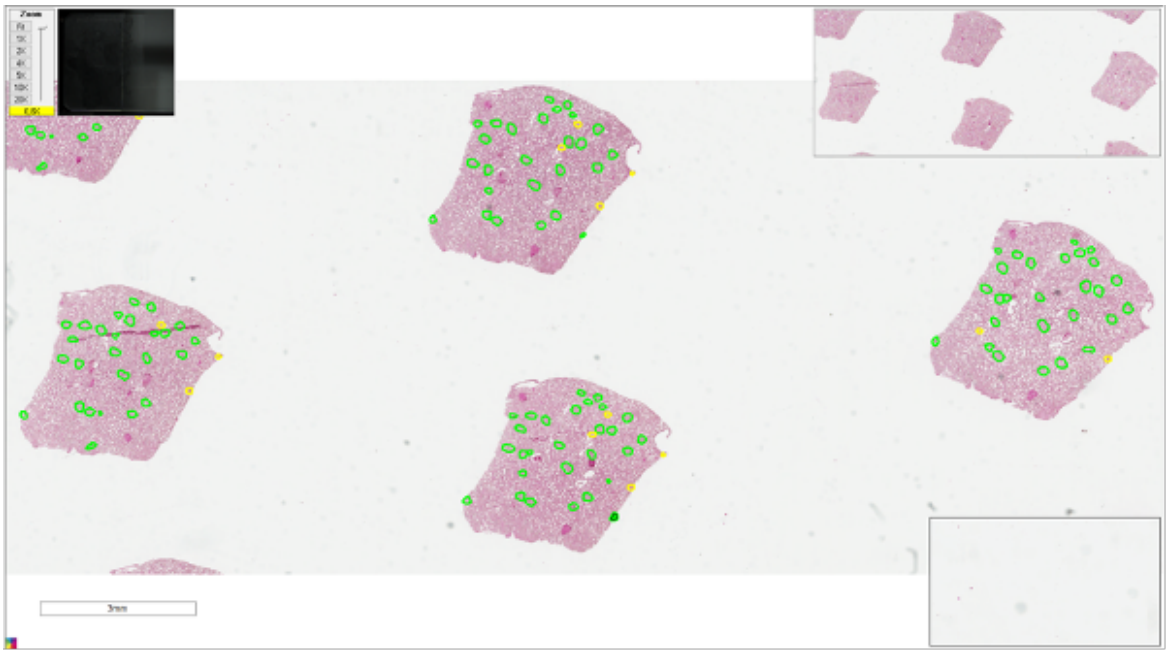


Fig. 2.24 Example of full kidney biopsy in ImageScope ($44999 \times 19241 \times 3$ - 4.04*GB*).

region of interest could be a reasonable strategy. Furthermore the knowledge of the tissues pose, allows its rotation to generate the minimum circumscribed bounding box leading to a size reduction of up to 50% [199–201] (Fig. 2.25).

Other approaches, aiming to retain maximum information, applies the processing algorithms on only a patch at a time of the whole image. This sliding window approach reduce the necessity to adapt the resolution of the images to the available computational resources, and lets the application of algorithms of any complexity requiring only the possibility to project the results to the original image if needed. The introduced drawback is the right choice of the windowing size and overlap between contiguous patches. Literature reports examples of the application of this techniques also in non-medical-contexts [162, 202, 203]; Meng *et al.* used this approach to perform accurate small object detection from large images (but lower than WSIs). The authors used small patches as input to a variant of Single Shot Multibox Detector (SSD) using VGG-16 as backbone network, resulting in a patch-level object detection. They used image pyramid to obtain scale invariance and projected the patch-level detections to original images. Hung *et al.* used Faster R-CNN on bright-field microscopy images of malaria-infected blood, outperforming their traditional segmentation and machine learning baseline; the authors used a low patch size reaching a feasible training and an implicit augmentation of the dataset. Kawazoe *et al.* used Faster R-CNN for kidney' elements detection in kidney biopsies slides; the authors chosen a proper window size and

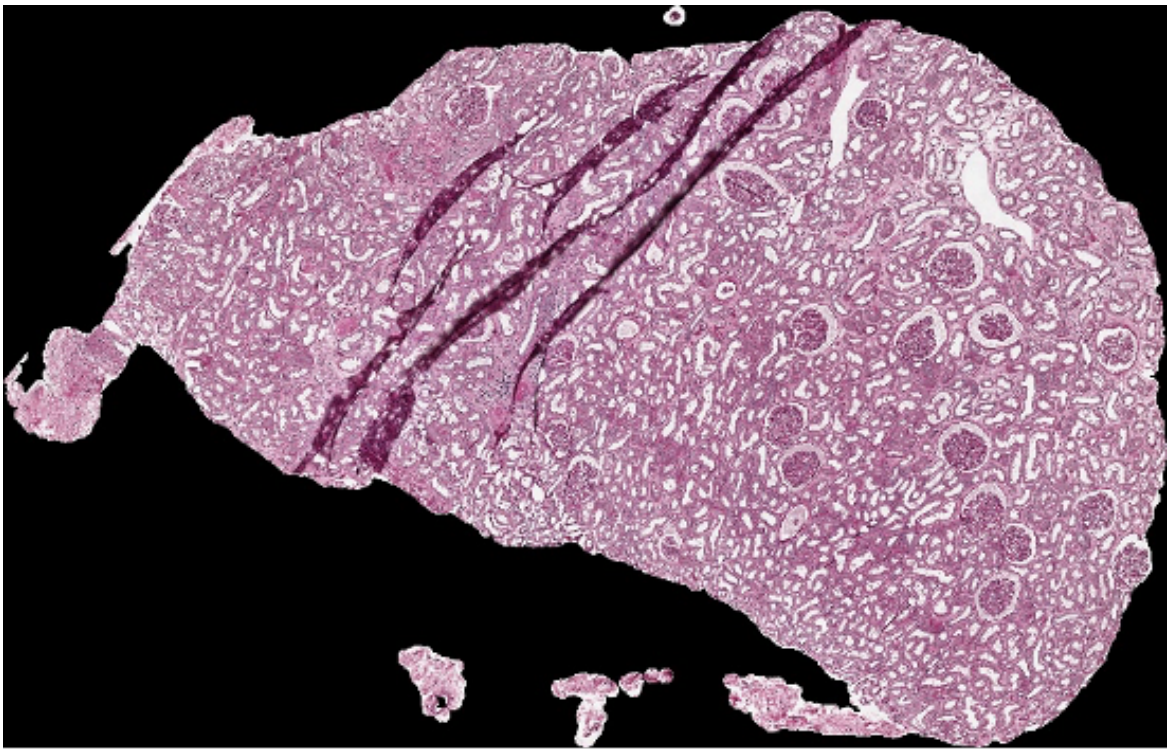


Fig. 2.25 Example of kidney section rotated to minimize the circumscribed bounding box ($2491 \times 1591 \times 3$ - $11.34MB$).

overlap to ensure that the element of interest to be detected could be included in at least one patch. Since this approach is of interest for this thesis, further information about the clinical domain and the paper goal will be investigated in Section 2.4.1.2.

Unbalanced data problem. The second problem that should be addressed in digital pathology is the data unbalancing; it did not occur only in the pathology domain, but it is also really common in all the field involving machine learning applications. The common problem in real life applications of machine learning and deep learning based classifiers is that some classes have a significantly higher number of examples in the training set than others. The problem has been comprehensively studied in classical machine learning, establishing that it can have significant negative effects on training traditional classifiers [24, 204]. Several methodologies are available to deal the problem, and the most commons make use of sampling approaches: oversampling or undersampling. A simplified implementation of the first one consist in the random replication of some data, whilst more sophisticated approaches generate new data starting from a model that usually preserve the statistic or the informative content of the original data, or apply some transformations, such as interpolation or image

augmentations; example applications on WSIs will be presented in the following *Stain color variation problem* paragraph. An opposite procedure is taken with undersampling, in which a careful random selection of samples is performed with a proportion among classes adapted to rebalance the dataset. This technique is usually effective with large datasets, where a reduction of samples size is more affordable.

Other approaches manipulate the learning algorithm used to train the model to achieve the balancing goal; in particular, custom cost functions are usually used introducing different weights to the misclassification and manipulating the direction of the learning algorithm [175, 205]. The study case presented in Section 3.3 make use of this solution.

Buda *et al.* conducted a systematic analysis on the class imbalance problem in deep learning using three benchmark datasets of increasing complexity: MNIST, CIFAR-10 and ImageNet. The authors investigated the effects of imbalance on classification and performed an extensive comparison of several methods to address the issue. Oversampling resulted the more suitable balancing techniques for convolutional neural network allowing also to reduce overfitting [206].

The unbalanced dataset problems affects medical applications particularly; it is really common to deal with dataset in which healthy samples are one or more order of magnitude numerous than non-healthy. Regarding medical imaging, the lack of data is one of the main problem to deal with; the application of complex model, such as the ones introduced by deep learning, requires large dataset to avoid overfitting and the annotation procedure, essential for supervised learning, is also really tedious. Commons applications use image transformations to augment the dataset, customising the core operations to suit the data nature and the research goals. Examples will be detailed in the following *Stain color variation problem* paragraph, whilst a study case affected by and dealing with this problem will be discussed in Sections 2.4.1.2 and 3.1.

Stain color variation problem. Stain variation is the phenomenon observed when distinct pathology laboratories stain tissue slides that exhibit similar but not identical color appearance [207]. Due to this color variability, image processing algorithms and complex machine learning models, such as convolutional neural networks, could have difficulties to generalise and maintain performances among different dataset.

Telleza *et al.* deeply analysed the problem of stain color variation and proposed several solutions based on two categories: stain color augmentation and stain color normalization [207]. The first category consist of image processing transformations aiming to reproduce the variability of datasets, and could be also used as oversampling techniques (see *Unbalanced*

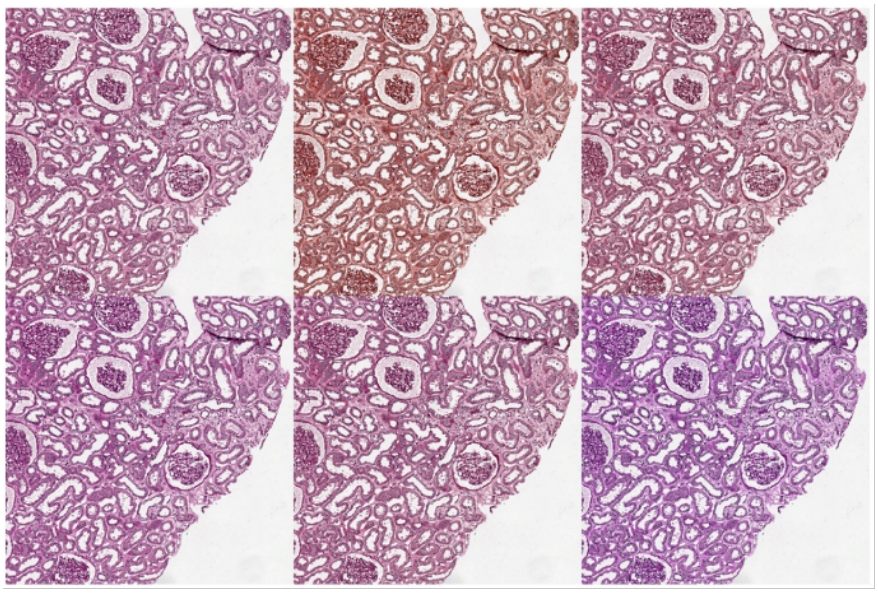


Fig. 2.26 Examples of HSV perturbation on non-sclerotic glomeruli. In order left-right and top-down: original image; $\Delta H = +0.12$, $\Delta S = +0.04$; $\Delta H = +0.04$, $\Delta S = -0.01$; $\Delta H = -0.03$, $\Delta S = +0.06$; $\Delta H = +0.00$, $\Delta S = +0.01$; $\Delta H = -0.09$, $\Delta S = +0.04$.

data problem paragraph above). The most interesting augmentation proposed by Telleza *et al.* , and useful for the aim of this thesis, are the colour shift in the Hue-Saturation-Value (HSV) colour space and the morphology transformation. The first augmentation consists in a randomly shift of hue and saturation channels in the HSV color space, aiming to enrich the color distributions of the dataset with slight differences; this application do not affect the morphology of the images but the color appearance only, simulating stain color variations. Example of HSV perturbation applications on non-sclerotic and sclerotic glomeruli are reported in Fig. 2.26 and Fig. 2.27, respectively.

Conversely, the morphology transformation are a set of morphological perturbations (size, texture and shape) that do not modify the color appearance; the most interesting ones are: scaling, additive Gaussian noise (that is signal-to-noise ratio perturbation), Gaussian blurring (able to mimic out-of-focus artefacts) and elastic deformation. Simard *et al.* proposed an interesting procedure to apply elastic deformation on images [208]. The procedure was proposed for the analysis of visual documents, but then it had a widespread application in medical imaging, as also shown by U-Net authors [171]. Image distortions such as translations, rotations, and skewing can be generated by applying affine displacement fields, and it is done by computing for every pixel a new target location with respect to the original location. A general displacement field could be described by equations $\Delta x(x,y) = \alpha\tilde{x}$ and $\Delta y(x,y) = \alpha\tilde{y}$ defining that the new pixel is shifted by a scale factor α from the origin

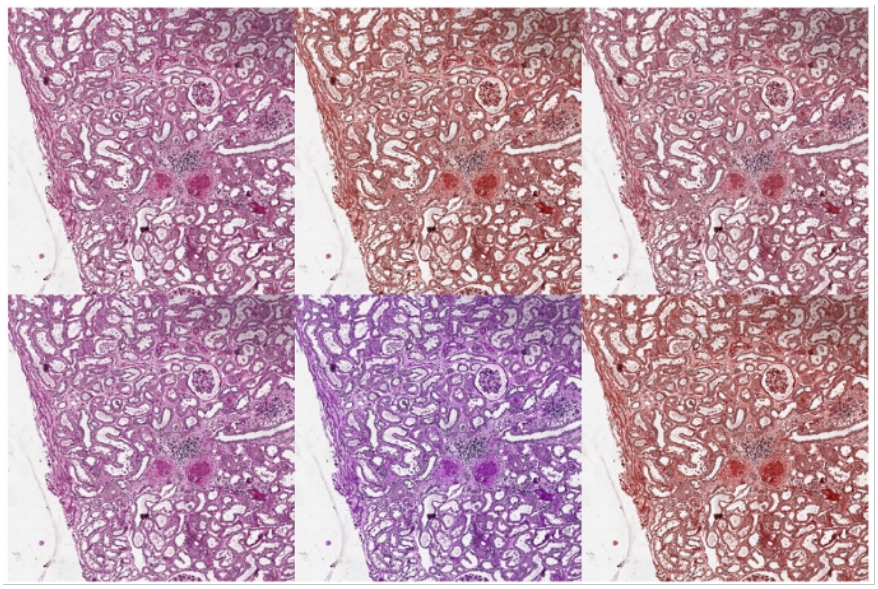


Fig. 2.27 Examples of HSV perturbation on sclerotic glomeruli. In order left-right and top-down: original image; $\Delta H = +0.18$, $\Delta S = +0.03$; $\Delta H = +0.06$, $\Delta S = -0.06$; $\Delta H = -0.04$, $\Delta S = -0.02$; $\Delta H = -0.11$, $\Delta S = +0.10$; $\Delta H = +0.18$, $\Delta S = +0.09$.

location $(x,y) = (0,0)$. Since α is a real value, the new position may not be a valid pixel integer position, then the new gray level has to be computed by interpolating the pixels surrounding the new position; for simplicity, authors suggest bilinear interpolation algorithm. To reproduce an elastic deformation, the authors, used a random displacement fields, define as $x(x,y) = rand(-1,+1)$ and $x(x,y) = rand(-1,+1)$, where $rand(-1,+1)$ is a random number between -1 and $+1$, sampled by a uniform distribution. The fields were then convolved with a Gaussian of standard deviation σ (in pixels) and, finally, the displacement fields are multiplied by a scaling factor α that controls the intensity of the deformation. The authors assess that using larger σ results in small values because the random values average 0 and the normalised displacement field (to a norm of 1) is then close to constant with a random directions; smaller σ, instead, makes the field looks like a completely random field after normalization. In general increasing the value of σ leads the displacements operate as an affine distortion and in extreme cases (very large values) as translations; for intermediate σ values, instead, the displacement fields look like elastic deformation, with an elasticity coefficient of σ. Examples of displacement fields are depicted in Fig. 2.28, while example of elastic deformation applications on non-sclerotic and sclerotic glomeruli are reported in Fig. 2.29 and Fig. 2.30, respectively.

Others minor and classic augmentation, but useful for the case studies of this thesis are image rotation and mirroring.

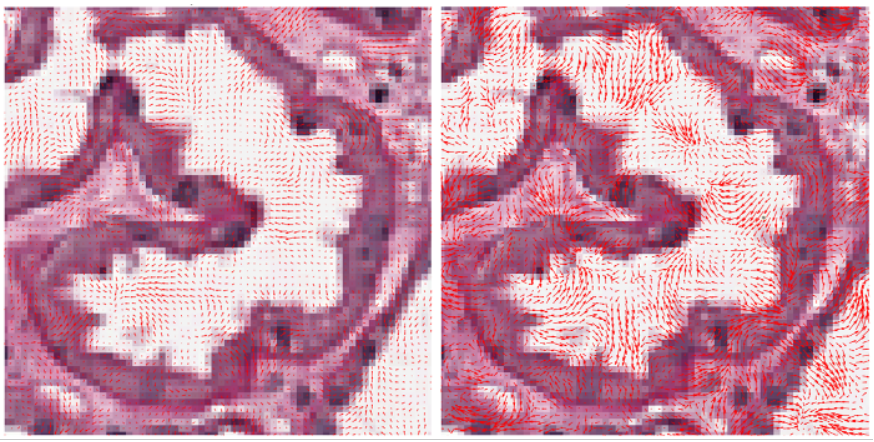


Fig. 2.28 Examples of displacement fields: $\sigma = 4.80$, $\alpha = 143$ (left); $\sigma = 4.63$, $\alpha = 281$ (right).

The second category proposed by Telleza *et al.* is the stain color normalization. It operates reducing color variation by using a normalization function that maps any given color distribution to a template one; this eliminate the problem of stain variance and the generic learning model no longer requires to generalise to unseen stains. Grayscale transformation is the simpler example of normalization function that transforms images from colour to grayscale space; this transformation is useful under the hypothesis that color information may be redundant and most of the information is present in morphological and structural pattern. Nevertheless, the authors observed low performance with their benchmark dataset suggesting that colours preserve important information. Reinhard *et al.* proposed a simple statistical analysis to transfer the color informations of one image to others, by choosing a proper source image and applying its characteristics to another one [209]. This technique can be used to normalize new images and reduce color variation inside or between datasets [210].

Others colours manipulation are based on histogram modifications. Histogram equalization (Algorithm 2) is an image processing technique that aims to adjust image contrast imposing the equal distribution of the colour intensity levels over the whole brightness scale; it allows contrast enhancement for intensity values closer to histogram maxima and contrast diminution near minima [211]. When the colour space dimension is greater than one, as in WSIs, histogram equalization have to be applied on colour space in which colour information is described by one channel only, such as LAB or HSV (the Algorithm 2 should be applied on the L and S components for LAB and HSV, respectively). As evident from the Algorithm 2, the transformation uses information dependent by the histogram of the original image, linking the equalisation performance to his goodness. In particular, images with significant lighter or darker regions will not benefit from the standard histogram equalization (an example is the

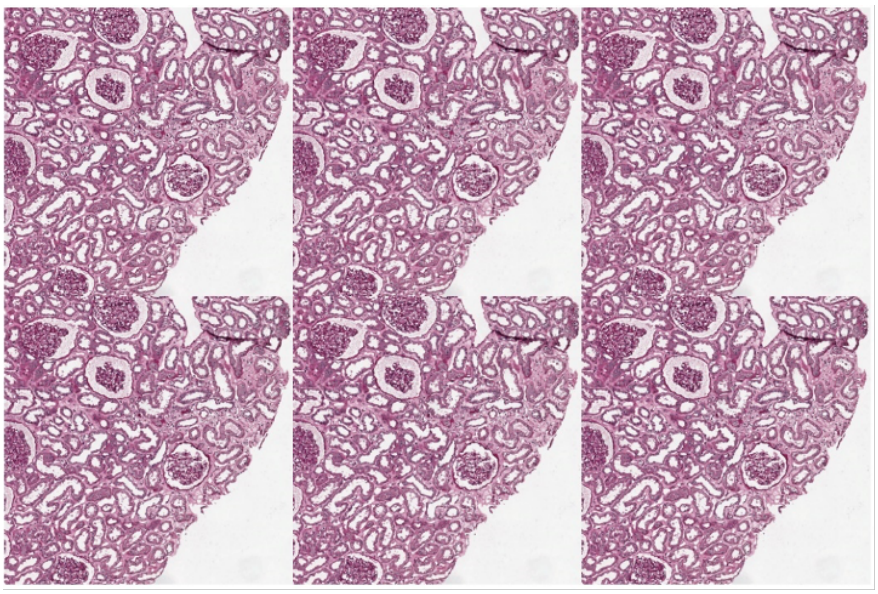


Fig. 2.29 Examples of elastic deformation on non-sclerotic glomeruli. In order left-right and top-down: original image; $\sigma = 4.07$, $\alpha = 101$; $\sigma = 4.14$, $\alpha = 127$; $\sigma = 3.64$, $\alpha = 204$; $\sigma = 3.59$, $\alpha = 277$; $\sigma = 4.29$, $\alpha = 180$.

wide white background present in WSIs). To overcome the problem Adaptive Histogram Equalization (AHE) can be used. It computes several histograms, each corresponding to a different region of the image, and uses them to redistribute the lightness values of the image, resulting in the improvement of local contrast and in the enhancement of image edges. A possible drawback is the contrast amplification of near-constant regions leading to an increase of the noise of these regions. This problem is limited by the use of particular adaptive histogram equalization, such as Contrast Limited AHE (CLAHE) [212], that aims to limit the contrast amplification (Fig. 2.31).

The last analysed stain color normalization is the histogram matching. When automatic enhancement is desired, histogram equalization is a good approach to consider because the results are predictable, but no assumption or constraint about the histogram shape are applied. Histogram matching is based on transformations aiming to obtain an histogram corresponding to a specified one [213]; note that histogram equalization can be considered as the particular case where the specified histogram is uniformly distributed [214].

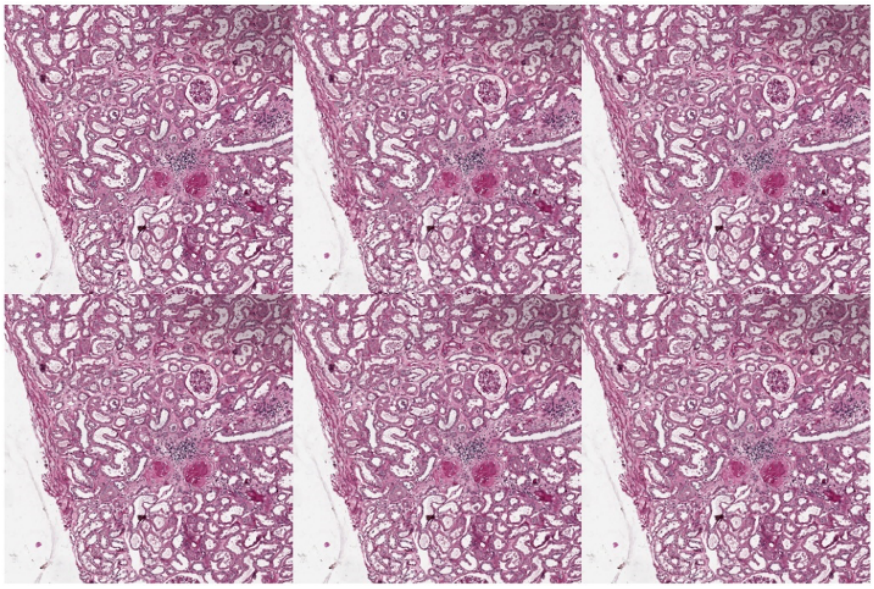


Fig. 2.30 Examples of elastic deformation on sclerotic glomeruli. In order left-right and top-down: original image; $\sigma = 3.01$, $\alpha = 270$; $\sigma = 3.35$, $\alpha = 135$; $\sigma = 4.55$, $\alpha = 119$; $\sigma = 3.16$, $\alpha = 104$; $\sigma = 3.79$, $\alpha = 176$.

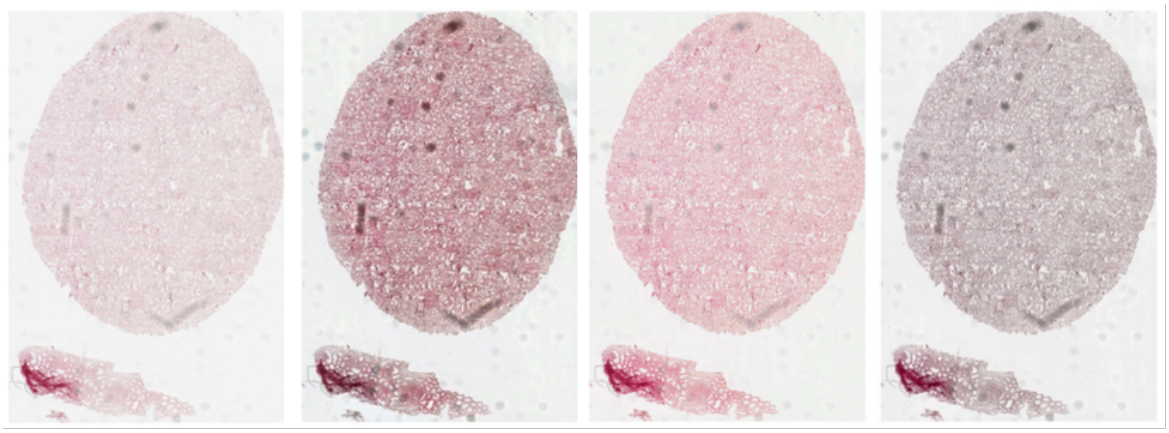


Fig. 2.31 Example of histogram equalization with CLAHE algorithm. From left to right: original image, CLAHE applied to each channel of RGB, CLAHE applied to saturation channel of HSV, CLAHE applied to lightness channel of LAB.

Algorithm 2: Histogram equalization from Sonka *et al.* [211]

• **Input:** Image I of size $N \times M$ and G grey levels.

1: Initialise to 0 an array H of length G

2: $\forall$ pixel $p \in I$ with intensity g_p, perform $H[g_p] = H[g_p] + 1$ (i.e., create the histogram of I)

3: Let g_{min} be the minimum g for which $H[g] > 0$ (i.e., the lowest grey value occurring in I)

4: Create the cumulative histogram H_c, as:

$$H_c[0] = H[0],$$
$$H_c[g] = H_c[g-1] + H[g],\ g = 1, 2, \ldots, G-1$$

5: Set $H_{min} = H_c[g_{min}]$

6: Set $T[g] = round\left(\frac{H_c[g] - H_{min}}{MN - H_{min}}(G-1)\right)$

7: Rescan the image and write the output image with grey levels $g_q = T[g_p]$

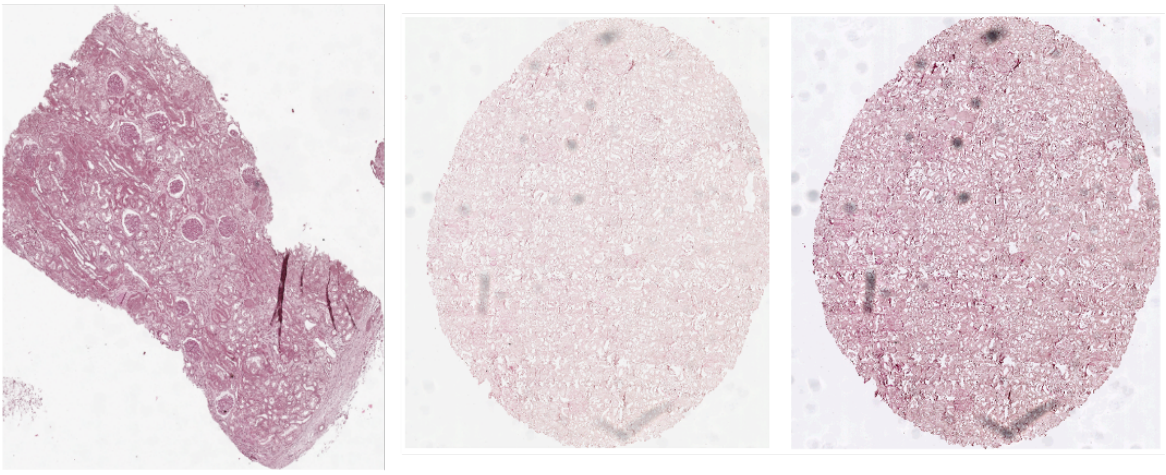


Fig. 2.32 Example of histogram matching: source high-contrast biopsy (left), target low-contrast biopsy (center), output (right).

2.4.1.2 Chronic Kidney Disease

Chronic Kidney Disease (CKD) is a pathological condition consisting in a functional degeneration of the kidney. CKD is the 12th cause of death with up to 1.1 millions of cases worldwide, and the increased mortality of the last years makes it one of the fastest rising causes of death, alongside diabetes and dementia [215, 216]. Kidney transplantation represents the primary therapy which is more effective than dialysis treatment in terms of long-term mortality risk and, at the same time, has a smaller impact on the public health system [217, 218]. Liyanage *et al.* estimated that, in 2010, 2.6 million people as against 4.9 million of patients, received renal replacement therapy worldwide, suggesting that at least 2.3 million people might have died prematurely because appropriate therapy could not be accessed [219].

Due to the increasing necessity of kidney transplants [220], different studies tried to widen the criteria of inclusion, which generally exclude kidneys considering the age of the donor and some characteristics related to quality and dimension of kidneys [221, 222]. Moore *et al.* performed a comparison between dual kidney transplantation (DKT) from expanded criteria donors (ECDs) and single kidney transplantation (SKT) from concurrent ECDs and standard criteria donors. The authors assessed that the use of dual kidney transplantation from marginal donors is a viable option and that renal function can be achieved provided that both kidneys are transplanted into a single recipient [223].

Remuzzi *et al.* proposed techniques to assess the kidney condition by histological biopsy [224]. The evaluation criterion, called Karpinski score, is based on the microscopic examination of the main kidney functional areas corresponding to four compartments: glomerular, tubular, interstitial and vascular. Each compartment receive a score ranging from 0 to 3, where 0 corresponds to normal histology and 3 to the highest degree of global glomerulosclerosis, tubular atrophy, interstitial fibrosis or arterial and arteriolar narrowing [224–226]. As a result, kidneys with a Karpinski score ranging from 0 to 3 and from 4 to 6 are considered suitable for single and dual transplant, respectively. Table 2.7 reports the Karpinski score. The study cases faced in this thesis focus on the automatic evaluation of kidney biopsies, dealing with one of the four pathological conditions evaluated in the Karpinski score: glomerulosclerosis.

Table 2.7 Semi-quantitative scale for renal biopsy scoring. Table from [225].

	Score		Description
Glomerular		*0*	no globally sclerosed glomeruli
		1	$< 20\%$ global glomerulosclerosis
		2	$20-50\%$ global glomerulosclerosis
		3	$> 50\%$ global glomerulosclerosis
Tubular		*0*	absent
		1	$< 20\%$ of tubules affected
		2	$20-50\%$ of tubules affected
		3	$> 50\%$ of tubules affected
Interstitial		*0*	absent
		1	$< 20\%$ of cortical parenchyma replaced by fibrous connective tissue
		2	$20-50\%$ of cortical parenchyma replaced by fibrous connective tissue
		3	$> 50\%$ of cortical parenchyma replaced by fibrous connective tissue
Vascular	Arteriolar narrowing (or hyaline arteriosclerosis)1	*0*	absent
		1	increased wall thickness but to a degree that is less than the diameter of the lumen
		2	wall thickness that is equal or slightly greater than the diameter of the lumen
		2	wall thickness that far exceeds the diameter of the lumen, with extreme luminal narrowing or occlusion
	Arterial sclerosis (or intimal fibrous thickening - fibroplasia)1	*0*	absent
		1	increased wall thickness but to a degree that is less than the diameter of the lumen
		2	wall thickness that is equal or slightly greater than the diameter of the lumen
		2	wall thickness that far exceeds the diameter of the lumen, with extreme luminal narrowing or occlusion

[1]For the vascular lesions, both arterioles and arteries are evaluated separately. However, for the final vascular score, the most severe lesion of either arterioles or arteries determines the final grade.

The evaluation of global glomerulosclerosis requires detection and classification of all the glomeruli present in a kidney biopsy, and relies on their discrimination between sclerotic (non-healthy) and non-sclerotic (healthy). A glomerulus is part of the nephron, the functional renal unit involved in blood filtration; the main distinctions between sclerotic and non-sclerotic glomeruli are the shape of the Bowman's capsule, the different dimension and the different texture due to blood vessels. Non-sclerotic glomeruli usually have an elliptic shape and are characterized by the presence of the Bowman's capsule separated from the capillary tuft with the mesangium by the Bowman's space. The ensemble of the nuclei of cells (blue points), the capillaries lumen (white areas) and the mesangial matrix (regions with similar tonality and different levels of saturation) shows a particular texture commonly called pomegranate texture. Sclerotic glomeruli, instead, are characterized by an increase in the extracellular matrix that obliterates the capillaries lumen and by a reduced or absent Bowman's space due to collagenous material. Fig. 2.33 and Fig. 2.34 report an example of non-sclerotic and sclerotic glomerulus, respectively. The Karpinski score takes into account the glomerulosclerosis information considering the ratio between sclerosed glomeruli and the overall number of glomeruli; for this reason the correct glomeruli detection and classification is crucial for the score assignment.

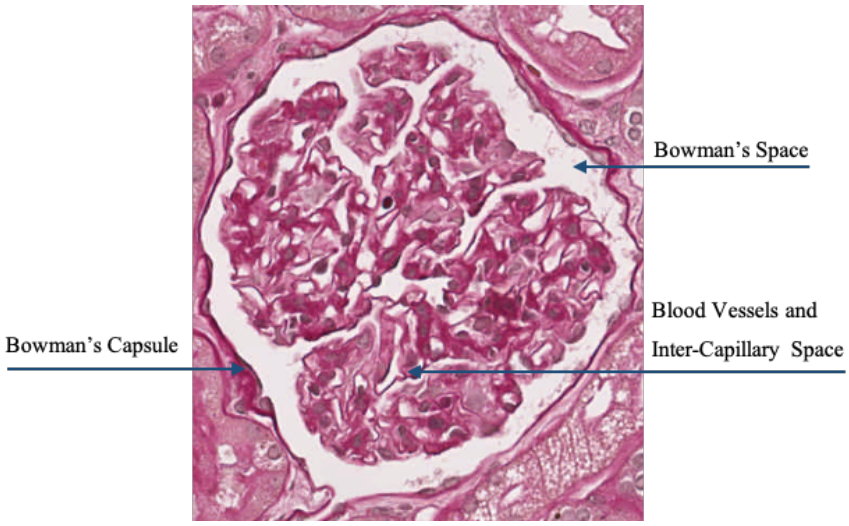


Fig. 2.33 Example of non-sclerotic glomerulus with the annotations of the main distinctive characteristics.

CAD systems for chronic kidney disease The traditional evaluation of glomerulosclerosis (and biopsies generally) is based on the visual analysis by trained pathologists of biopsy slides using a light microscope that is usually a time-consuming, prone to error and subjective

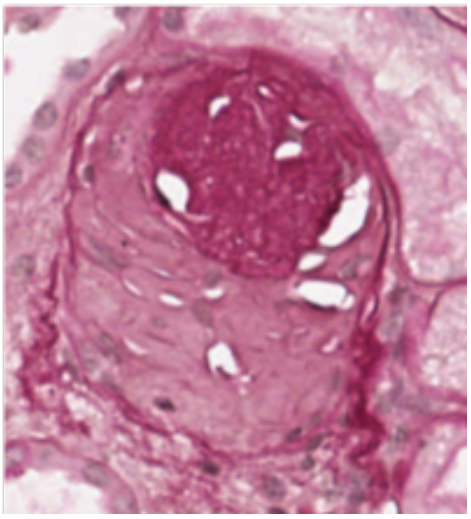


Fig. 2.34 Example of sclerotic glomerulus.

procedure; in particular, the glomeruli wide intensity variation and inconsistency in terms of shape and size increase the task complexity, while the high variability between the observers result in poor reproducibility among pathologists, which may cause an inappropriate organ discard. Therefore, the development of new techniques able to objectively and rapidly interpret donor kidney biopsy to support pathologist's decision making is strongly fostered. The increasing availability of whole-slide scanners, which facilitate the digitization of histopathological tissue, generated a strong demand for the development of Computer-Aided Diagnosis (CAD) systems; as for others medical imaging fields the spread of deep learning techniques and frameworks has led to a revolution in the CAD development.

In the realm of digital pathology, several studies propose CAD systems for glomerulus identification and classification in renal biopsies [161, 180, 181, 198, 200, 210, 227–234].

Traditional CAD systems for chronic kidney disease CAD systems based on image processing approaches aim to extract meaningful features mainly based on shape and texture analysis; the features are then usually used to feed machine learning techniques such as classical Artificial Neural Networks (ANNs).

Simon *et al.* proposed a texture-based features set as a simple but effective automatic method for glomeruli localisation [233]. The authors applied the algorithm on renal tissue sections and biopsies of large histopathological WSIs. The features extracted from an adaptation of the Local Binary Pattern (LBP) algorithm were used to train a Support Vector Machine (SVM) model. The authors reported high precision ($> 90\%$) and reasonable recall ($> 70\%$) as results.

To perform a comprehensive detection of glomeruli in images of whole kidney sections, Kato *et al.* proposed a new descriptor called Segmental HOG (Histogram of Oriented Gradients) [232]. The authors claimed the robustness of the solution and high-quality segmentation outputs; furthermore, the authors compared Segmental HOG with Rectangular HOG showing that the first approach reached significant improvements in detection performance.

Several authors focus on the analysis of glomeruli's shape and colour. Kotyk *et al.* proposed a novel solution to face the wide intensity variation and the inconsistency in terms of shape and size of the glomeruli in the renal corpuscle [235]. The proposed approach, based on Particles Analyzer technique, allowed the detection of the renal corpuscle and the following measurement of glomerulus diameter and Bowman's space width. The solution was tested on rats renal corpuscle images acquired using a light microscope. The authors reported that the proposed solution detected efficiently renal corpuscles, even with glomeruli deformation, such as split of the Bowman space due to glomerular hypertrophy, and accurately measures the glomerulus diameter and Bowman's space width. Furthermore, the authors reported a significant difference between the control and patients groups in terms of glomerulus diameter. An analysis of the effects of significant diversity of colour and tissue shape on whole slide images was performed by Zhao *et al.* [236], and focuses on two main study cases: automatic detection of interstitial inflammation and tubular cast. The authors focused on the extraction of Bowman's capsule width to design an automated glomerulus extraction framework from the micrograph of the entire renal tissue. The system was tested on non-human primates renal tissues with Haematoxylin and Eosin (H&E) staining. Samsi *et al.* focused on the kidney biopsies image analysis for quantifying tissue characteristics relevant to Lupus [237]. The proposed approach is based on colour segmentation to identify the Bowman's space around the glomeruli, followed by a grouping procedure based on size, location and orientation. The authors tested the developed algorithms on H&E stained mouse renal biopsies. Zheng *et al.* proposed a framework based on convolutional neural network for nucleus-guided feature extraction of breast histopathological images [238]. The detected nuclei were used to train a convolutional neural network, then used as image-level features extractor. The reported results show that the extracted features represent histopathological images and the classification framework overcome the state-of-the-art methods.

Deep learning CAD systems for chronic kidney disease A complete different analysis work-flow constitute the main core of deep learning CAD application on the chronic kidney disease problem. Bukowy *et al.* developed a convolutional neural network to detect glomeruli in trichrome-stained kidney sections. The procedure was tested on rat kidneys and the re-

ported results, regarding the classification of healthy and damaged glomeruli, show average precision and recall of 96.94% and 96.79%, respectively [239]. Ledbetter *et al.* proposed a Convolutional Neural Network to predict kidney function (evaluated as the quantity of primary filtrate that passes from the blood through the glomeruli per minute) in chronic kidney disease patients from whole slide images of their kidney biopsies [200]. Gallego *et al.* proposed a method based on the pretrained AlexNet model [103] to perform glomerulus classification and detection in kidney tissue segments [210]. Gadermayr *et al.* focused on the segmentation of the glomeruli. The authors proposed two different CNN cascades for segmentation applications with sparse objects. They applied these approaches to the glomerulus segmentation task and compared them with conventional fully-convolutional networks, coming to the conclusion that cascade networks can be a powerful tool for segmenting renal glomeruli [229]. Temerinac-Ott *et al.* compared the performances between a CNN classifier and a support-vector machines (SVM) classifier which exploits features extracted by histogram of oriented gradients (HOG) [240] for the task of glomeruli detection in WSIs with multiple stains, using a sliding window approach. The obtained results showed that the CNN method outperformed the HOG and SVM classifier [228]. Kawazoe *et al.* faced the task of glomeruli detection in multistained human kidney biopsy slides by using a Deep Learning approach based on Faster R-CNN [198]. Marsh *et al.* developed a deep learning model that recognizes and classifies sclerotic and non-sclerotic glomeruli in whole-slide images of frozen donor kidney biopsies. They used a Fully Convolutional Network (FCN) followed by a blob-detection algorithm [241], based on Laplacian-of-Gaussian, to post-process the FCN probability maps into object detection predictions [231]. Ginley *et al.* proposed a CAD to classify renal biopsies of patients with diabetic nephropathy [230], using a combination of classical image processing and novel machine learning techniques. Hermsen *et al.* adopted ensemble of five U-Nets, for the segmentation of ten tissue classes from WSIs of periodic acid-Schiff (PAS) stained kidney transplant biopsies [242].

In Section 3.1 will be presented and discussed different deep learning pipelines for the glomeruloscerosis evaluation. The problem has been investigated from classification, object detection, semantic and instance segmentation point of view; furthermore a preliminary insight about classical image processing algorithm has been conducted.

2.4.2 Radiology

Several works have been conducted in the field of radiological imaging in order to support clinicians with diagnosis and staging of different pathologies, such as tumours. Most diffused

areas of interest, due to the wide diffusion of tumours case, are breast, liver and kidney[243]. The high incidence of breast cancer in women requires an accurate assessment of breast glands with imaging techniques. In this field, mammography still represents the gold standard for screening purposes. Risky subjects, instead, undergo more invasive, but more sensitive and specific, imaging analysis, such as MR, CT and Digital Breast Tomosynthesis (DBT). Although over the years the detection, classification and segmentation of breast lesions have been accomplished with traditional image processing techniques and ML algorithms for supporting decisions, there are several works in the literature dealing with the characterisation of breast lesions using different imaging systems and innovative DL strategies [244–246]. Regarding the classification of breast lesions from imaging acquisitions, a comparison between a traditional approach based on the classification of hand-crafted features, and a classification pipeline based on the classification by a CNN has been conducted by Bevilacqua *et al.* [247]. In this work, several CNN architectures were used as feature extractors to allow the classification by a subsequent classifier. In the traditional approach, instead, morphological and textural features are usually evaluated by ANNs.

As for breast cancer, liver carcinoma shows extremely high mortality worldwide [243]. The segmentation of liver is crucial for several clinical procedures, including radiotherapy, volume measurement and computer-assisted surgery. Among hepatic tumours, Hepatocellular carcinoma (HCC) is the most common type of primary liver cancer in adults and is the most common cause of death in people with cirrhosis [248]. As for breast cancer, several works have been recently proposed introducing DL approaches for the segmentation of liver or hepatic tumours, or staging carcinoma [249–252]. As an example, a comparison between two approaches for staging HCCs has been conducted by Brunetti *et al.* [10]. In this work, the authors compare an approach classifying hand-crafted features from segmented ROI of HCCs with a second approach performing the staging of hepatic lesions with a CNN. In particular, the Google Inception v3 [253] was retrained with the transfer learning approach to perform such classification.

The study case about Autosomal Dominant Polycystic Kidney Disease faced in this thesis will be deeply discussed in the following section.

2.4.2.1 Autosomal Dominant Polycystic Kidney Disease

Autosomal Dominant Polycystic Kidney Disease (ADPKD) is a hereditary disease characterised by the onset of renal cysts that lead to a progressive increase of the Total Kidney Volume (TKV). Specifically, ADPKD is a genetic disorder in which the renal tubules become structurally abnormal, resulting in the development and growth of multiple cysts within the

kidney parenchyma [254]. There are two different gene mutations involved in the disease. The ADPKD type I, caused by the PKD1 gene mutation, involves the 85 - 90% of the cases, usually affecting people older than 30; the mutation of the PKD2 gene, instead, leads to ADPKD type II (affecting the 10 - 15% of the cases), and mostly regards children developing cysts already when in the maternal uterus. The PKD1 and PKD2 mutations have the same clinical characteristics, even though the latter one is associated with a milder clinical phenotype and a later onset of End-Stage Kidney Disease (ESKD). In all the cases, the size of cysts is extremely variable, ranging from some millimetres to 4 - 5 centimetres [255].

Currently, there is not a specific cure for ADPKD and the Total Kidney Volume (TKV) estimation over time is used to monitor the disease progression. Tolvaptan has been reported to slow the rate of cysts enlargement and the following progressive kidney function decline towards End-Stage Kidney Disease (ESKD) [256, 257]. Since all the actual pharmacological treatments aim at slowing the growth of the cysts, the design of a non-invasive and accurate assessment of the renal volume is of fundamental importance for the estimation and the assessment of the ADPKD progression over time.

Literature reports several methods performing the TKV estimation and trying to correlate non-invasive evaluated metrics with body surface and area measurements [258, 259]. Traditional methodologies are based on imaging, such as Computed Tomography (CT) and Magnetic Resonance (MR), and include stereology and manual segmentation [260, 261]. In detail, stereology is the superimposition of a square grid, with a defined cell positions and spacing, on each slice of a volumetric acquisition (CT or MR). The area of all the cells containing parts of the kidneys of each slice, combined with the information of the acquisition thickness, allows the computation of the final volume. Manual segmentation, instead, requires the manual contouring of the kidney regions for each slice. Several tools supporting this task have been developed, most including digital freehand contouring tools or interactive segmentation systems to assist the clinicians while delineating the region of interest. The main drawback of manual or near-manual segmentation is the subjectivity of the human operator performing the task; in particular, the phenotype, the presence of co-morbidities and cysts in neighbouring organs make difficult to achieve an accurate and objective assessment of the TKV, requiring also experience and training. To overcome this limitation and to standardise the segmentation methodology, several approaches for the semi-automatic kidney segmentation and TKV estimation, such as the mid-slice or the ellipsoid methods, have been investigated [262–264]. Although these methodologies are faster and more compliant than the manual ones, these are far from being accurate enough to be used in clinical protocols [265, 266].

In recent years, innovative approaches based on deep learning strategies have been introduced for the classification and segmentation of images. In details, deep architectures, such as deep neural networks or convolutional neural networks, allowed to perform image classification tasks, detection of Regions Of Interest or semantic segmentation [10, 267–269], reaching higher performance than traditional approaches [247].

Regarding the type of source images, recent studies about imaging acquisitions for assessing kidneys growth suggested that MR should be preferred to other imaging techniques [270]. Different research works allowed the estimation of TKV starting from CT images thanks to the higher availability of the acquisition devices and the more accurate and reliable measurement of TKV and cysts volume. On the other side, CT protocols for ADPKD are always contrast-enhanced using a contrast medium harmful for the health of the patient under examination; also, CT exposes the patients to ionising radiations. On these premises, the automatic or semi-automatic segmentation of images from MR acquisitions for improving the TKV estimation capabilities should be further investigated for improving the state-of-the-art performances.

In Section 3.2 will be presented and discussed two different approaches based on deep learning architectures for the automatic segmentation of polycystic kidneys starting from MR acquisitions. These two approaches, based on object detection and semantic segmentation, lead to a fully-automated pipeline for the detection and the segmentation of areas containing kidney in MR images.

2.4.2.2 Organs Segmentation

The volume quantification of organs is of fundamental importance in the clinical field, for diagnosing pathologies and monitoring their progression over time. Imaging techniques offer fast and accurate methods for performing this task in a non-invasive way. In fact, starting from volumetric imaging acquisitions, such as computed tomography or magnetic resonance, it is possible to perform the three-dimensional segmentation of organs, thus obtaining their volumetric information. This task, however, revealed to be time-consuming, requiring expert medical doctors to manually label each bi-dimensional slice in the volumetric acquisition. This procedure was also susceptible to differences inter- and intra-operator [271]. Taking into account these premises, researchers made considerable efforts in developing semi-automatic or automatic segmentation methods, especially for those body areas containing organs whose morphology could physiologically vary over time, or due to pathologies, such as kidneys or liver.

Traditional approaches for liver and spleen segmentation. Traditional image processing approaches for biomedical image segmentation involve thresholding algorithms, i.e. methods that allow the segmentation of regions of interest using different thresholds that can be found through several techniques, and Region Growing (RG) segmentation techniques, i.e. methods that start building the segmentation mask from a set of initial points, called seeds, then adding neighbouring points which are compliant to some homogeneity criteria.

In literature, there is a realm of region growing algorithms suited for the liver segmentation task, including approaches for both 2D and 3D segmentation. Choosing the right inclusion criteria for neighbouring pixels is crucial for obtaining acceptable segmentation performances. Most of the algorithms focus on the definition of proper homogeneity criteria. Gambino *et al.* presented an automatic 3D segmentation method that provides the seed's choice by minimizing an objective function and a homogeneity criterion based on Euclidean distance between the texture features associated to the voxels [272]. Other authors used an homogeneity criterion based on the difference between the already segmented area intensity and the pixel gradient [273, 274] or the pixel intensity [275–277]. Other approaches based on homogeneity criteria can be found in [278, 279]. Other authors focused on the pre-processing phase. In the attempt to resolve the region isolation problem in region growing algorithm, Lakshmipriya *et al.* employed a Non Sub-sampled Contourlet Transform (NSCT) to enhance the liver's edges and a bidirectional region growing algorithm [280]. Rafiei *et al.* focused on the pre-processing phase by using a contrast stretch algorithm and an atlas intensity distribution to create voxel probability maps [281]. Zhou *et al.* developed a semi-automatic liver segmentation algorithm exploiting intensity separation, region growing techniques and the morphological hole-filling operator to refine the segmentation results [282]. Other pre-processing methods can be found in [283, 284]. Elmorsy *et al.* focused on the post-processing phase, where they took advantage of an entropy filter to choose the best structural element to use [285]. Czipczer *et al.* combined a region growing algorithm and the results of a Convolutional Neural Network (CNN) model. To find the seed, they exploited an active contour model, whereas the masks generated by the CNN are refined through the GrowCut algorithm [286]. These two masks are then fused with a logical AND operation to obtain the final mask [287]. Bevilacqua *et al.* applied a pre-preprocessing consisting of contrast enhancement and cropping, followed by local thresholding, extraction of the largest connected component and morphological operators, before of propagating the obtained mask in both directions to reach all the slices [288]. Other approaches including image processing procedures both before and after the region growing could be found in [289, 290].

Different approaches have been tried in literature for spleen segmentation. Mihaylova *et al.* used the Chan-Vese active contour model to realize an automatic segmentation algorithm. To select the initial contour, they used template matching. After having segmented the area in the first slice, they used this area as initial contour for the following slice [291]. Subsequently, Mihaylova *et al.* presented a multi-stage approach based on segmentation methods, such as active contours without edges and K-means clustering. To define the initial contour, they created two atlas models [292]. Behrad *et al.* combined the features extracted from a neural network with those extracted from the image partitioned with a watershed transform. This process is repeated, varying the parameters of the segmentation algorithm, until the error between the features extracted from the two approaches is very small [293]. Jiang *et al.* used the ISO algorithm to segment the spleen. This algorithm can cause over-segmentation, so they used Principal Component Analysis algorithm to eliminate the muscles that can cause misclassification [294]. Gauriau *et al.* adopted a multi-object template deformation framework to segment liver, kidneys, spleen and gallbladder, and used a random forest regression algorithm with shape priors to obtain confidence maps for each organ. The shape of each organ is then included in an energy optimization technique that also considers image-derived forces [295]. Reza Soroushmehr *et al.* proposed a slice-wise segmentation algorithm based on the position of ribs and vertebrae to determine the ROI of the spleen. After determining the area in the first slice, its centroid is used as initial spleen location for the following slice [296].

Deep learning approaches for liver and liver vessels segmentation. Recent works investigated the use of convolutional neural networks, and deep learning strategies, in order to design and implement automatic clinical decision support systems based on medical images. Medical decision support systems also include the segmentation of organs from volumetric imaging acquisitions [104, 297, 298]. For example, De Vos *et al.* made an effort for localizing anatomical structures in 3D medical images using CNNs, with the purpose to ease tasks such as image analysis and segmentation [299]. Regarding the abdominal area, there is a growing interest in CT scan analysis for diagnosis purposes and therapy planning. In fact, the segmentation of organs is crucial for several clinical procedures, including radiotherapy, volume measurement and computer-assisted surgery [300]. Rafiei *et al.* developed a 3D to 2D Fully Convolutional Network (3D-2D-FCN) to perform automatic liver segmentation for accelerating the detection of trauma areas in emergencies [301]. Lu *et al.* developed and validated an automatic approach integrating multi-dimensional features into graph cut refinement for the liver segmentation task [300]. Kim *et al.* proposed a 3D patch-based U-Net,

followed by a graph-cut post-processing, for the multi-organ segmentation of liver, stomach, duodenum and right/left kidneys. The authors employed a Multi-Class Cross-Entropy loss function with six classes. Their proposed method outperforms atlas-based approaches, whilst shows comparable results with the inter-observer delineations of organs [302].

A further application of the segmentation task applied to liver regards the vessels identification. According to Couinaud model [303], hepatic vessels represent the anatomic borders of hepatic segments and, consequently, segmentectomies, based on the precise identifications of these vascular landmarks, are crucial in the modern hepatic surgery because they avoid unnecessary removal of healthy liver parenchyma and reduce the complications of most extensive resections [304]. In literature, there are different approaches for liver vessels segmentation. Oliveira *et al.* proposed a segmentation method exploiting a region-based approach [305], where a gaussian mixture model was used to identify the threshold to be selected for adequately separating parenchyma from hepatic veins. Yang *et al.* proposed a semi-automatic method for vessels extraction. They used a connected-threshold region-growing method from the ITK library [306] to initially segment the veins. To find the threshold, they exploited the histogram of the masked liver. However, this process has to be supervised by an expert user through a graphic interface [307]. Goceri *et al.* proposed a method called Adaptive Vein Segmentation (AVS): they exploited K-means clustering for initial mask generation; then they applied post-processing procedures for mask refinement, followed by morphological operations to reconstruct the vessels [308]. Chi *et al.* used a context-based voting algorithm to conduct a full vessels segmentation and recognition of multiple vasculatures. Their approach considered context information of voxels related to vessels intensity, saliency, direction, and connectivity [309]. Zeng *et al.* , instead, proposed a liver-vessels segmentation and identification approach, based on the combination of oriented flux symmetry and graph cuts [310].

In Section 3.3 will be presented and discussed two different approaches based on classic image processing algorithms and deep learning architectures for the semi- and full-automatic spleen, liver and liver vessel segmentation.

2.4.3 Electromyographic signals analysis

The arguments discussed in this section are more speculative and not directly aimed at developing CAD systems; in this section will be discussed the application of autoencoders, presented in Sections 2.2.1.2, to signal processing and in particular to surface electromyography analysis. The obtained results are promising if compared to the ones obtainable from

standards approaches proposed by literature, and make this study case worth to be presented and discussed in this thesis. A brief literature review about sEMG signals and the use of machine learning and deep learning as processing techniques will be following reported [114, 115], whilst the study case will be presented in Section 2.4.3.2 and Chapter 4.

In many medical fields surface electromyography (sEMG) is frequently used [311], such as neurophysiology [312], ergonomics and occupational medicine [313], posture analysis [314], movement and gait analysis [315], EMG-based biofeedback [316], exercise physiology and sports [317], as well as human-machine interaction/interfaces (HMI) [318–320]. In detail, electromyographic signals (EMG) are biomedical signals that provide representations of the electrical potential fields produced by the membrane depolarization of the outest muscle fibers. An EMG signal corresponds to a train of motor unit action potential (MUAPs) showing each muscle response to neural stimulation and presents a random behavior. Shapes and firing rates of MUAP in EMG signals revealed to be important source of information that can be used in several applications. In particular, EMG signal detection requires the use of intramuscular or surface electrodes positioned at a certain distance from sources, i.e., muscle fibers. Moreover, EMG detectors, especially surface electrodes, collect signals from different motor units at the same time thus leading to an interaction of different signals. Just before the amplification, the amplitude range of the EMG signal is ± 5 mV and is affected by several types of noise: inherent noise in electronics equipment, ambient noise and motion artifacts due to the movement of both electrode interfaces and cables. There are also other factors affecting the EMG signals, besides noise [321]: a) electrode structures and placements, b) physiological, anatomical, biochemical characteristics of muscles fibers and the amount and type of tissues between muscle surfaces and electrodes, and c) crosstalk from nearby muscles. All such factors strongly affect the characteristics of collected signals (e.g. signal amplitudes and frequency contents), thus explaining the intra-subject / inter-subject variability that may be observed when acquiring EMG signals. Clinical and human machine interaction applications of EMG signals should clearly rely on reliability and repeatability of used techniques. The repeatability of sEMG measurements has been tested by many researchers, and a critical issue concerning features of single-channel sEMG signals, obtained in different tests and days, concerns with the repeatability of electrode positions and inter-electrode distances [322–325]. It can be observed that the reproducibility of estimates of the sEMG characteristics under isometric or dynamic conditions is generally not excellent. This observation is partially caused by a persisting lack of standards in this field. A substantial contribution to this issue has been received by some researchers' effort in defining the standard procedures to follow for acquiring EMG signals [326], starting from the use of electrodes grids and the

automatic identification of regions of interest. Besides electrode design, many researchers also started to investigate new signal processing techniques that could robustly decipher all key information encoded in the EMG signals; some of these techniques are based on artificial intelligence [327].

Researchers interested in myoelectric man-machine interfaces have long been aware of the high potential of the AI in EMG signal (pre-)processing [328, 329] and myoelectric prosthesis represent one of the most worthy application examples. In fact, thanks to the direct association between the action potential produced by the motor neurons and the electrical activity induced in the innervated muscle fibers, muscles cab be considered as biological amplifiers of efferent nerve activity in applications of man-machine interfacing. Nowadays, data-driven approaches, which are mainly based on machine learning techniques, represent the most used solution for implementing the mapping between EMG signals and the device to be driven [320, 330–332] after the model-driven approaches that use EMG signals as the input to specific physical models of the musculotendon system [318, 333–337].

Over the past decades the weakness of the myoelectric prosthetic hands offered by industry and the need to develop more intuitive and efficacious EMG-based human-prosthesis interfaces, have clearly boosted the research on data-driven approaches [319]. Therefore, several researchers started focusing on developing new machine learning-based methods for detecting the intended hand gesture from forearm muscle activations and better controlling prosthetic devices [338]. ML techniques have demonstrated to be valid in several others domains where the quantitative EMG (or QEMG) plays an important role [339]. As an example, ML approaches have been widely used to develop intelligent systems for supporting clinicians in diagnosing and staging diseases that affects the human motor system, such as myopathy, neuropathy, amyotrophic lateral sclerosis, Parkinson's disease [86, 339–347].

In biomedical applications, as well as in other contexts, the increasing amount of multi-modal physiological information, together with an increased problem complexity and all subsequent difficulties concerning the extraction of meaningful hand-crafted and domain-dependent features, limit the power of the traditional shallow machine learning approaches, despite the considerable research works carried out for optimising performance of available classifiers [82, 348–355]. Deep learning overcomes these limitations allowing an increasing transformation of data into a more abstract representation.

Deep learning is actually a growing breakthrough technology in data analysis [10, 105, 119, 297, 356–361] and, as already stated, it is becoming as the leading ML approach both in general image processing and computer vision domains [120, 362–369]. Moreover, promising results emerge from deep learning networks in various medical fields, since deep

learning can be intended as an improvement of artificial neural networks, based on more layers that enable higher abstraction levels and better predictions from data [247, 370].

Researchers have therefore begun to investigate the ability of DL to process and decode sEMG data also thanks to the recent launch of several EMG recording benchmark databases, e.g. NinaPro [371], BioPatRec [372], CapgMyo [373], UCI Database [374], CSL-HDEMG [375], PhysioNet [376] and MASS-DB [377]. Such a growing trend in science also includes many other physiological signals, for instance electroencephalogram (EEG), electrocardiogram (ECG), and electrooculogram (EOG), as discussed in two recent surveys [114, 378, 379].

2.4.3.1 From Raw EMG Signals to Deep Network Input

The use of different EMG signal acquisition setups, i.e. sparse and array electrodes, combined with the possibility to employ several kind of deep learning techniques enables many solutions for designing network inputs that are computed from the raw EMG signals. The first pre-processing step commonly consists in filtering any EMG signal with a digital high-pass/ band-pass filter to remove the low-frequency artifacts, e.g. movement artifact and baseline noise contamination [380, 381]. A valid alternative regards the application of the wavelet transform (WT) that is used to reconstruct the original signal with signal components without noise information [382]. The power line interference has also to be removed, and if it is not done by the EMG amplifier a specific digital filter has to be used, such as the spectral Hampel filter [383].

The pre-processing step is followed by a signal segmentation procedure that aims at extracting several portions of EMG signals using a time-windows. All information encoded within the time windows of every considered EMG signals will be then used to construct a specific example used to train, validate or test an ad-hoc deep network. This means that the employed network will provide as output for each time window, a vector of class probabilities for a classification problem or a set of estimated variables in case of regression problems. The time-window length is a crucial parameter to be subsequently set, a large window contains more temporal information but at the same time causes a delay between the event to be detected and the related network output. On the other hand, a small window is adequate for real-time applications but considers a limited amount of information. The analysis of literature shown a large variability of the window length, typical time-window length values are 30ms, 50ms, 100ms, 150ms, 200ms and 300ms. The time-window length has a particular relevance when the deep network is included in closed loop myoelectric controllers where the input latency is a fundamental factor to consider. As an example, a maximum time delay

of 300ms can be acceptable when controlling of prostheses through EMG signals [384]. Sliding time-window are usually extracted considering an overlap (also called increment) that is defined as a percentage of window length. Another important factor that is considered with the window length choice, is the sampling frequency of EMG signals [385]. It is also important notice that many authors choose a time-window length value that is equal to the value used by the authors of already published works thus allowing a robust comparison of the model performance. In fact, the possibility of comparing results plays an important role when working with public datasets. However, some authors decided not to consider the windowing approach and prefer to provide EMG signals at each sampling instant as input to the network [386]; such approach could be especially followed when dealing with high-density EMG arrays since they can provide a lot of activation data at a single time sample.

After all EMG signals have been segmented with overlapped time-windows, the process to build network inputs is clearly highly dependent on the preferred deep network architecture. Existing approaches can be mainly divided in two main groups according to the fact that the selected architecture (or its first module in case of mixed architectures, e.g. CNN-RNN) is either a CNN or a DNN, an AE, a DBN or a RNN. When dealing with DNNs, AEs, DBNs and RNNs networks input have vector-like shapes. In most of the cases, such vectors are composed of the values of hand-crafted features extracted by the window segment of each acquired EMG signal. The revised literature considered both the time-domain features and frequency-domain ones that are usually used in EMG processing [387].

A different approach has to be followed when a convolutional neural network is considered for EMG signal processing. CNNs that are employed for traditional image processing usually take as input either an $M \times N$ or $M \times N \times 3$ size array for gray or RGB images, respectively. As a consequence, information contained within EMG signals have to be arranged in a 2D or 3D-dimensional array. More in details, the specific approach to follow depends on the used setup that can consider the acquisition of either high-density sEMG signals [373, 375] or sparse multi-channel sEMG ones [388]. A solution can intuitively consider to arrange EMG signals in a sEMG-image where each electrode can be regarded as a pixel of the image. Such solution is certainly valid when using high-density sEMG signals that are collected by a grid of sEMG electrodes [386, 389]. Then in this case the size of the sEMG image will be equal to the electrode array size. However, when considering sparse multi-channel sEMG signals, the number of electrodes is limited and their placement is sparse, thus the above presented approach cannot be directly implemented. Among all multiple solutions that have been adopted to deal with sparse electrodes, the main two used techniques consider an $N \times L$

dimension matrix where N is the number of electrodes and L is the time-window length, or a matrix built assembling the spectrogram of each EMG signal computed on the time-window.

2.4.3.2 Muscle Synergy Extraction

In the last decade, electromyographic (EMG) signals have been widely used as human-machine interface for advanced devices [330]. This is particularly relevant in clinical environments, where the adoption of this devices can provide new capabilities in neurorehabilitation techniques [390–392]. In this context, myocontrol grabbed the attention of many researchers, for its potential utility in the motion intention detection and prostheses control domains [330, 393, 394] as well as teleoperation systems [395, 396], without invasive signal acquisitions. Although many different strategies have been developed in the last decades, such as direct control models [397], neuromusculoskeletal models [333, 334, 398–401] and multiple linear regression models [336, 402], some obstacles still represent a gap between prototypes and commercial devices. The lack of simultaneous control and the high computational cost of some complex models [403] do head off a truly natural human–machine interaction. Pursuing this purpose, bio-inspired models suggested a way to preserve a simple and intuitive interface, as done in many other fields, through the exploitation of complex muscle activation patterns called muscle synergies. This biological model comes from the hypothesis that the central nervous system decodes high level user's intent and drives, concurrently, a subset of motor primitives instead of independently activating muscles. Although the neurological origin of synergies has been supposed [404–406], this theory still remains unproven and many issues are still unresolved [407].

Several works assessed how muscle synergies could be useful to detect abnormal co-contraction patterns, e.g. due to stroke [408, 409] or spinal cord injury [410] or case history evolution throughout limbs rehabilitation [411, 412]. Such evidence suggests that therapies could be improved gathering information from muscle activation patterns. Moreover, the acquisition of new motor skills leads to either the development of new activation patterns or changes in the components of existing ones [413, 414]. From the application point of view, by combining those patterns, it would be possible to achieve different, even complex, motor tasks [415] also granting more robustness than other control strategies. Previous studies showed how synergistic models are useful for achieving either multi-DoF planar movements with the upper-limb [416] and wrist [417] or upper-limb force prediction during isometric contractions [336]. In the current state-of-art, many algorithms have been used for time-invariant synergies extraction [418], starting from surface EMG (sEMG) data, such as principal component analysis, independent component analysis, linear discriminant analysis,

clustering (e.g. k-means) and Non-Negative Matrix Factorization (NNMF). The latter gained particular attention because of its intrinsic capabilities of extracting non-negative features that match the positive physical constraint of muscles activations [419]. These results have encouraged EMG researchers to replicate synergies behavior as a bio-inspired dimensionality reduction method [420] based normally on the NNMF algorithm [419]. The use of synergies have been demonstrated to outperform traditional linear regression methods [421] and may present more accurate results in scenarios where patterns, not contained in the training set, appear during the validation phase [422, 423]. Hence, EMG-driven synergy-based controllers are revealing a promising approach for achieving a natural interaction between human and assistive/medical robots [335, 402, 424]. Nevertheless some of the limitations that traditional approaches face when dealing with sEMG, such as cross-talk, electrodes positioning, fatigue and many others, are magnified whenever synergies perform dimensionality reduction [387, 425].

In this scenario, undercomplete autoencoders have been investigated for this thesis as a new computationally efficient method for bio-signal processing and, consequently, synergies extraction. Furthermore, a modified autoencoder has been designed as attempt to develop a muscle synergy-based myo-controller that is tailored to the specific subject by simultaneously considering the synergy extraction and the mapping between the synergy activations and the variables used in the task space (forces or moments). The compressed input representation and the motion intent estimation have been compared to the outcomes of the most used algorithms described below, evaluating both the signal total variance reconstruction rate and the prediction/classification performance; the complete study is reported in Chapter 4.

Non-Negative Matrix Factorization. Given a matrix M, the NNMF is a factorization algorithm able to compute the two matrices W and C such that:

$$M \approx W \cdot C, \tag{2.40}$$

with the property that all three matrices have no negative elements. Considering muscle synergy extraction, M is a matrix $N \times P$ containing the muscles activation observations during a task, where N is the number of recorded muscles and P the number of total time samples; W is the synergy matrix having size equal to $N \times Q$, where Q is the number of extracted synergies; C is the matrix that contains the synergy activation signals and has dimension equal to $S \times P$. Given this nomenclature, $m(t)$ and $c(t)$ indicate a single column (that corresponds to a single sample time) of M and C, respectively. Once the synergy model has been defined, that is, the synergy matrix W has been computed, the synergy activation

vector $c(t)$ related to EMG signals vector $m(t)$ non-considered for the W definition can be computed as follows [402]:

$$c(t) = W^{+} \cdot m(t), \tag{2.41}$$

where W^{+} is the pseudo-inverse matrix of W.

Joint moment estimation based on muscle synergies. A continuous EMG-based movement intention detector module has to be able to take the processed EMG signals of the involved muscles and compute an estimation of both the direction and amplitude of the movement. Considering a robotic interface controlled with an admittance control, such estimation has to be a force/torque vector. For example, an upper limb wearable exoskeleton could assist the patient movement if it moves in the same direction as the direction the patient arm is applying the force.

Under certain conditions [402], the EMG-based force/moment estimation can be based on a linear combination of the processed EMG signals as follow:

$$T = H \cdot f_{EMG} \tag{2.42}$$

where T represents the vector of the force/moment components, f_{EMG} is the vector of the instantaneous EMG-based features and H is the matrix relating EMG features to force/moment estimated using multiple linear regressions of each applied force/moment component. If the movement is constrained on a plane, T is 2×1 vector, f_{EMG} is a $M \times 1$ vector, and H is a $2 \times M$ matrix, where M is the number of considered EMG-based features.

2.4.4 Movement Disorders

The term "movement disorders" refers to a group of nervous system (neurological) conditions that cause abnormal increased movements, which may be voluntary or involuntary. Movement disorders can also cause reduced or slow movements [426]. Among the several types of movement disorders, dystonia and Parkinson's disease are of interest for this thesis. Dystonia is defined as the condition involving sustained involuntary muscle contractions with twisting, repetitive movements; it may affect the entire body (generalized dystonia) or one part of the body (focal dystonia) [426]. Parkinson's disease, instead, is a slow progressive neurodegenerative disorder, and causes tremor, stiffness (rigidity), slow decreased movement (bradykinesia) or imbalance. It may also cause other non-movement symptoms [426].

The following two sections will provide general medical information and analyse the literature about Blepharospasm (a focal dystonia) evaluation and handwriting analysis as a methodology for Parkinson' disease assessment and grading.

2.4.4.1 Blepharospasm

Idiopathic blepharospasm (BSP) is an adult-onset focal dystonia characterised by bilateral, synchronous and symmetric dystonic spasms in the Orbicularis Oculi (OO) muscle [427–430]. Dystonic spasms can be phenomenologically heterogeneous, with either brief or prolonged spasms and a narrowing or closure of the eyelids [431]. In addition to spasms, patients affected by BSP might present a spectrum of additional signs/symptoms, including: sensory symptoms in the eyes that indicate ocular diseases (e.g. dry eye syndrome) [432], increased spontaneous blink rate [433], presence of sensory tricks to transiently improve eyelid spasms (stretching, massaging or touching the eyebrow, the eyelid, or the forehead), apraxia of eyelid opening [434] and dystonia in other body parts. Clinical evaluation of BSP severity poses a number of challenges, and several drawbacks might limit the widespread use of most of the existing severity scales. Since involuntary eye closure is the most disabling BSP feature, it would be reasonable to assume that objective measures of eye closure might be a good measure of BSP severity. Recently, algorithms based on the analysis of standard video recordings have been developed and try to measure the percentage of time the patients' eyes were closed while they were instructed to keep them open [435]. However, such a methodology allows only to measure the total time of eye closure without identifying the events contributing to the episodes of eye closure, that are, blinking, brief spasm with complete eye closure, or prolonged spasms with complete eye closure. The relevance of these phenomenological aspects in the evaluation of BSP severity is supported by recent evidence indicating that the type of spasm might identify BSP subtypes that are characterised by varying severity and the tendency of dystonia to spread to adjacent anatomic regions [427].

Peterson *et al.* [435] utilised the Computer Expression Recognition Toolbox (CERT) to compare clinical rating scales of blepharospasm severity with involuntary eye closures. The software performs a frame-by-frame video analysis to evaluate the eye closure time and compares the results obtained implementing three commonly used clinical rating scales: the Burke-Fahn-Marsden Dystonia Rating Scale [436], the Global Dystonia Rating Scale [437] and the Jankovic Rating Scale (JRS) [438]. The results demonstrated that CERT has convergent validity with conventional clinical rating scales and, hence, it can be used with video recordings to measure blepharospasm symptom severity automatically and objectively.

However, the previous severity scales are limited by a number of potential drawbacks. In particular, they comprehensively measure dystonia severity in all body parts, and severity grading is based on the intensity of dystonic contractions merged with [436] or weighted [437] by duration and daily frequency of the spasms. In other words, the main problem with these scales is that they adopt the same grading modality for dystonia at different body sites, each of which ideally requires a specific severity assessment system. To date, the only severity scale specifically developed to assess BSP severity is the Jankovic Rating Scale [438]. In fact, this scale includes two subscales that measure the intensity and frequency of eyelid spasms, both based on a 5-point grading system. However, this severity scale has not yet been validated in terms of reproducibility.

Recently, Defazio *et al.* presented a novel scale, called Blepharospasm Severity Rating Scale (BSRS), for rating BSP severity; the scale allows all the above-mentioned limitations to be overcome [431].

The visual recognition of facial movements and expressions in patients with blepharospasm is subtle and hardly perceptible, making the objectification of blepharospasm severity a difficult task and even expert neurologists can differently rate its severity in the same patients. To address these issues, software tools have been recently developed [435]. However, in order to properly characterise BSP severity, facial recognition systems must be utilised and they have to be designed not only to measure the eye closure time, but also to recognise facial expressions and movements typically related to events contributing to the eye closure episodes.

In a preliminary study conducted before this thesis research, a frame-by-frame video analysis was performed to measure the geometry of specific facial features [439]. The distances between particular reference points were measured, and investigations on how these distances changes in BSP-related facial expressions were carried out. Threshold values were then fixed for these distances to distinguish between pathological and non-pathological facial movements. However, the adopted approach showed important generalisation limitations: (i) the strategy for fixing the threshold values for the distance of facial landmarks cannot be generalised to all the patients, requiring the determination of specific threshold values for the distinction of pathological from non-pathologic conditions; (ii) the acquisition protocol was not standardised due to the difficulties to set proper environmental conditions that, in the clinical context, might change over time and from place to place. The last reason did not allow the standardisation of the acquisition system (e.g. the CCD camera with which patients are video-recorded), which is a requirement of crucial importance for adopting the strategy based on threshold values.

In Section 5.1 will be presented a CAD tool capable of overcoming the main limitations of the previous approach implementing a deep neural model with a topology optimised via genetic algorithms. The software, based on standard video-recordings from commonly available video cameras, is capable not only to measure the percentage time of eye closure, but also to recognise blinking, brief and prolonged spasms, which are the typical facial movements that take place in patients with blepharospasm. The proposed solution is a practical system very suited to the clinical context where the environmental conditions cannot be easily standardised; it is a promising tool for supporting and assisting physicians to rate blepharospasm severity according to the BSRS scale.

Blepharospasm Severity Rating Scale (BSRS). The blepharospasm severity rating scale includes six items, for each of which a score S must be assigned [427]. In detail:

Item A1 concerns the type of eyelid spasm occurring in the patient.

- If a brief spasm (duration < 3 s) with complete rim closure take place $\rightarrow$ score $S(A1) = 1$;
- if a prolonged spasm (duration $\geq$ s) with partial rim closure take place $\rightarrow$ score $S(A1) = 2$;
- if a prolonged spasm (duration ≥ 3 s) with complete rim closure take place $\rightarrow$ score $S(A1) = 3$.

Item A2 concerns the apraxia of eyelid opening.

- If apraxia is present $\rightarrow$ score $S(A2) = 2$;
- if apraxia is absent $\rightarrow$ score $(A2) = 0$.

Item A3 concerns the spasms occurring during the writing of a stereotyped sentence.

- If spasms of the orbicularis oculi occur $\rightarrow$ score $S(A3) = 1$;
- if spasms of the orbicularis oculi do not occur $\rightarrow$ score $S(A3) = 0$.

Item A4 concerns the average duration of the prolonged spasms.

- If the average duration is $3-4$ s $\rightarrow$ score $S(A4) = 1$;
- if the average duration is $4.1-5$ s $\rightarrow$ score $S(A4) = 2$;
- if the average duration is more than 5 s $\rightarrow$ score $S(A4) = 3$.

Item B1 regards the frequency of blinks and brief spasms.

- If 1 to 18 blinks and brief spasms take place per minute $\rightarrow$ score $S(B1) = 1$;
- if 19 to 32 blinks and brief spasms take place per minute $\rightarrow$ score $S(B1) = 2$;
- if more than 32 blinks and brief spasms take place per minute $\rightarrow$ score $S(B1) = 3$.

Item B2 regards the frequency of prolonged spasms.

- If 1 to 3 prolonged spasms take place per minute $\rightarrow$ score $S(B2) = 1$;
- if 3.1 to 7 prolonged spasms take place per minute $\rightarrow$ score $S(B2) = 2$;
- if more than 7 prolonged spasms take place per minute $\rightarrow$ score $S(B2) = 3$.

The total score is given by the sum:

$$SIn = S(A1) + S(A2) + S(A3) + S(A4) + S(B1) + S(B2). \tag{2.43}$$

2.4.4.2 Handwriting Analysis in Parkinson' Disease

Parkinson's Disease (PD) is the second most common neurodegenerative disorder after Alzheimer's disease, that leads to several neuro-motor deficits. It is well known that PD patients exhibit problems when they perform movements that are executed sequentially due to the loss of coordination among the motor sequence components. As a result, sequential movements are more segmented and characterized by pauses between sub-movements [440].

The motor coordination difficulty reflects also on handwriting which is a highly over-learned fine and complex manual skill, as well as one of the most common activities performed by people of all ages and in a variety of leisure and professional settings; handwriting involves an intricate blend of cognitive, sensory and perceptual-motor components [441]. Therefore, it is not surprising that abnormal handwriting is a well recognized manifestation of a neurodegenerative disease. The difficulties in the handwriting process affecting PD patients are mainly two:

- difficulties related to the control of the movement amplitude, e.g., decreasing the size of the characters (micrographia) and failing in keeping the stroke width of the characters constant as the writing progresses [442–449];
- not regular and bradykinetic movements that lead to an increased movement duration, decreased speed and accelerations, and unstable velocity and acceleration [450–455].

It follows that, if patients with PD are less able to increase speed during handwriting, movement amplitude should have a linear relationship to movement duration. Thus, in contrast to the peak acceleration in handwriting of age matched controls, which is governed

by the isochrony principle, peak acceleration in parkinsonian handwriting could be unrelated to writing size [456]. In fact, according to the isochrony principle, a strong tendency exists to keep the execution time of these complex trajectories, independent of the movement size [457].

Several research groups have investigated the use of computer-aided handwriting analysis tools, based on handwriting's features, to differentiate PD patients from healthy subjects. Helsper *et al.* published a study that investigated the handwriting differences between preclinical PD patients and healthy controls [458]. The authors analysed two lines of the handwritten text (sampled from a longer written text) and proposed an approach that considers (1) the extraction of 10 features from text segments written by test subjects as a first step, and then (2) the computation of a single resulting feature set based on the mean, the standard deviation and the frequency of the occurrences. The authors statistically proved the validity of their approach confirming that preclinical PD handwriting may demonstrate unique features many years before their diagnosis. Longstaff *et al.* studied the relation between the inclination of PD patients to scale the character size and reduce the speed of drawing movements and the movement variability [459]. The experiment is based on the analysis of several geometrical writing patterns with different shape and size drawn with a pen on a graphics tablet. By analysing the extracted features the authors stated that there is a substantial divergence in the quality of movements between PD patients and healthy people A different recording set-up has been used by Ünlü et al. [460] that recorded the pressure and the inclination of an electronic pen during writing tasks. Their proposed approach considered the extraction of 8 different features and the use ROC curves to analyse the diagnostic possibilities both in term of sensitivity and specificity. Their results showed that the most representative feature is based on the difference between the writing pressure and the tremor of the pen tilt angle; it seemed that for medicated patients, the tremor control was better achieved for movements (handwriting) than for constant pressure (pen tilt). The obtained area under the curve in the best case resulted to be equal to 0.963. Electronic pen and tablet have been also used by Rosenblum *et al.* to collect the position, the pressure and the angle of the pen tip during the writing of two main patterns (i.e., name and fixed address) [461]. The average values of the pressure and velocity acquired during the entire task and other spatial and temporal characteristics of each stroke, allowed them to differentiate PD patients from control subjects with a sensitivity of 95.0%. Finally, Drotar *et al.* proposed a methodology based on 11 features, extracted from kinematic handwriting analysis of different tasks, to build a predictive model of PD reaching an accuracy of 79.4%, and similar values of specificity and sensitivity [450].

Model-free handwriting analysis. The computer-aided model-free handwriting analysis is based on the extraction and classification of particular features starting from the assumption that the characteristics (or features) of one or more particular biometric signals/parameters can synthesize and represent a particular aspect of the user's handwriting. The analysis can be done both during the execution of the task (online) and after it is executed (offline), and usually requires the application of processing algorithms on signals and/or images that, starting from the pattern created, can succeed to extract features of interest. Then, these are usually used to train classifiers able to provide a separation between different subjects based on their clinical status.

The model-free techniques that will be discussed in Section 5.2 are based on both online and offline extraction of features, respectively during and after the handwriting task, as well as on feature reduction and classification algorithms. Furthermore, the analysis of literature suggested that an important aspect regarding the handwriting task is the type of the surface on which the person is writing on, since it can deeply affect the task itself. On smooth and slippery surfaces, such as that of a tablet or of the back of a credit card, people can have difficulty in writing and/or signing [462]. The sensation of sliding over a slippery surface suggests that the fine motor control required for adjusting pen movements can be disturbed [463].

To not introduce further variables in the proposed technique, the use of devices that could influence writing tasks, such as tablets, was avoided in a first study, and the handwritten text/drawing from normal paper sheets were scanned and analysed by exploiting vision-based features. A second study, instead, investigated advantages and drawbacks of features generable by using tablets as input device for Parkinson' disease assessment and grading. Both the researches make also use of dynamic features extracted from sEMG signals acquired at the arm level.

Chapter 3

Deep Learning for Medical Imaging

This chapter will focus on the application of deep learning techniques for the development of CAD system pipelines for medical imaging processing. Three study cases will be presented and where possible a traditional image processing pipeline has been proposed for comparison. Each case results have been compared with the literature.

Part of this chapter has been published in international conferences and journals [161, 175, 181, 227, 297, 298, 364, 464, 465], and the complete study case described in Section 3.1.2 is currently under review for journal publication [466].

3.1 Deep Learning in Pathology: Chronic Kidney Disease Study Cases

The first study case is the evaluation of the global glomerulosclerosis for the Chronic Kidney Disease presented in Section 2.4.1.2. The glomeruli evaluation has been conducted pursuing the four main machine learning goals described in Chapter 2: classification, object detection, semantic and instance segmentation. Furthermore, a classification pipeline based on classical image processing algorithms has been first and foremost investigated to have a preliminary insight of the problem.

3.1.1 Materials

All the glomeruli evaluation applications are based on the same dataset described below.

Data Description Whole Slide Images used in the following sections were collected between 07/2011 and 02/2015 by physicians from the Department of Emergency and Organ

Transplantations (DETO) of the Bari University Hospital (Italy). All the provided biopsies were PAS stained sections from formalin fixed paraffin embedded tissue. Slides were following digitized using the Aperio ScanScope CS high-resolution whole-slide scanner, a scanning objective with a $20\times$ magnification and a corresponding resolution of $0.50\mu m/pixel$.

The WSIs were collected from a total of 26 kidney digital biopsies coming from 19 donors and stored at full resolution in SVS file format (an Aperio file format consisting of pyramidal tiled TIFF with non-standard metadata and compression). Each WSI contains several biopsy sections, ranging from one to seven section per WSI. The whole dataset counts an average value of four biopsy section per WSI and a total amount of 105 sections. The collected images present wide differences in colour and saturation, even if all treated with PAS staining. Examples of saturation differences are reported in Fig. 3.1.

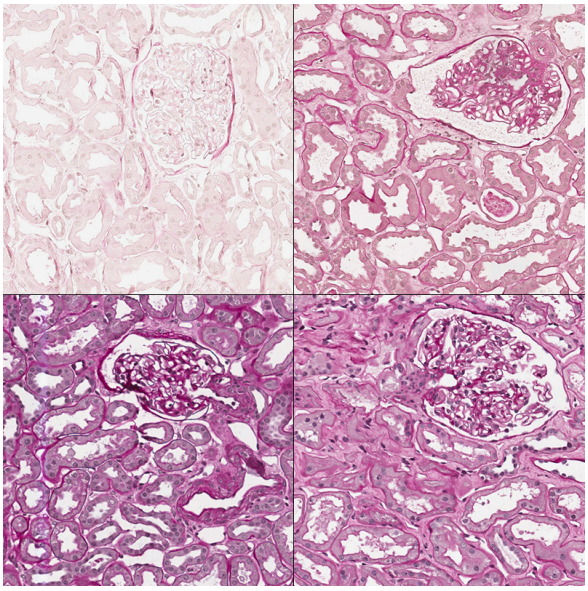


Fig. 3.1 Example of the variability introduced by saturation differences on PAS stained kidney biopsies.

Dataset Creation The techniques studied and reported in the following sections are all based on supervised learning, and require an annotated ground truth. Since in common practice, physicians use the aforementioned Aperio ImageScope software to visualize the WSIs, the annotation capability of the same has been used. The software allow to export image annotations through the XML file format, that is easy readable and parsable; furthermore, a compatible XML file can be generated by the proposed workflows, easily integrated in Aperio and used by clinician for outcome visualisation and analysis. This allows to complete a full CAD system with seamless integration with a user-friendly software for

clinicians. For the annotation, two medical graduands manually identified and labelled the glomeruli independently; subsequently, a renal pathologist validated the final annotations. The procedure was performed in the Aperio ImageScope tool, by outlining the real glomeruli region and labelling the glomerulus as sclerotic and non-sclerotic. All the annotated and labelled regions were extracted and used for the dataset creation; The obtained initial dataset was composed of 428 sclerotic glomeruli and 2344 non-sclerotic glomeruli, with a ratio between the two classes of $1/5.48$. Specifically, a total of 2772 glomeruli were labelled, and an average value of 26 glomeruli per section and 106 glomeruli per biopsy were collected.

The dataset was subsequently divided into two subsets for train-validation (trainval) and a test phase. In particular, the trainval subset has been further split into train and validation subset and the last one is used for hyperparameters tuning; 20% of the original dataset, instead, has been used as test subset, and the information of the test targets has been used to assess final performances only. The selection has been achieved randomly with the constraint that if a glomerulus appears in the test subset, all the other glomeruli belonging to the same biopsy must appear in the test subset only. This is equivalent to state that the trainval/test division has been performed at biopsy level. The constrained division avoids that particular context information could be present in both the datasets leading to an unfair dataset split.

A detailed overview of the dataset is reported in Table 3.1.

Table 3.1 Dataset overview.

Dataset	**WSIs**	**Non-sclerotic**	**Sclerotic**	**Ratio**
Trainval subset	*19*	*1852*	*341*	*5.43 : 1*
Test subset	*7*	*492*	*87*	*5.66 : 1*
Total	*26*	*2344*	*428*	*5.48 : 1*

3.1.2 A Classic Approach for Glomeruli Classification

One of the main task required by the Karpinski score evaluation is the classification of the glomeruli between two main conditions: sclerotic and non-sclerotic glomerulus. As stated in the previous sections, literature accounts several independent applications of feature extraction algorithms to achieve this purpose; in particular, the authors make use of specific and unique image processing algorithms applied on different types of staining and non-human WSIs.

In the following sections, a combination of different feature extraction algorithms has been designed and evaluated. The proposed set of features, come from a collection of two wide-used, well-known and general-purpose features extractor algorithms families; in particular, morphological and texture features have been computed. These feature families include some of the algorithms proposed in literature and they were extracted from human WSIs with Periodic acid–Schiff (PAS) staining. The set of features was then reduced by means of feature reduction algorithms and then used as input to a shallow Artificial Neural Network. Starting from the dataset presented in the previous *Materials* section, to reduce the variability introduced with the manual annotation, each labelled region was surrounded by a rectangular bounding box with a 1.1 overestimation factor for each dimension. The high number of parameters needed to tune the used image processing algorithms require a subsequent dataset division; then, the trainval subset has been further split into a train set and a validation set. The last one is used for tuning hyperparameters and for assessing the trend of the loss function and of the accuracy during the training process.

As depicted in Fig. 3.2, the discrimination process could be divided in three main steps. The first two allow the extraction of several features and the reduction of their space by means of feature reduction algorithm; the last one leads to the assignment of the label. The details of the full workflow are subsequently described.

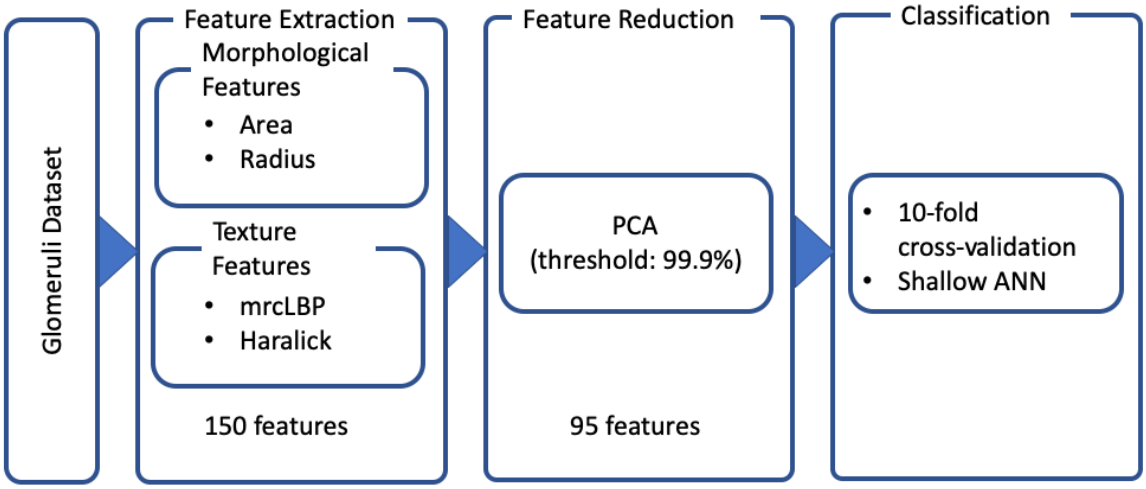


Fig. 3.2 Full features extraction and classification workflow.

3.1.2.1 Features Extraction

The features extraction is the first step of the work-flow allowing the definition of a set of characteristics able to define and discriminate between the two different types of glomeruli.

Based on the human reasoning used by the physicians able to address the problem, the best features to face the problem are those related to two main image processing techniques: morphological and texture-based features.

As stated in Section 2.4.1.2, the main distinctions between sclerotic and non-sclerotic glomeruli are the shape of the Bowman's capsule, the different dimension and the different texture due to blood vessels (Fig. 2.33 and Fig. 2.34). The evaluation of the output of image processing algorithms, the threshold values and the decision regarding the best algorithms configuration have been done on train set only.

Morphological Features Regarding the morphological characteristics, two features related to the Bowman's capsule and the Bowman's space were selected.

The first feature is computed as the sum of the areas related to the Bowman's capsule, the blood vessels areas and the inter-capillary spaces. Due to the PAS staining, these structures are characterized by a whiteness colouration and the detection of the mask describing the region is based on three parallel image processing procedures. Each process took into account the channels of three different colour space: RGB, CMYK and Lab. In detail:

- green channel of RGB colour space, as it is the most representative of the glomerulus structure [235];
- complementary of Magenta from the CMYK colour model has been chosen due to the detectable empirical significance of this colour component (Fig. 3.1, Fig. 2.33 and Fig. 2.34);
- *a* and *b* components of Lab colour space due to the link with the human colour vision.

An example of the application of the processes on non-sclerotic and sclerotic glomeruli is reported in Fig. 3.3 and Fig. 3.4, respectively.

The extraction of the masks from green and magenta channels follows the same following steps:

1. *binarisation*: to keep the pixels related to white regions, a threshold value has been empirically set to 190 [236];
2. *morphological operators*: to clean the image obtained from the previous step, erosion, dilation and median filtering have been used with a disk of radius ranging from 1 to 3 as structuring element.

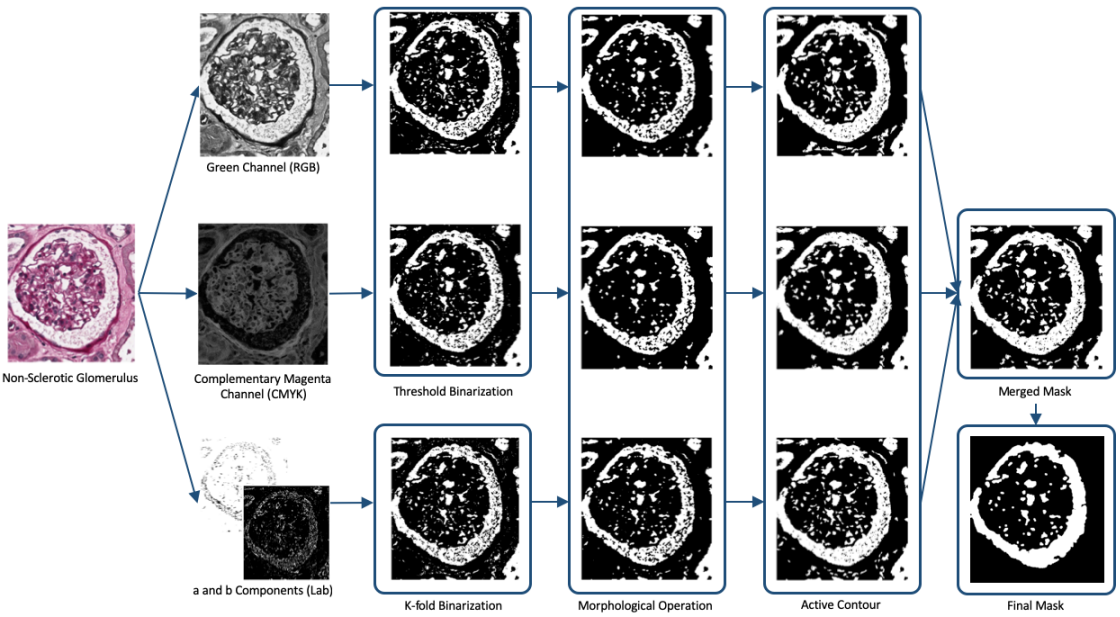


Fig. 3.3 Example of application of morphological features work-flow on non-sclerotic glomerulus. Each row depicts the results of applying the image processing steps on the different colour spaces.

3. *active contour*: to clean the shape of the obtained mask, active contour algorithm [467] has been used with 200 iterations (the chosen number of iterations avoid an extreme smoothing of the glomerulus shape).

The three previous steps led to the computation of two masks, one for the green and one for the magenta channel; the last mask was computed from *a* and *b* components of Lab colour space. The *ab* matrix has been used as input to k-means clustering algorithm [468]; the number of clusters was set to 5, and the number of repetitions of the clustering process using new initial cluster centroid positions to avoid local minima was set to 3. The mask was computed subsequently by retaining just the pixels belonging to the cluster with the greatest mean grey-scale intensity value. Then the steps two and three of the Green-Magenta segmentation process were applied. The number of clusters has been empirically chosen as suggested by the visual observation of the input images. The number of iteration, instead, has been chosen as a trade-off between speed and results; the criterion for the final mask computation and the right choice of the clusters number (5) brought to results stability despite parameter changes.

The final mask was the composition of the resulting three masks computed with a majority criterion: only the pixels belonging to at least two masks were considered. The obtained

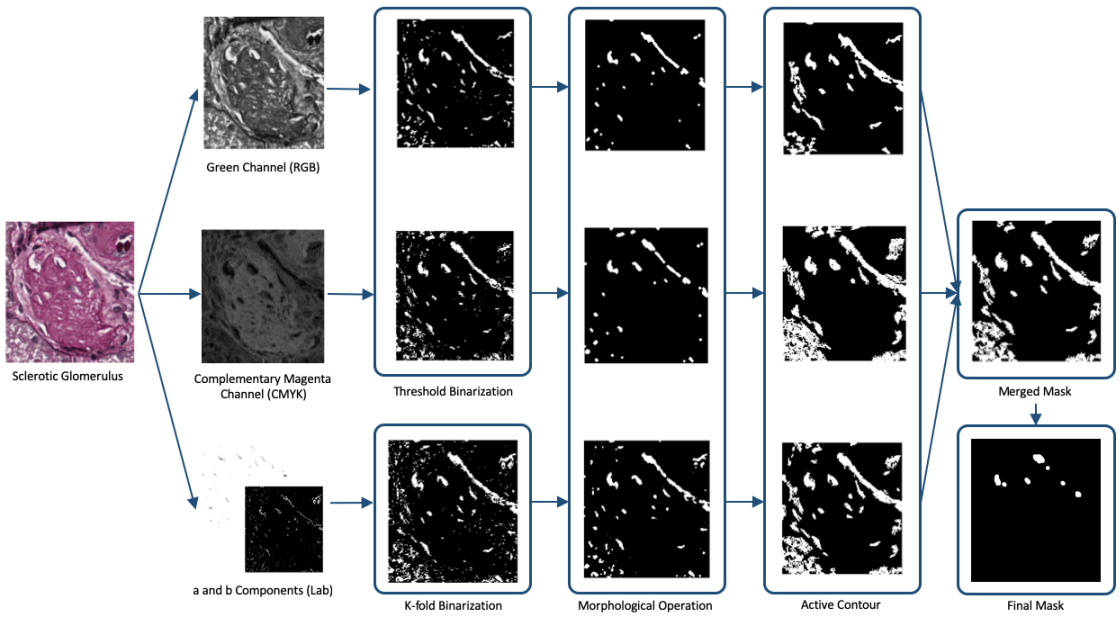


Fig. 3.4 Example of application of morphological features work-flow on sclerotic glomerulus. Each row depicts the results of applying the image processing steps on the different colour spaces.

mask was processed to remove artefact and not interesting regions; in detail, too small regions (lesser than 1000 pixels) were removed, and a logical AND with a circle of radius equal to the smaller dimension of the image subtracted by $1/8$ of its value was performed. Fig. 3.5 shows the overview of the Bowman's space segmentation workflow.

Starting from the final mask (Fig. 3.3 and Fig. 3.4), the feature of interest was the sum of Bowman's space, blood vessels and the inter-capillary region of the glomerulus, that is, in the presented work-flow, the sum of white region. This value was finally normalised considering the image area. A results comparison of the application of the work-flow mentioned above both on sclerotic and non-sclerotic glomeruli is reported in Fig. 3.6.

The second morphological feature was related to the diameter of the glomerulus. Assuming that the white region inside the mask computed for the last feature was related to the shape of the glomerulus, the convex hull containing all these regions was computed. Then, considering the convex hull ROI as a circle, the diameter of a circle with the equivalent area were computed.

As a result of the morphological work-flow, a total of two features were computed: the area and the radius.

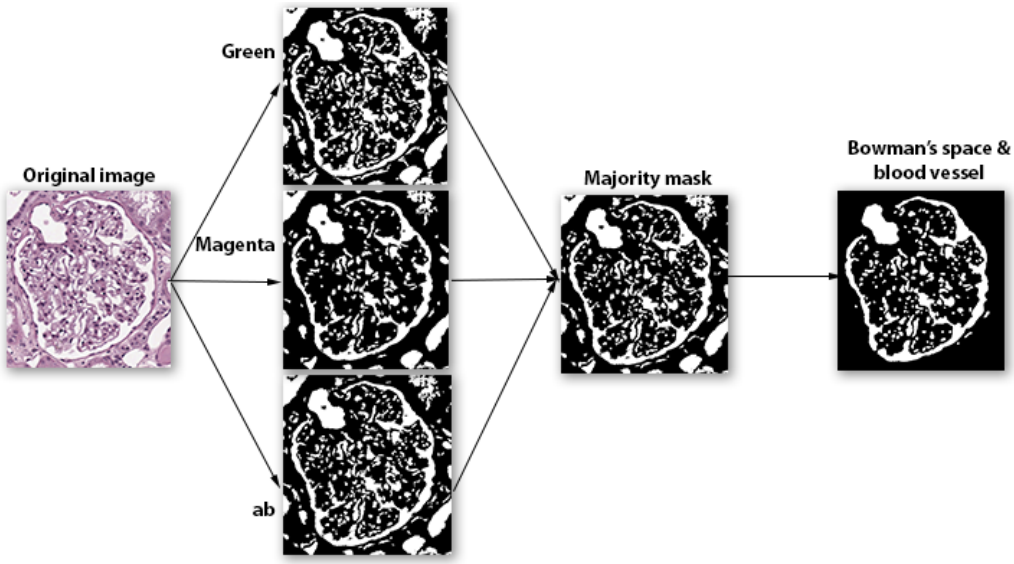


Fig. 3.5 Example result of the segmentation of Bowman's space.

Texture Features Due to the particularity of the glomerulus texture and the differences in blood vessels and inter-capillary space between sclerotic and non-sclerotic, two well-known texture analysis algorithms were used: Local Binary Pattern (LBP) and Haralick features.

Wan *et al.* analysed different types of texture features, founding LBP features as representative in classifying human breast tissue images, without tissue staining, by optical coherence microscopy [469]. Authors proposed two LBP variants: average LBP and block based LBP, that provided an enhanced encoding of the texture structures if compared with the classic LBP feature. The authors tested the proposed algorithms on human breast tissue samples, with and without breast carcinoma, and the corresponding H&E histologies were used as ground truth for comparison. The authors reported that the classification reached up to 93.8% of accuracy with a combination of the aforementioned features. Simon *et al.* [233], instead, proposed a multi-radial colour LBP (mrcLBP) as a suitable variation of classical LBP to face the glomerulus identification problem. In detail, it is the application of the LBP algorithm to the three RGB colour channel with different radius values (1, 3, 9 and 27) and with invariance to rotation. The same configuration was chosen for this application and applied to the raw RGB glomerulus images. The obtained features were ten for each radius, thus leading to a total number of 120 (10 features per radius, 4 radius, three channels).

The second set of texture-based features was obtained from the extraction of Haralick features. The four Grey-Level Co-occurrence Matrix (GLCM), one for each direction, has been computed; then, the 14 Haralick indexes were computed, leading to 56 features. To reduce this number, the mean and the range (difference between the maximum and the

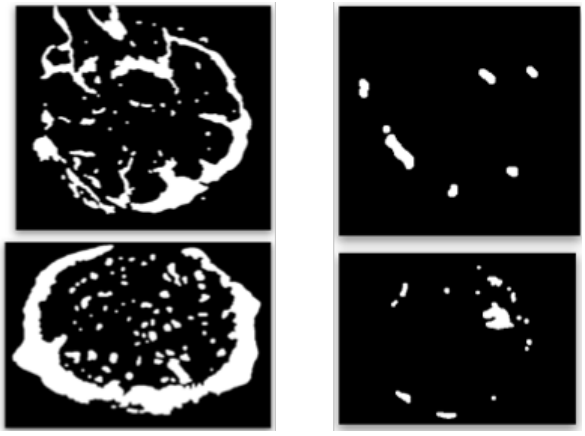


Fig. 3.6 Results comparison between the application of Bowman,s space segmentation on non-sclerotic (left) and sclerotic (right) glomeruli.

minimum values) among the four directions was calculated. The final features were 28 (14 mean and 14 range, one for each Haralick feature).

As a result of the texture features extraction, a total of 148 features were computed.

3.1.2.2 Features Preprocessing

The created set of features is the union of both morphologic and texture-based features. An overall number of 150 features was achieved. Due to the possibility of correlation among the different subsets of features, and to reduce the total number of inputs to the classification step, Principal Component Analysis (PCA) was applied as feature reduction algorithm. Prior to PCA, each feature was z-score normalized.

As stated in the *Materials* section, the dataset was previously split into train and test set with the aim to fairly take all the image pre-processing and classification decisions on the train set only. The feature reduction algorithm, instead, does not need or use the label information; for this reason, the application of PCA could be done on the whole dataset or on the training dataset only. Both methods present advantages and drawbacks. Applying the PCA on the whole dataset has the vantage to take into account all the information inside the dataset, but at the same time it requires the reapplication of the feature reduction process on new data. Assuming that the statistic of the dataset is the same and equal for each subset (this is likely true since the train and the test set were randomly created), the PCA can be applied on the train set only, with the drawback that it is necessary to preserve the transformation matrix to manipulate the test set before the testing phase. As an example, both the solution were applied on the dataset, and due to the complexity of the classification problem, 99.9%

of variance has been chosen as the threshold value. The application of PCA to reduce the input space led to 95 and 93 features for the whole dataset and the training subset only, respectively. Due to the small differences between the number of the two reduced dataset, to take into account all the information inside the dataset, and to avoid the necessity to preserve the transformation matrix for the test phase, the first approach has been chosen.

As a result of the PCA as feature reduction algorithm, a total number of 95 features were computed and will be used for the classification phase.

3.1.2.3 Glomeruli Classification

The glomeruli classification steps are based on ANN and specifically on a shallow ANN architecture.

The design of the ANN architecture and the tuning of its parameters were taken considering the trainval set only, whereas all the reported results and performance discussions refer to the test set. To generalise, to avoid overfitting and to obtain a classifier independent from the input dataset, k-fold was used as cross-validation technique. Several network initialisation inside each fold and hard voting among the folds was used both to obtain independency from a particular network initialisation and to compute the overall fold class label.

The input of the classifier was the features set obtained from the image processing algorithm and the subsequent PCA feature reduction; the number of input features was 95, and 10-fold cross validation was used. The fixed training parameters were the following: one hidden layer, tansig and softmax as activation functions for the hidden and output layer, respectively, crossentropy as loss function and scaled conjugate gradient as backpropagation algorithm. A training early stop criterion, based on the validation set, was implemented to promote generalisation and to avoid overfitting; the stop criterion occurs if performance on validation set did not decrease inside a sliding window of 6 epochs. The other available stop criteria (i.e., max number of epochs, time, etc.) have been set to not interfere. The last relevant parameter is the number of neurons for the hidden layer, and the choice of the right value is afterwards reported.

In the following paragraphs the different assumption and decision related to the network configuration and tuning are described.

Unbalanced Dataset Problem. As reported in the *Materials* section, the training set was affected by a heavy unbalanced distribution between sclerotic and non-sclerotic glomeruli (with a proportion of 5.5 non-sclerotic over one sclerotic glomerulus). Data augmentation as

balancing technique was not suitable in this case since the particular set of features taken into account presents invariance to the main image transformations.

The first solution used to address the problem was the use of the Matthews Correlation Coefficient (MCC) [182] as a general performance comparison among the folds. As described in Section 2.3.1, MCC (Eq. 2.20) takes into account the ratio of the confusion matrix sizes suiting better than accuracy (Eq. 2.13) or F-measure (Eq. 2.19) on unbalanced datasets.

The second solution make use of the ROC curve to choose the correct classification threshold value. Two approaches were analysed. The first one (Approach A) assumes the optimal value as the first intersection point between the ROC curve and a line with slope equal to the ratio between the total number of negative and positive samples and sliding from the upper left corner of the ROC plot ($(FPR, TPR) = (0,1)$). The second approach (Approach B), proposed by Songet *et al.* [470], evaluates the point of minimum distance from the point (0, 1) of the ROC plot. The equation is reported in Eq. 3.1.

$$\min_i(\sqrt{(1-sensitivity(i))^2+(1-specificity(i))^2}) \tag{3.1}$$

The comparison of the two methods in terms of different performance indexes (Eq. 2.13, 2.14, 2.15 and 2.20) is reported in Table 3.2. Due to the medical domain of the work, a higher recall is preferred, thus the second method was chosen (i.e., in the medical domain, a correct prediction of positives to a disease factor is more important than the prediction of negatives).

Table 3.2 Comparison between the two ROC thresholding approaches. The reported values are the mean among the 10-fold.

Performance Index	**Approach A**	**Approach B**
Accuracy	*0.9898*	*0.9865*
Precision	*0.9775*	*0.9332*
Recall	*0.9575*	*0.9880*
MCC	*0.9612*	*0.9520*

Network Tuning The architecture of the shallow ANN chosen for glomeruli classification was fixed to one hidden layer. To choose the right number of neurons per layer, the performance of 95 networks were compared. In detail, several networks with the number of neurons for the hidden layer ranging from 1 to 95 were trained (95 is the constant number of input features). Based on the best MCC value computed as the mean MCC of the folds, the

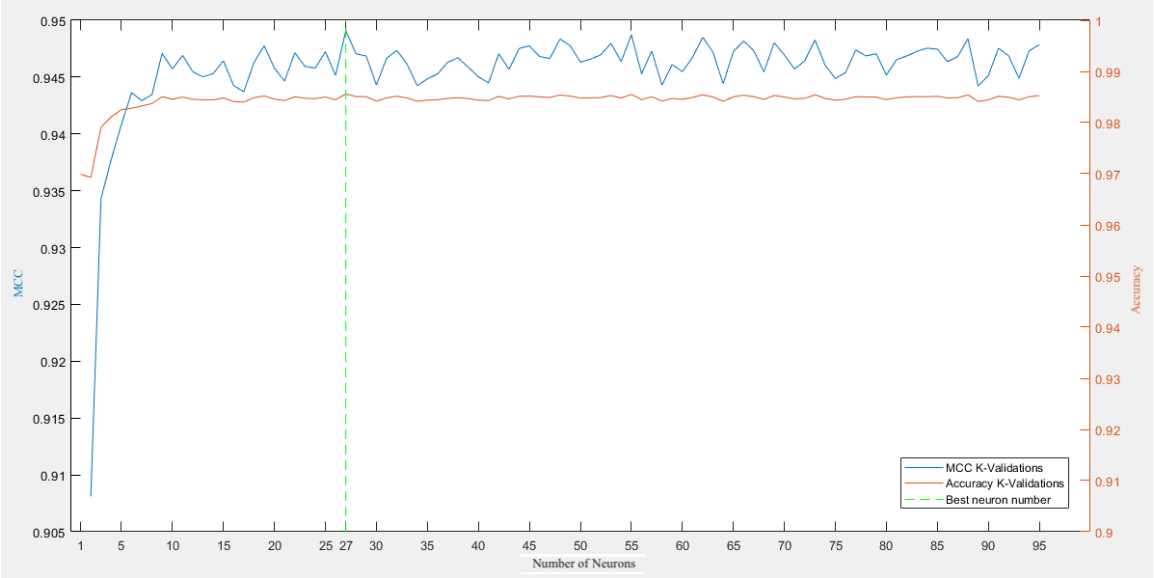


Fig. 3.7 MCC and accuracy trend based on number of neurons.

final number of neurons for the hidden layer was set to 27. A graphical representation of the trend of MCC and accuracy indexes is shown in Fig. 3.7; the final artificial neural network configuration is summarised in Table 3.3.

Table 3.3 Artificial neural network configuration for glomeruli classification. *An in-depth explanation about the neurons number choice is reported in *Network Tuning* paragraph. **See Section Glomeruli Classification for further details.

Parameter Name	**Value**
# input	95
Topology	[27*, 1]
Activation functions	[tansig, softmax]
Loss function	Cross-entropy
Backpropagation algorithm	Scaled conjugate gradient
Early stop criterion	Validation fail**
Cross validation method	k-fold ($k = 10$)

3.1.2.4 Results

Several metrics were considered for the evaluation. In particular, Accuracy (Eq. 2.13) and Matthews Correlation Coefficient (MCC) (Eq. 2.20) were evaluated on the independent test set, according to the confusion matrix reported in Table 2.5.

Table 3.4 Metrics comparison of 10 network initialization.

Performance Indexes (mean $\pm$ standard deviation)	
Accuracy	*0.9874 $\pm$ 0.0018*
Precision	*0.9844 $\pm$ 0.0111*
Recall	*0.9310 $\pm$ 0.0153*
MCC	*0.9501 $\pm$ 0.0074*

Table 3.5 Metrics comparison of the best network.

Performance Indexes	
Accuracy	*0.9914*
Precision	*1.0000*
Recall	*0.9425*
MCC	*0.9659*

To evaluate the workflow stability, 10 runs of the whole process were performed, and the corresponding results are summarized in Table 3.4; the results are reported in terms of mean and standard deviation. The best result, instead, is reported in Table 3.5 and the corresponding confusion matrix is reported in Table 3.6. Examples of misclassified glomeruli are reported in Fig. 3.8.

As reported in Table 3.4, the classification workflow achieved a mean MCC and Accuracy of 0.95 and 0.99, respectively, and low variability over the 10 independent iterations ($MCC\ std = 0.01$ and $Accuracy\ std < 0.00$). Good precision and recall were obtained too (precision: 0.98 ± 0.01, recall: 0.93 ± 0.02), showing a better performance in the non-sclerotic evaluation (all the non-sclerotic glomeruli were detected in the best case).

The misclassified glomeruli are usually affected by staining artefacts or present partial sections due to slice preparation and staining (mainly the ones positioned on the edges); common examples are reported in Fig. 3.8. These images were submitted to domain experts

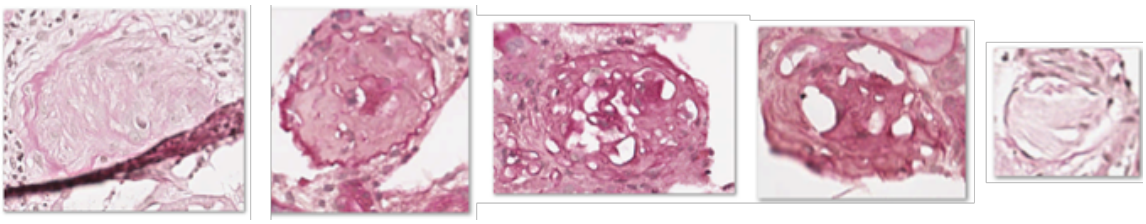


Fig. 3.8 Examples of false negative misclassified by the best model. Some of the misclassified samples present staining artefacts of are partially complete due to slice preparation; the domain experts stated that glomeruli affected by these kind of artefacts are usually discarded.

Table 3.6 Confusion Matrix of the best network.

		True Condition	
		Positive (sclerotic)	**Negative (non-sclerotic)**
Predicted Condition	**Positive (sclerotic)**	*82*	*0*
	Negative (non-sclerotic)	*5*	*492*

that assessed the goodness of the classification results; in particular, the staining artefacts and the partial glomeruli makes difficult their classification and they are usually discarded by pathologists even in the clinical practice.

Compared to the literature, the proposed workflow make use of a combination of several image features coming from the morphological and texture feature families; furthermore, the application of the PCA as features reduction algorithm and the optimal tuning of the artificial neural network by means of data cross-validation, led to an improvements of the results if compared to the reported literature [233, 239]. The work-flow overcome the common data unbalancing problem by using MCC as performance comparison coefficient and ROC curve. Finally, the reported results suggest that the proposed workflow set-up is a valid and reliable solution for the investigated domain, supporting the clinical practice of discriminating the two classes of glomeruli. However, there are still common glomeruli misclassification in images affected by artefacts, which are usually discarded by pathologists even in the clinical practice. Furthermore, the integration of the proposed CAD pipeline with the CAD system designed and tested by Bevilacqua *et al.* [201] for the segmentation and discrimination of blood vessels versus tubules from biopsies in kidney tissues, will allow to build-up a complete CAD system, based on image processing, for the analysis of histopathological WSIs.

3.1.3 A Deep Learning Approach for Glomeruli Classification

The first application of deep learning techniques on the glomerulosclerosis problem regard the classification between sclerotic and non-sclerotic glomeruli. Since this was the first insight about deep learning architectures, it was compared with an online service: IBM Watson Visual Recognition [471]. The service allows the use of pre-trained networks to accomplish common and very specific tasks, such as generic image classification, face and age detection; the system offers the possibility to create and train a custom classifier to suit specific business needs and extend the classification capabilities to images not available in the pre-trained models. The Watson Visual Recognition service requires to upload images to learn and create a new classifier. Each example file is trained against the others, and positive examples are stored as classes. These classes are grouped to define a single classifier but return their own scores. The steps to create a new classifier are shown in Fig. 3.9. The advantages of the Watson service is the simplicity of use; the training of a new classifier requires to provide a compressed file with the two classes with at least ten samples. Furthermore the tool gives some suggestions to improve the results:

- provide images with a resolution of at least $224 \times 224 pixels$. Since the whole pipeline is a black box for the user, this size limitation leads to guess that classic topology, such as AlexNet, constitute the backbone of the classifier;
- include a negative class. When there is more than one class, the negative class contains all the elements that don't belong to none of the other classes;
- include approximately the same number of negative images as positive ones. An unequal number of images might reduce the quality of the trained classifier;
- if different models are created, to choose the best one, the creation of a validation set is necessary. On this dataset all the model are evaluated to find the best one that will be the final model to be used for the test.

Several models were trained using the available configuration and by using the two class (multi class model) and the positive class only where the negative class is deduced by the modes as "non-positive" (binary class model); furthermore, the dataset positive class was augmented with $90°$, $180°$, $270°$ rotations and a random distortion, and all the classes were converted to grayscale. The resulting combination are:

- augmentation only;
- augmentation and resizing;

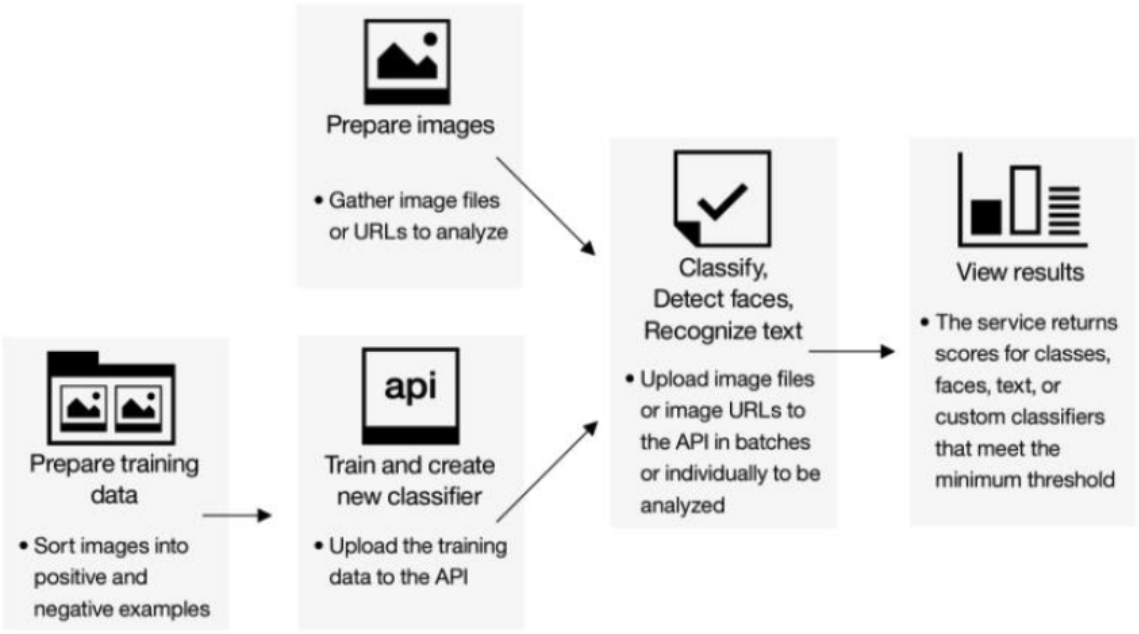


Fig. 3.9 Visual Recognition process with custom classifier. Image from Elhassouny *et al.* [471].

- augmentation and grayscale conversion;
- augmentation, gray scale conversion and resizing.

The best model on validation set was chose and used for final results. A cross validation could not be used because all the model training were deterministic resulting in the same final performance (with a given input training set and model).

3.1.3.1 Custom Model

A custom and local model was created to be compared with the Watson online service. The model is based on Inception V3 [253] and the same input resolution was used to provide the most similar input to the Watson model. Furthermore, to reduce the computational cost and speed up the training the images were converted to gray scale. To resolve the unbalanced data problem the same operations were performed on the positive class: 90°, 180°, 270° rotations and random distortion. All the images were finally normalised to zero mean and unit standard deviation. The training phase was configured with batch size equal to 32 and Adam optimiser with learning rate set to 10^{-6}; to help the back-propagation steps, the under-represented, and more important for clinician, positive class received a greater weight. The model was

Table 3.7 Results comparison.

Model	Accuracy	Precision	Specificity	Recall	F-measure	MCC
Proposed	*0.98*	*0.95*	*0.99*	*0.91*	*0.93*	*0.92*
Watson	*0.99*	*0.97*	*0.99*	*0.97*	*0.97*	*0.97*

Table 3.8 Custom model results over the folds.

Fold	Accuracy (mean ± std)	Precision (mean ± std)	Specificity (mean ± std)	Recall (mean ± std)	F-measure (mean ± std)	MCC (mean ± std)
1	*0.97 ± 0.00*	*0.90 ± 0.02*	*0.98 ± 0.00*	*0.92 ± 0.02*	*0.92 ± 0.02*	*0.90 ± 0.01*
2	*0.97 ± 0.00*	*0.89 ± 0.00*	*0.98 ± 0.00*	*0.94 ± 0.01*	*0.91 ± 0.02*	*0.92 ± 0.00*
3	*0.98 ± 0.00*	*0.93 ± 0.01*	*0.99 ± 0.00*	*0.92 ± 0.01*	*0.93 ± 0.00*	*0.91 ± 0.00*
4	*0.97 ± 0.00*	*0.89 ± 0.01*	*0.98 ± 0.00*	*0.90 ± 0.02*	*0.89 ± 0.00*	*0.88 ± 0.01*
5	*0.97 ± 0.00*	*0.87 ± 0.01*	*0.98 ± 0.00*	*0.93 ± 0.01*	*0.90 ± 0.01*	*0.88 ± 0.01*

cross-validated with 5 folds, and early stop criterion on validation loss was set to 40 epoch to halt the training.

3.1.3.2 Results

For Watson models, the oversampled, grayscale converted and resized dataset achieved the best performance on the validation set, and the final results on the test set are reported in Table 3.7. Custom model performance over the five folds are reported in Table 3.8, while result comparison between the Watson and the best custom model are summarised in Table 3.7; Watson model provided performance slightly better than the proposed custom model.

An important analysis can be done on the misclassified images. Fig. 3.10 depict the images that the best execution of the custom model misclassified. Two of the three images misclassified in Fig. 3.10a present artefacts along the whole image due to a bad acquisition and this probably compromised the classification; Fig. 3.10b, instead, reports false negative examples, a guess about the misclassification is that the model has been influenced by the remaining part of the extracellular matrix.

Examples of the Watson model misclassification are depicted in Fig. 3.11; the same considerations about the custom model errors can be done in this case.

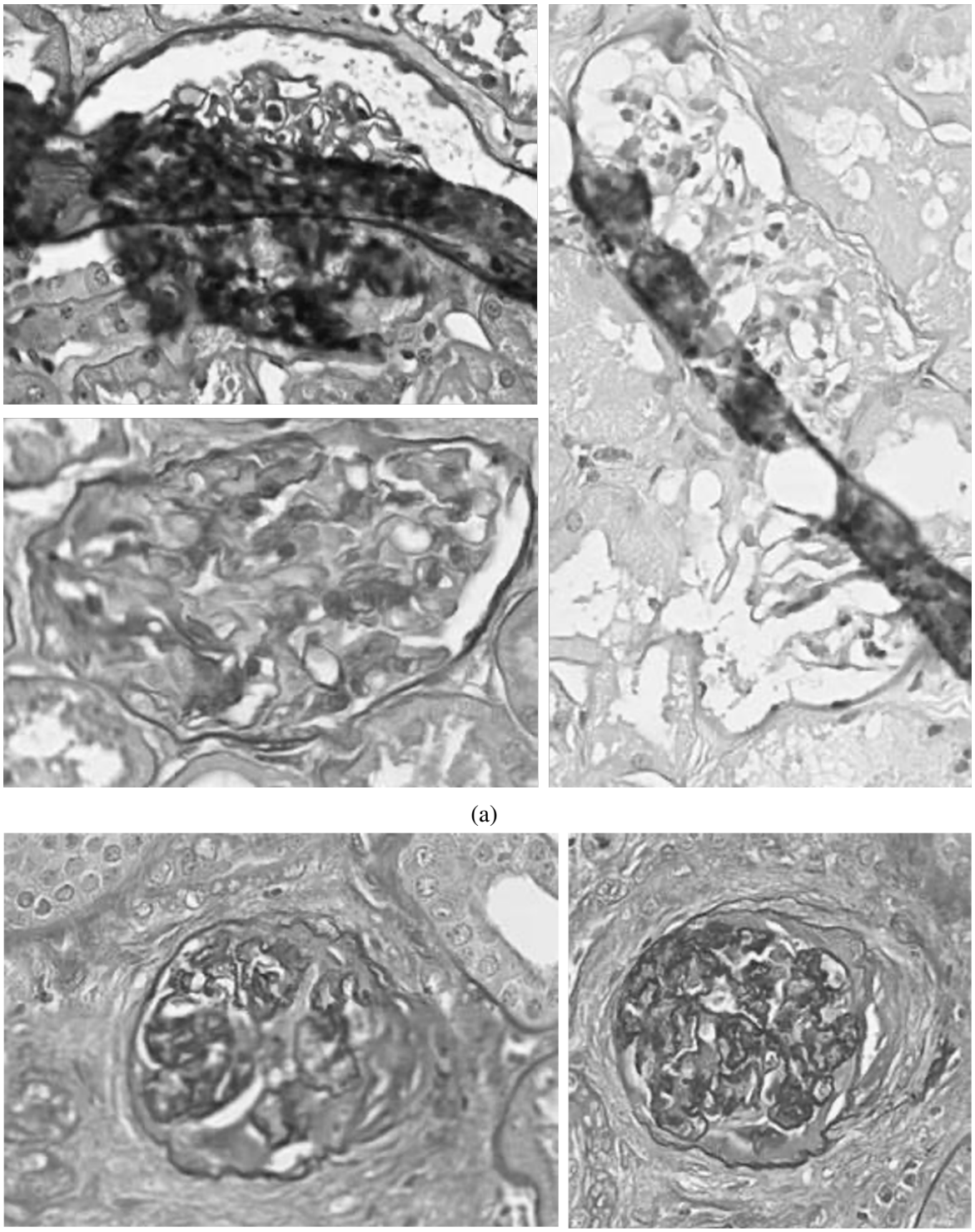
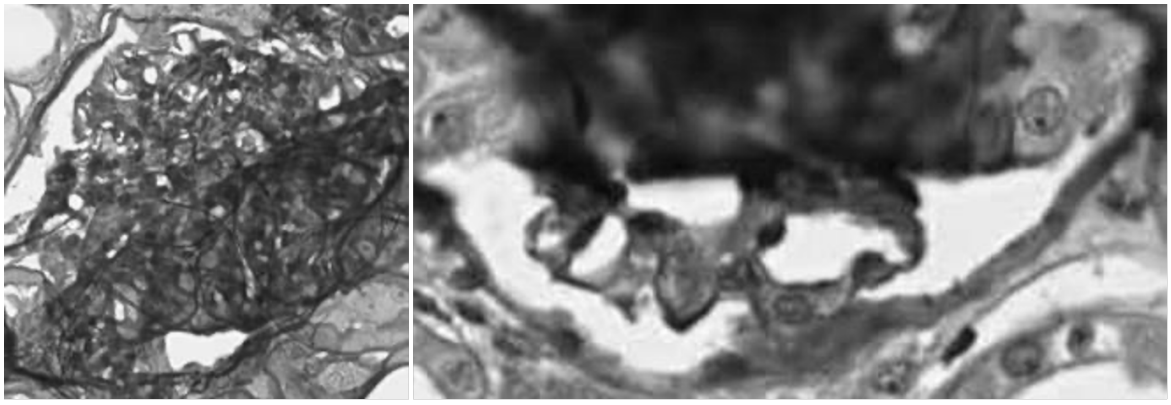
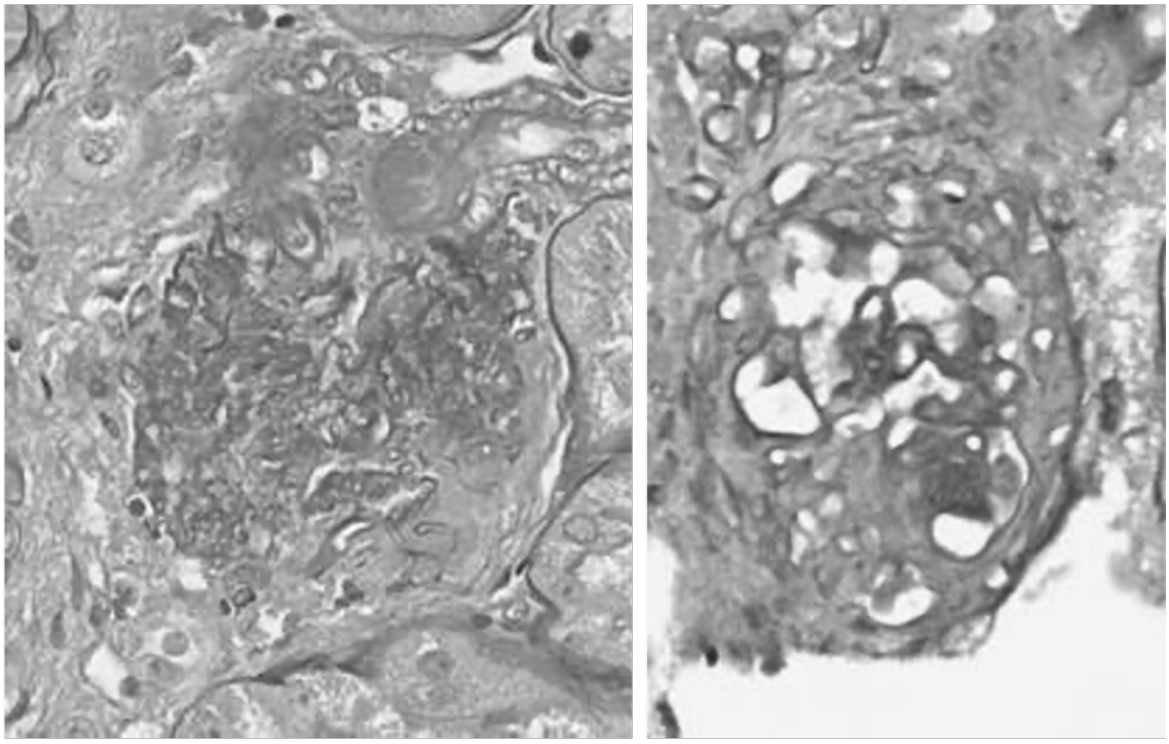


Fig. 3.10 Examples of misclassified glomeruli from the custom model: (a) false positive, (b) false negative.



(a)



(b)

Fig. 3.11 Examples of misclassified glomeruli from the Watson model: (a) false positive, (b) false negative.

3.1.4 A Deep Learning Approach for Glomeruli Detection

The results obtained in the previous sections permit to state that images about glomeruli and extracted from PAS stained slices, have enough information to allow the discrimination between sclerotic and non-sclerotic. To further investigate the capabilities of machine learning algorithms, and in particular deep learning ones, in this section a full deep learning based workflow for glomeruli detection will be presented; the workflow outline is reported in Fig. 3.12. The deep learning model is based on the Faster R-CNN detector.

3.1.4.1 Work-flow Design and Model Configuration

The sections segmentation is a trivial task due to the easiness of the segmentation of coloured regions on near-white background; at this purpose classical image processing techniques, well-known and already used in literature [200, 201], have been used; to facilitate and reduce the computational cost of the following steps, the extracted section ROIs were under-sampled with a four factor. In detail, the original WSIs magnification of $20\times$ is reduced to $5\times$ by under-sampling; this lead to a down-sampling of the images from a mean resolution of about 8000×8000 pixels to about 2000×2000 pixels. All the resizing operations are obtained by means of bi-cubic and nearest-neighbour interpolation for digital pathology images and categorical masks, respectively (the annotation have to be rescale according to the corresponding raw images to be used).

The under-sampled biopsy sections are then divided with stride of 250×250 into patches of size 500×500. The stride has been chosen to guarantee an overlap of 250×250, so that there is at least one patch in which each glomerulus is fully contained. Since the dimensions of glomeruli in images at full resolution ($20\times$) are lesser than 800×800, at undersampled resolution ($5\times$) they are lesser than to 200×200, thus the claimed condition is easily obtained. In this way any glomerulus is discarded from training data (the same procedure can be applied in inferencing phase, avoiding glomeruli missing). Dividing the original image into patches poses the problem on how the partially contained glomeruli should be considered in the training patch (Figure 3.13a and Figure 3.13b show example of not complete glomeruli). At the purpose of solving this issue, it has been introduced an hyperparameter, called $tolerance$, indicating the maximum percentage of glomerulus size allowed to be out of patch before considering that glomerulus as a positive example for training. For example, a $tolerance = 0$ will allow only glomeruli fully contained in each patch as positive examples for training. This means that, even if a glomerulus is out of $1px$, it will not be used as positive sample. Testes show that optimal values for this parameter are

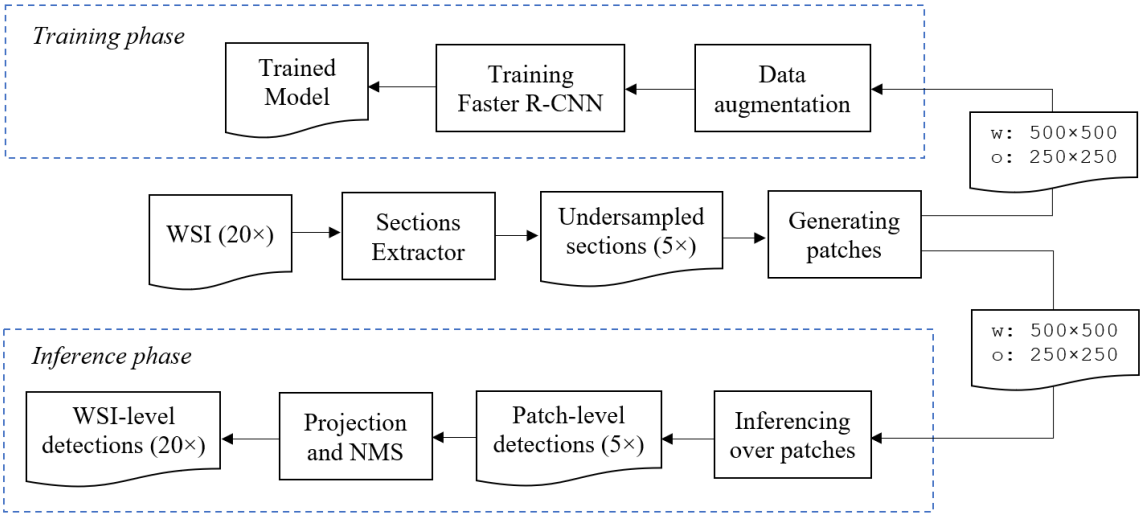
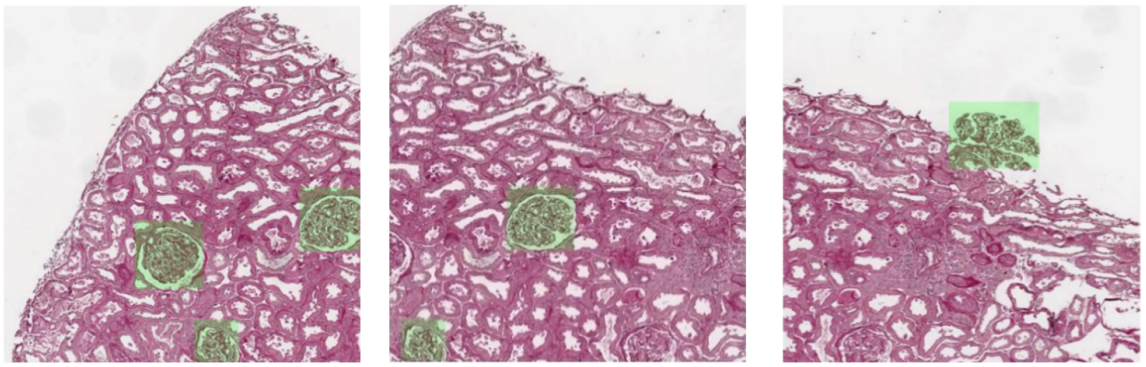


Fig. 3.12 Object detection work-flow based on Faster R-CNN model. The top part describes how to perform the training of the model, exploiting the train-validation set (19 whole slide images). The bottom part explains how to use the trained model for performing inference on the test set sections (7 whole slide images).

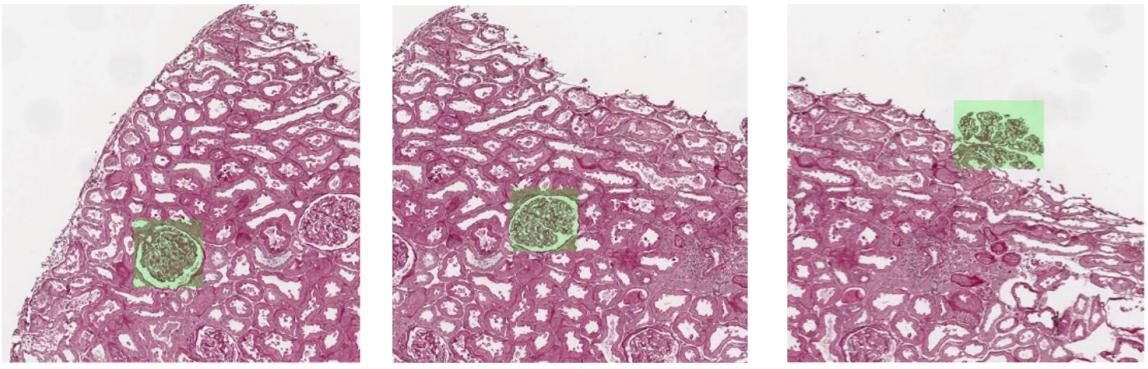
approximately in the range $[0.2, 0.4]$. The final proposed Faster R-CNN detector has been trained with $tolerance = 0.3$.

Even if the dataset presented in the previous *Materials* section, composed of 26 WSIs and 101 sections, have an high number of glomeruli, for detection tasks the number is relatively low. Furthermore, due to the strong unbalancing between healthy and non-healthy glomeruli (with a ratio of $1/5.48$.), oversampling has been chosen for rebalancing. In particular, each training patch containing at least one sclerotic glomerulus (that is the underrepresented class), has been augmented by rotating the patch by 90°, 180° and 270°. In this way, the number of scleroic glomeruli is roughly quadruplicated. Since the augmented patch may contain also non-sclerotic glomeruli, a slight increase of their number has to be accepted; in any case, a more balanced dataset is obtained.

The model has been trained on small patches and it is not directly applicable for inferencing on new sections; moreover, the sections sizes may be too large (up to 2500×2500) and infeasible to process. To use the model in the inferencing phase it necessary to pre-process the new sections; then, the same procedure used on training data, can be applied to obtain patches of size 500×500 with stride of 250×250. As before, this allow to reduce glomeruli miss, but introduce the new problem of the replicate detection of glomeruli; due to overlap, the projection of patch-level detections on original image will produce a multiple detection of glomeruli found out the overlap region (Fig. 3.14). NMS algorithm is used to suppress



(a)



(b)

Fig. 3.13 Effects of *tolerance* hyper-parameter on incomplete glomeruli. Patches with $tolerance = 0.3$ (a) and $tolerance = 0$ (b).

duplicate bounding boxes (Algorithm 1); in detail, two iterations are applied: a first one that make use of the standard NMS algorithm and a second one based on NMS using IoM instead of IoU. The second modified iteration is useful in the case of small bounding boxes mainly contained inside larger ones, and is applied only when IoM equals or exceeds a threshold fixed to 0.5. A value of 0.3 is used for the standard NMS in the first step. The application of NMS on the bounding boxes detected in Fig. 3.14 is depicted in Fig. 3.15.

Regarding the training of Faster R-CNN detector, the used hyper-parameters configuration is reported in Table 3.9. Faster R-CNN is composed by four different stages, each with its own hyper parameters; each stage is trained for ten epochs with Adam optimiser and the learning rate is lowered in stages three and four. The tuned values are reported in Table 3.10.

3.1.4.2 Results

The trained model obtained an mAP of 0.803, with the overall results summarised in Table 3.12 and confusion matrix in Table 3.11

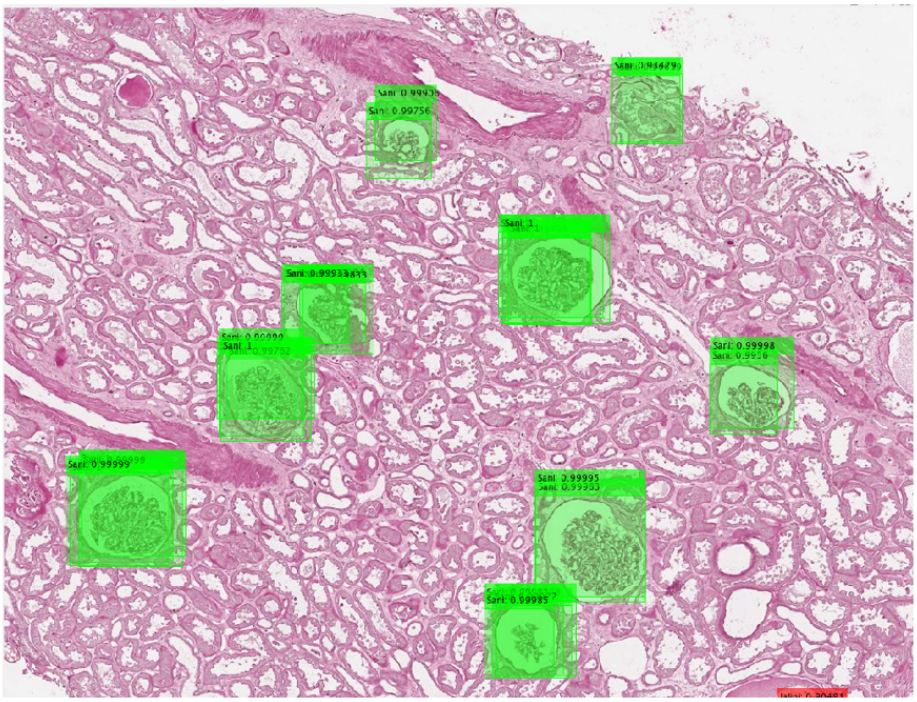


Fig. 3.14 Overlapped bounding boxes after projection in full image.

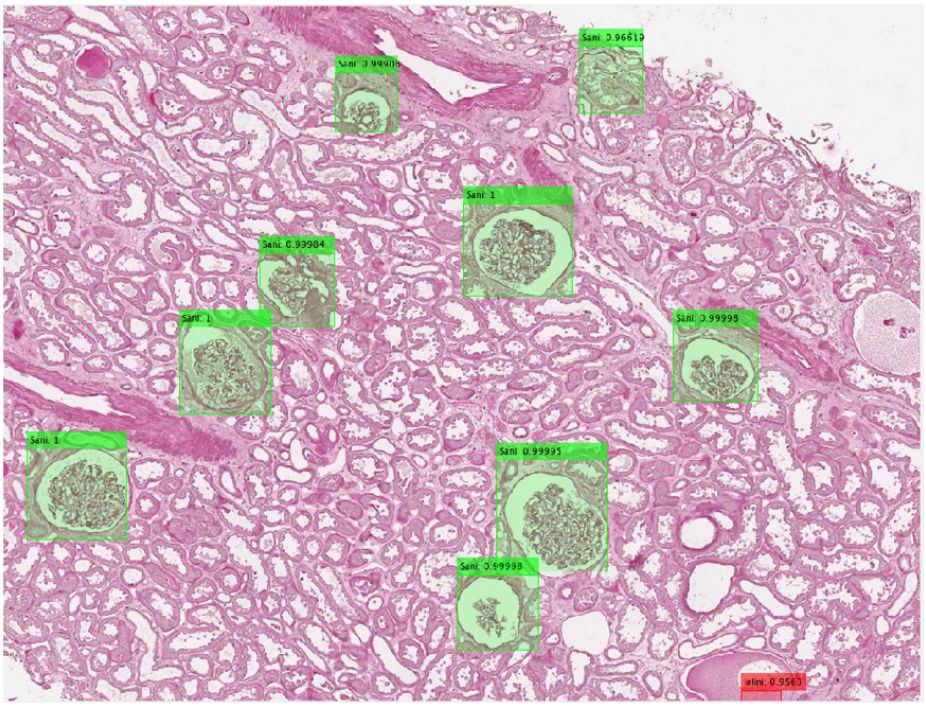


Fig. 3.15 Detection on section after Non-Maximum Suppression based on Intersection over Union.

Table 3.9 Faster R-CNN hyperparameters.

Faster R-CNN	
Hyperparameter	**Value**
CNN	*resnet50*
Negative Overlap Range	*[0, 0.3]*
Positive Overlap Range	*[0.6, 1]*
# Regions To Sample	*256*
Box Pyramid Scale	*1.2*
# Strongest Regions	*512*

Table 3.10 Hyperparameters per stages of Faster R-CNN.

Hyperparameter	**Stage**			
	1	**2**	**3**	**4**
Optimiser	*ADAM*			
Max Epochs	*10*			
Batch Size	*1*			
Initial Learning Rate	*0.0001*	*0.0001*	*0.000001*	*0.000001*

Table 3.11 Object detection confusion matrix with the baseline Faster R-CNN work-flow.

		Prediction		
		Sclerotic	Non-Sclerotic	Background
Ground Truth	Sclerotic	*61*	*7*	*19*
	Non-Sclerotic	*0*	*463*	*29*
	Background	*35*	*62*	–

Table 3.12 Detection metrics with the baseline Faster R-CNN workflow.

Class	**Recall**	**Precision**	**F-score**
Non-Sclerotic	*0.941*	*0.870*	*0.904*
Sclerotic	*0.701*	*0.635*	*0.667*

In order to compare the clinical validity of the workflow performance, the corresponding Karpinski score is computed and compared with that of expert pathologists. As described in Section 2.4.1.2 and reported in Table 2.7 (the involved table section is reported in Table 3.13),

Table 3.13 Karpinski Glomerular Score [225].

Score		Description
Glomerular	*0*	no globally sclerosed glomeruli
	1	$< 20\%$ global glomerulosclerosis
	2	$20-50\%$ global glomerulosclerosis
	3	$> 50\%$ global glomerulosclerosis

the ratio is computed as the number of sclerosed glomeruli divided by the overall number of glomeruli ($Ratio = \frac{S}{S+NS}$), then the score is computed as following: 0, if there are no globally sclerosed glomeruli; 1, if there is $< 20\%$ global glomerulosclerosis; 2, if there is $20-50\%$ global glomerulosclerosis; 3, if there is $> 50\%$ global glomerulosclerosis. The comparison between Faster R-CNN output and ground truth is shown in Table 3.14. The results shown that Faster R-CNN approach makes five errors in assessing the Karpiski score: four times it gives a score of 1 instead of 2, and one time it gives a score of 2 instead of 1.

Respect to literature the proposed workflow accomplish glomerular detection in kidney biopsies considering also the classification between non-sclerotic and sclerotic glomeruli. Starting from the table proposed by Kawazoe *et al.* [198], Table 3.15 reports the full comparison between proposed approach and recent researches; the Faster R-CNN based model performs well in the non-sclerotic glomeruli detection, with very high recall and precision values, but metrics for sclerotic glomeruli suffer from a higher number of false negatives.

Even if the obtained result are promising, further investigations are needed; in particular others deep learning techniques have to be investigated As an overall evaluation, it is possible to observe that glomerular detection and classification tasks should be approached as an instance segmentation tasks. Even if object detection approaches can guarantee respectable results, they do not exploit the mask information in the dataset. Following sections will report these applications and comparison results.

Table 3.14 Karpinski Score, results on hold-out test set. Comparison between Faster R-CNN and ground truth annotations. NS stands for non-sclerotic, S stands for sclerotic. Score belongs to the range $[0-3]$.

Donor	**Kidney**	**Section**	**Faster R-CNN**				**Ground Truth**			
			NS	**S**	**Ratio**	**Score**	**NS**	**S**	**Ratio**	**Score**
1	Left	*1*	*31*	*3*	*0.09*	*1*	*30*	*3*	*0.09*	*1*
		2	*32*	*2*	*0.06*	*1*	*30*	*2*	*0.06*	*1*
		3	*29*	*5*	*0.15*	*1*	*28*	*4*	*0.13*	*1*
		4	*31*	*5*	*0.14*	*1*	*25*	*4*	*0.14*	*1*
		5	*30*	*1*	*0.03*	*1*	*31*	*1*	*0.03*	*1*
		6	*35*	*3*	*0.08*	*1*	*31*	*1*	*0.03*	*1*
2	Right	*1*	*9*	*8*	*0.47*	*2*	*10*	*5*	*0.33*	*2*
3	Right	*1*	*40*	*8*	*0.17*	*1*	*38*	*2*	*0.05*	*1*
	Left	*1*	*38*	*3*	*0.07*	*1*	*41*	*4*	*0.09*	*1*
4	Right	*1*	*23*	*7*	*0.23*	*2*	*17*	*5*	*0.23*	*2*
		2	*29*	*4*	*0.12*	*1*	*25*	*3*	*0.11*	*1*
		3	*29*	*5*	*0.15*	*1*	*25*	*3*	*0.11*	*1*
		4	*28*	*9*	*0.24*	*2*	*25*	*5*	*0.17*	*1*
5	Right	*1*	*23*	*3*	*0.12*	*1*	*22*	*4*	*0.15*	*1*
		2	*27*	*3*	*0.10*	*1*	*28*	*5*	*0.15*	*1*
6	Right	*1*	*14*	*3*	*0.18*	*1*	*13*	*6*	*0.32*	*2*
		2	*13*	*3*	*0.19*	*1*	*13*	*6*	*0.32*	*2*
		3	*13*	*3*	*0.19*	*1*	*14*	*5*	*0.26*	*2*
		4	*12*	*1*	*0.08*	*1*	*12*	*2*	*0.14*	*1*
		5	*16*	*4*	*0.20*	*2*	*14*	*6*	*0.30*	*2*
		6	*20*	*4*	*0.17*	*1*	*17*	*10*	*0.37*	*2*

Table 3.15 Comparison with literature extending the one proposed by Kawazoe *et al.* [198]

Author	**Sp**	**Stain**	**WSIs**	**Method**	**Class**	**Performances**		
						Recall	**Precision**	**F-Measure**
Kato *et al.* [232]	R	D	*20*	R-HOG + SVM	A	*0.911*	*0.777*	*0.838*
				S-HOG + SVM	A	*0.897*	*0.874*	*0.866*
Temerinac-Ott *et al.* [228]	H	M2	80	R-HOG + SVM	A	*N/A*	*N/A*	*0.405-0.551*
				CNN	A	*N/A*	*N/A*	*0.522-0.716*
Gallego *et al.* [210]	H	PAS	*108*	CNN	A	*1.000*	*0.881*	*0.937*
Simon *et al.* [233]	M	HE	*15*	mrcLBP + SVM	A	*0.800*	*0.900*	*0.850*
	R	M1	*25*		A	*0.560-0.730*	*0.750-0.914*	*0.680-0.801*
	H	PAS	*25*		A	*0.761*	*0.917*	*0.832*
Kawazoe *et al.* [198]	H	PAS	*200*	Faster R-CNN	A	*0.919*	*0.931*	*0.925*
		PAM	*200*		A	*0.918*	0.939	*0.928*
		MT	*200*		A	*0.878*	0.915	*0.896*
		Azan	*200*		A	*0.849*	0.904	*0.876*
Proposed	H	PAS	*26*	Faster R-CNN	A	*0.917*	*0.846*	*0.880*
					NS	*0.941*	*0.870*	*0.904*
					S	*0.701*	*0.635*	*0.667*

Stain acronyms: HE - Hematoxylin and eosin, PAS - Periodic acid–Schiff, D - Desmin, M1 - HE/PAS/JS/TRI/CR, M2 - HE/PAS/CD10/SR, JS - Jones silver, TRI - Gömöri's trichrome, CR - Congo red and SR - Sirius red.
Species (Sp) acronyms: H - Human, R - Rat, M - Mouse.
Method acronyms: R-HOG - rectangle-histogram of oriented gradients, S-HOG - segmental-histogram of oriented gradients.
Class acronyms: A - All (no distinction between non-sclerotic and sclerotic glomeruli), NS - non-sclerotic glomeruli and S - sclerotic glomeruli.

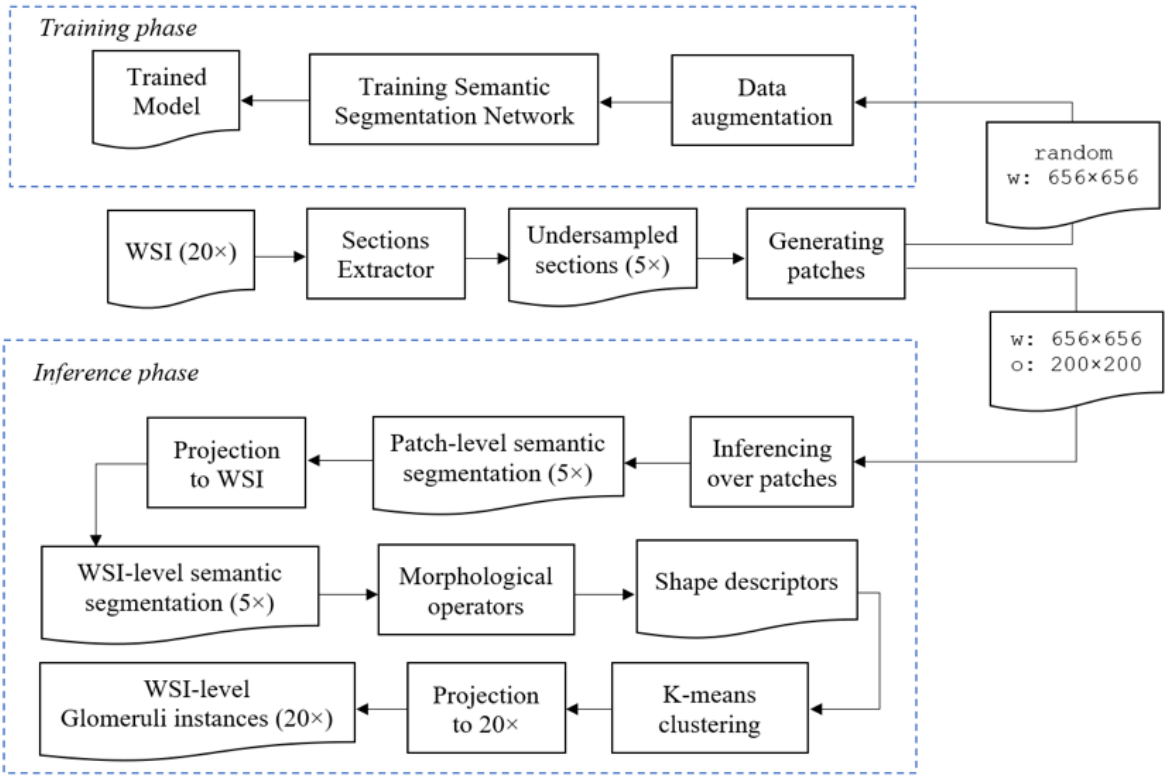


Fig. 3.16 Semantic segmentation work-flow. The top part describes how to perform the training of the model. The bottom part explains how to use the trained model for performing inference and the following morphological and clustering post-processing steps.

3.1.5 A Deep Learning Approach for Glomeruli Semantic Segmentation

A different goal, based on another deep learning technique, has been pursued in this section. To use the finest information provided by the pathologists, that is the exact glomerulus contour, a semantic deep learning approach has been investigated starting from a modified version of SegNet and DeepLab v3+ networks.

The task focuses on the semantic segmentation of glomeruli, but to calculate the Karpinski histological score and to compare the performance with the object detection approach, a subsequent object detection like output is calculated. In particular the semantic segmentation CNN is trained to obtain a pixel-level classification, distinguishing between pixels which belongs to background, sclerotic and non-sclerotic glomeruli; then, the ensemble of classified pixels are turned into object detections regions. The detected and classified regions are finally used for the Karpinski score evaluation.

Table 3.16 Augmentations.

Data Augmentation		
Group	Type	Details
1	Rotate	$\theta = 90$, $p = 0.25$
	Flip left-right	$p = 0.25$
	Flip upside-down	$p = 0.25$
	Resize	$resize \in [0.8, 1.2]$, $p = 0.25$
2	Gaussian Noise	$\sigma \in [0, 0.01]$, $p = 0.1$
	Gaussian Blur	$\sigma \in [0, 0.1]$, $p = 0.1$
	Elastic Deformation	$\sigma \in [2, 5]$, $\alpha \in [100, 300]$, $p = 0.2$
3	HSV shift	$\Delta S \in [-0.1, 0.1]$, $\Delta H \in [-0.1, 0.1]$, $p = 0.5$

3.1.5.1 Work-flow Design and Model Configuration

The semantic segmentation work-flow with final glomerular detector is depicted in Fig. 3.16.

As for object detection (Section 3.1.4), the first step is the segmentation of the tissue sections presents in the slides; the same procedure has been applied. After the rescaling the mean image size is about 2000×2000, that is too big to be processed by the complex CNN models and the available hardware. Patches of size 656×656 are then careful randomly sampled during the train phase avoiding to feed the model with an unbalanced number of samples per class. According to *Stain color variation problem* paragraph from Section 2.4.1.1, Hue-Saturation-Value (HSV) shifts and morphological transformations are considered; in particular, elastic deformation is used in order to generate plausible alterations of glomeruli shapes. The patches are augmented as reported in Table 3.16 and the transformation are applied on-the-fly for each epoch within random ranges so the network always processes slightly different input data, thus reducing the risk of overfitting. Examples of elastic deformation and HSV shift are reported in Fig. 3.17 and Fig. 2.26-2.27, respectively. Regarding the used transformations, they are applied in the order listed in Table 3.16 and, in particular, the group one augmentations are independently performed each with a given probability p, while only one operation is selected each time from group two and the only group three transformation is applied with a given probability. All the transformation are as standard, except that a mirroring padding, instead of zero padding, is used when a resize shrinks the image size. Augmentations which alter the morphology or orientation of images are also executed on the mask.

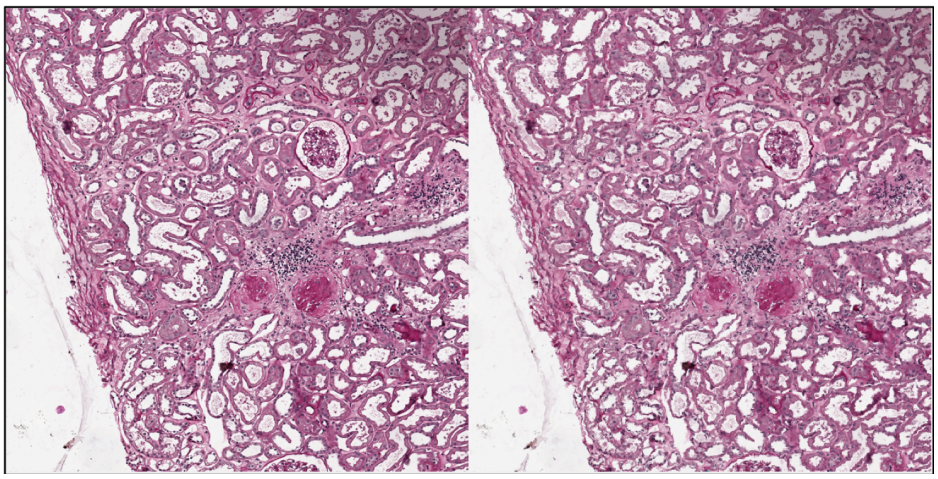


Fig. 3.17 Example of elastic deformation: original image (left), application of elastic deformation with $\sigma = 6.29$, $\alpha = 340$ (right).

During the inference phase, patches of size $656 \times 656 pixels$ and an overlap of 200 pixels are selected. Note that in the training phase, no overlap is needed; this because, the semantic approach have to extract information about pixels and their surrounding and no constrains on regions are required. However, in the inferencing phase, when the approach involves sliding windows, an evaluation on a larger context is suggested [171].

Adapting a semantic segmentation network to perform object detection poses some challenges. The task of semantic segmentation consists of labelling only individual pixels, which mainly captures textural information. Architectures explicitly tailored for object detection, such as Faster R-CNN [472] discusse in Section 3.1.4 or Mask R-CNN [121] that will be discussed in Section 3.1.6, are based on procedures that make use of anchor boxes; semantic ones, instead, just tries to classify pixels individually. To extend the semantic segmentation model into an instance segmentation one, the patch-level pixel classification are projected to the original WSI ($5\times$) to get the WSI-level predicted mask; the whole WSI is then post-processed with morphological operators and clustering algorithms.

Morphological operators are applied only on the obtained binary masks. Firstly, the shapes of objects is smoothed by using a morphological closing with a disk of radius 5 pixels as structuring element, and a morphological flood-fill operation. Then, small objects and spare noisy points are deleted by a sequence of a opening operator with a disk of radius 10 pixels as structuring element, and an area opening operator able to remove connected regions with area below 1000 pixels. Examples are shown in Fig. 3.18, where binary masks are overlapped to the biopsy images for visualization purposes. Masks relative to non-sclerotic and sclerotic glomeruli are green and red coloured, respectively.The sequence of

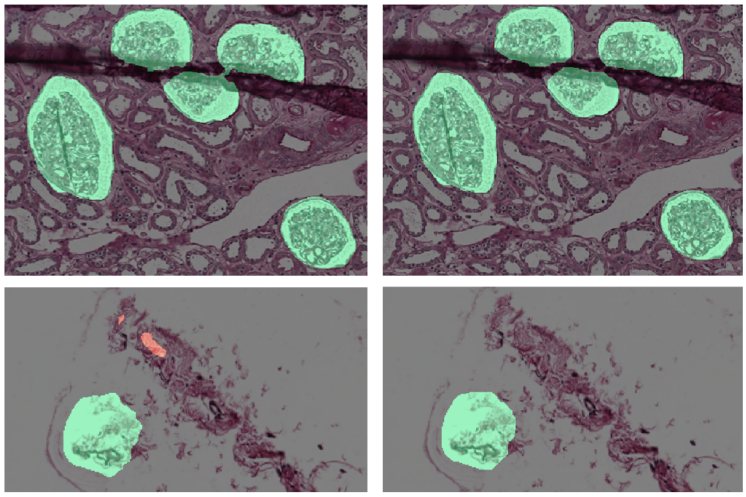


Fig. 3.18 (Left) Semantic Segmentation output. (Right) After Morphological Operators.

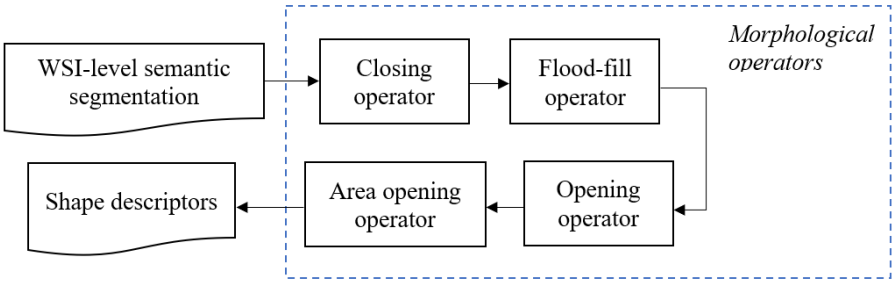
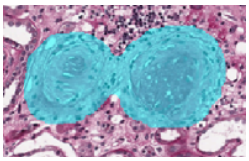


Fig. 3.19 Morphological operators sequence applied to the output masks from the semantic segmentation network. The output of the morphological post-processing is used for calculating shape descriptors in order to eventually perform clustering.

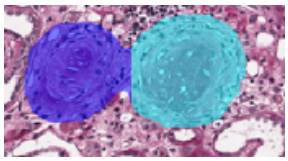
morphological operators used is reported in Fig. 3.19. Lastly, starting from the assumption that individual glomeruli have convex shapes, so that their area is similar to their convex area, the shape information for each of these objects is analysed to understand if there are touching objects that have to be clustered.

The knowledge acquired with the application of classic image processing algorithms (Section 3.1.2), suggested that clustering, could be a viable technique. A K-means clustering is applied with a K number of cluster dependent by the information of the overall detected area. In detail, the difference between the convex area enclosing the entire object (convex hull) and the effective pixels area is computed according to Equation 3.2.

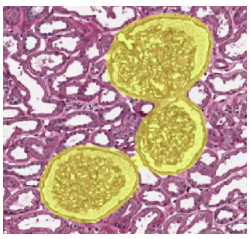
$$\Delta area = convexhull - pixelarea \tag{3.2}$$



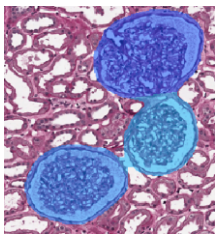
(a) Sclerotic glomeruli before clustering.



(b) After clustering, with $K = 2$.



(c) Non-sclerotic glomeruli before clustering.



(d) After clustering, with $K = 3$.

Fig. 3.20 Examples of K-means clustering for both sclerotic and non-sclerotic glomeruli. The number K of clusters is determined according to Equations 3.2 and 3.3.

Then a K value is empirically selected on the training dataset according to Equation 3.3. Examples of glomeruli before clustering are reported in Fig. 3.20a and Fig. 3.20c. The corresponding images after clustering are shown in Fig. 3.20b and Fig. 3.20d.

$$K = \begin{cases} 1 & \Delta area \leq 900 \\ 2 & 900 < \Delta area \leq 5000 \\ 3 & \Delta area > 5000 \end{cases} \tag{3.3}$$

Finally, the obtained mask is oversampled by a four factor and projected to $20\times$ resolution.

Model configuration. As stated before, the proposed semantic approach has been investigated starting from a modified version of SegNet and DeepLab v3+ networks. In detail, the last layer of both networks has been replaced by a pixel-wise classification layer with the three output classes of interest: background, sclerotic and non-sclerotic glomeruli, and accordingly pixel-wise cross-entropy as loss function and inverse class frequencies as class weights. The training hyperparameters for each model are summarised in Table 3.17. The high batch size difference is due to model complexity differences and hardware constraints (DeepLab v3+ is based on ResNet-18 backbone, that is more lightweight than SegNet).

Table 3.17 Hyperparameters.

Hyperparameter	**SegNet**	**Deeplab v3+**
Optimizer	SGDM	SGDM
Initial Learn Rate	0.001	0.001
Learning Rate Drop Schedule	piecewise	piecewise
Learning Rate Drop Period	10	10
Learning Rate Drop Factor	0.3	0.3
Momentum	0.9	0.9
L2 Regularization	0.005	0.005
Batch Size	1	8
Data Shuffle	every epoch	every epoch
Max Epochs	30	30
Validation Patience	10	10
Validation Frequency	1 per epoch	1 per epoch

3.1.5.2 Results

The proposed approach is divided into two main procedures: the glomeruli semantic segmentation, that is the pixels-level classification, and the glomeruli object detection; for this reason, results about each procedure, will be independently reported in the following paragraphs.

Regarding the the semantic segmentation task two group of metric have been calculated: dataset metrics and class metrics. Dataset metric include all the performance indexes evaluated over the whole dataset: global accuracy, mean accuracy, mean IoU, weighted IoU and mean F-score. The class metrics, instead, includes semantic segmentation metrics calculated for each class: accuracy, IoU and mean F-score. For a detailed description about the cited performance indexes see Section 2.3.3.

Pixel-level dataset metrics for both SegNet and DeepLab v3+ are reported in Table 3.18. The pixel-level class metrics of SegNet and DeepLab v3+ are reported in Table 3.19 and 3.21, respectively. The normalized pixel-level confusion matrices are in Table 3.20 and 3.22. Pixel-level confusion matrices are normalized by row.

Table 3.18 Dataset Metrics.

CNN	**Global Accuracy**	**Mean Accuracy**	**Mean IoU**	**Weighted IoU**	**Mean F-score**
SegNet	*0.98346*	*0.86385*	*0.71352*	*0.97156*	*0.81784*
Deeplab v3+	*0.99179*	*0.76884*	*0.72873*	*0.98434*	*0.84614*

Table 3.19 Class Metrics SegNet.

Class	**Accuracy**	**IoU**	**Mean F-score**
Sclerotic	*0.68594*	*0.49215*	*0.69686*
Non-Sclerotic	*0.91925*	*0.66546*	*0.83239*
Background	*0.98636*	*0.98294*	*0.99243*

Table 3.20 Normalized pixel-level Confusion Matrix SegNet.

		Prediction		
		Sclerotic	Non-Sclerotic	Background
Ground Truth	Sclerotic	*68.59%*	*0.44%*	*30.97%*
	Non-Sclerotic	*0.00%*	*91.93%*	*8.07%*
	Background	*0.10%*	*1.26%*	*98.64%*

Table 3.21 Class Metrics Deeplab v3+.

Class	**Accuracy**	**IoU**	**Mean F-score**
Background	*0.99690*	0.99172	0.96684
Non-Sclerotic	*0.88199*	*0.80872*	*0.93306*
Sclerotic	*0.42764*	*0.38574*	*0.63852*

Table 3.22 Normalized pixel-level Confusion Matrix Deeplab v3+.

		Prediction		
		Sclerotic	Non-Sclerotic	Background
Ground Truth	Sclerotic	*42.76%*	*6.67%*	*50.57%*
	Non-Sclerotic	*0.02%*	*88.20%*	*11.78%*
	Background	*0.03%*	*0.28%*	*99.69%*

The best results on non-sclerotic glomeruli have been obtained using DeepLab v3+, whilst for sclerotic glomeruli the best model was SegNet. An example of the output of the semantic segmentation framework is depicted in Fig. 3.21.

Object detection extension. For the object detection task, confusion matrices are calculated assuming that a true positive match between predicted and ground truth masks has

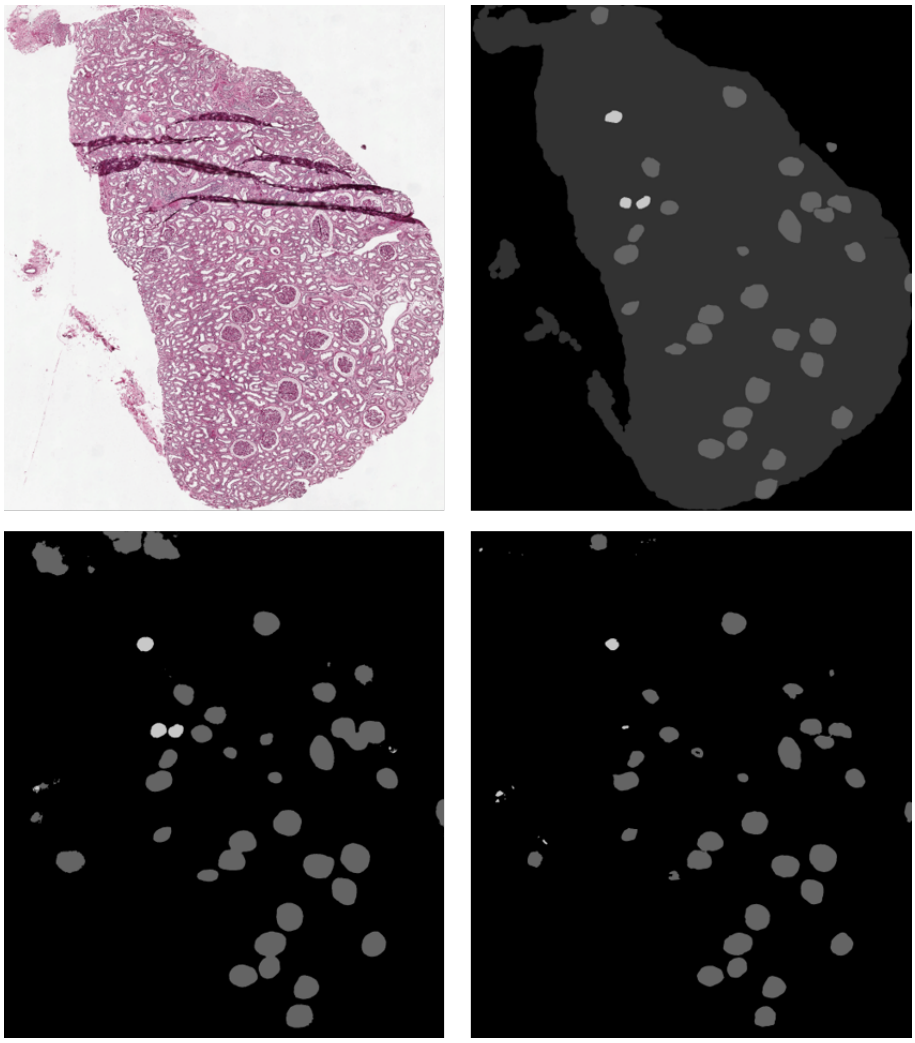


Fig. 3.21 Top Left: original image. Top Right: ground truth. Bottom Left: SegNet prediction. Bottom Right: DeepLab v3+ prediction. Sclerotic glomeruli and non-sclerotic ones are white and gray colored, respectively.

pixel-wise IoU of at least 0.2. Besides confusion matrices, the metrics used for assessing the results of the object detection task are: precision, recall and F-score.

The object detection confusion matrices for SegNet and DeepLab v3+ are reported in Table 3.23 and 3.24, respectively.

Analysing the results is possible to state that the proposed approach allowed to obtain high performances both at pixel and object detection level. The semantic segmentation achieved mean F-score higher than 0.81 and Weighted IoU higher than 0.97 for both SegNet and Deeplab v3+ approaches; the glomeruli detection achieved 0.924 as best F-score for non-sclerotic glomeruli and 0.730 as best F-score for sclerotic glomeruli. Analysed literature

Table 3.23 Object Detection Confusion Matrix SegNet

		Prediction		
		Sclerotic	Non-Sclerotic	Background
Ground Truth	Sclerotic	*58*	*1*	*28*
	Non-Sclerotic	*0*	*436*	*56*
	Background	*14*	*86*	–

Table 3.24 Object Detection Confusion Matrix Deeplab v3+

		Prediction		
		Sclerotic	Non-Sclerotic	Background
Ground Truth	Sclerotic	*41*	*7*	*39*
	Non-Sclerotic	*0*	*449*	*43*
	Background	*1*	*24*	–

Table 3.25 Performance Comparison for Detection Metrics

Author	**Model**	**Class**	**Recall**	**Precision**	**F-score**
Marsh *et al.* [231]	FCN + Blob-Detection	Non-Sclerotic	*0.885*	*0.813*	*0.848*
		Sclerotic	*0.698*	*0.607*	*0.649*
Proposed	SegNet	Non-Sclerotic	*0.886*	*0.834*	*0.859*
		Sclerotic	*0.667*	*0.806*	*0.730*
	DeepLab v3+	Non-Sclerotic	*0.913*	*0.935*	*0.924*
		Sclerotic	*0.471*	*0.976*	*0.636*

account three main works that face the problem of glomerular classification. Ginley *et al.* considered the glomerular assessment for patients affected by diabetic nephropathy but not for transplantation purposes [230]. Hermsen *et al.* considered many tissue classes, but the number of sclerotic glomeruli in their datasets is too small for a comparison with the proposed method [242]. Marsh *et al.* considered the problem of global glomerulosclerosis from kidney transplant biopsies with haematoxylin and eosin (HE) stain [231]. The performance comparison between the proposed methods and Marsh *et al.* paper is reported in Table 3.25 and is possible to affirm that obtained results show improvements over the literature. The SegNet based model obtained a better F-score for both the glomeruli classes.

The DeepLab v3+ based model obtained a better F-score for non-sclerotic glomeruli and a slightly worse F-score for sclerotic glomeruli. In conclusion, it is possible to state that CNNs for semantic segmentation are a viable approach for the purpose of glomerular segmentation and classification, allowing to obtain a reliable estimate of the global glomerulosclerosis. The results were validated by the renal pathologists which assessed the reliability of the proposed workflow; the applied methodology constitutes a milestone in the creation of a CAD system for the renal transplant assessment, easing pathologists in accomplishing the laborious task of evaluating the eligibility of a kidney for transplantation and providing a rapid and accurate result.

3.1.6 A Deep Learning Approach for Glomeruli Istance Segmentation

The application of classic image processing algorithm and deep learning for object detection and semantic segmentation, achieved good result in the task of glomeruli processing and in the evaluation of global glomerulosclerosis. A last deep learning application is discussed in this section: instance segmentation. In computer vision, the end-to-end instance segmentation methods (e.g., Mask-RCNN [121]) have shown their advantages on the detect-then-segment approaches, by performing detection and segmentation tasks simultaneously. As a result, the end-to-end Mask-RCNN approach became the standard method in recent glomerular segmentation studies [180]. The semantic segmentation networks, as the one proposed before, can guarantee very high pixel-level results, but they may perform worse in the object detection task, if compared to specialized architectures. In this section, will be proposed a deep learning framework based on Mask R-CNN for glomerular detection and classification with an end-to-end instance segmentation approach. in detail, will be evaluated the possibility to train an end-to-end instance segmentation neural network, by exploiting Mask R-CNN, trying to learn all the required features in a unified process, and reducing the need of post processing operations; furthermore, a variant of the standard Non-Maximum Suppression algorithm (Algorithm 1), named Non-Maximum-Area Suppression (NMAS), is proposed to improve the performances in the sliding window step.

3.1.6.1 Work-flow Design and Model Configuration

The full pipeline used for the instance segmentation is reported in Fig. 3.22. As for other approaches, during the training phase, random patches of 1024×1024 pixels are sampled and then, for each epoch, on-the-fly randomly augmented (random selection of none, one or two transformations reported in Table 3.26).

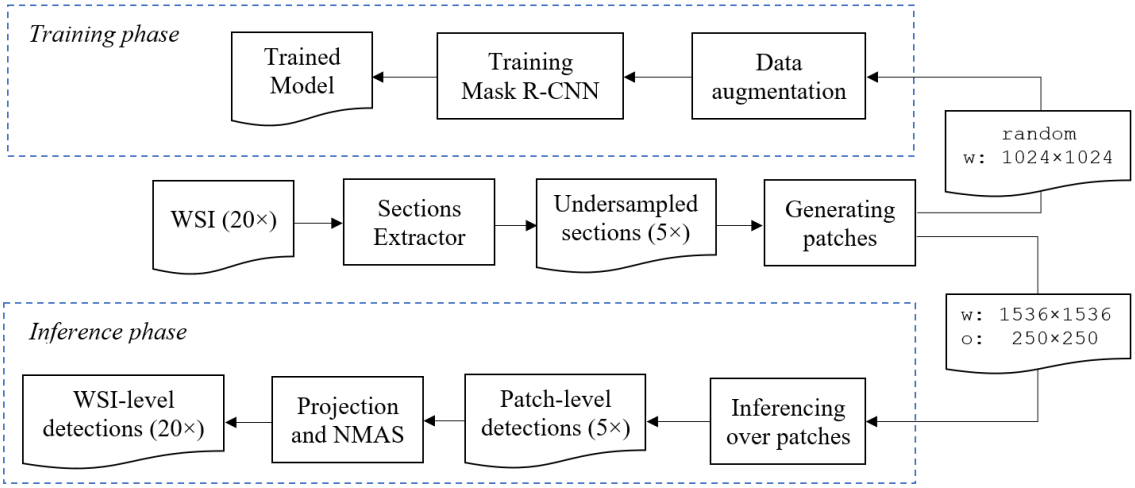


Fig. 3.22 Instance segmentation work-flow based on Mask R-CNN model. The top part describes how to perform the training of the model, exploiting the train-validation set (19 whole slide images). The bottom part explains how to use the trained model for performing inference on the test set sections (7 whole slide images). In the inference phase the proposed Non-Maximum-Area Suppression (NMAS) algorithm is used.

During the inferencing phase, instead, thanks to the less memory requirements, larger windows with size 1536×1536 and overlap of 250×250 are used. To overcome the NMS algorithm (Algorithm 1), low capability to handle overlapped bounding boxes, NMAS algorithm (Algorithm 3) is used instead of it during the projection of the patch-level detections to WSI-level, whit appreciable improvement. It allows to join the information of the areas related to the involved bounding boxes and their confidence scores; the new parameters $f_j = w_j h_j s_j^2$ has been introduced considering both the area of the bounding box ($w_j h_j$) and the square of the confidence (s_j^2) to penalise its contribution (s_j ranges in $[0,1]$). Another improvement comes from the use of IoM jointly with IoU to detect overlapping boxes. IoM easily allows to recognize bounding boxes mainly contained in other ones, that is common in overlapping sliding window approaches.

In Fig. 3.23 is possible to appreciate an example of sub-optimal bounding box and the improvement obtained thanks to NMAS.

Regarding the model, ResNet-50 has been used as backbone, since it allows good feature extraction despite it is lighter than ResNet-101 [130]. The training procedure starts from the model pretrained on the COCO dataset and involved only the first part of the network for 20 epochs; then ResNet stage four and layers above were trained for 40 epochs and, finally, all the layers of the network were trained for 40 epochs lowering the learning rate to 0.0001. For reproducibility the others hyperparameters are reported in Table 3.27 and refer

Table 3.26 Augmentations for Mask R-CNN approach.

Data Augmentation	
Type	**Details**
Flip upside-down	$P(flip_{ud}) = 0.5$
Flip left-right	$P(flip_{lr}) = 0.5$
Rotate	$\theta \in \{90°, 180°, 270°\}$
Multiply	$\alpha \in [0.8, 1.1]$
Gaussian Blur	$\sigma \in [0, 0.1]$

Algorithm 3: Non-Maximum-Area Suppression (NMAS).

input : $B_i = b_1, ..., b_{N_i}$, the N_i initial detections
$b_j = (x_j, y_j, w_j, h_j), j = 1, ..., N_i$
$S_i = s_{i_1}, ..., s_{i_{N_i}}$, the N_i initial scores
T_{iou}, the NMAS threshold on IoU
T_{iom}, the NMAS threshold on IoM

output : $B_o = b_1, ..., b_{N_o}$, the $N_o \leq N_i$ final detections
$S_o = s_{o_1}, ..., s_{o_{N_o}}$, the $N_o \leq N_i$ final scores

$B_o = \{\}$;
$S_o = (w_1 h_1 s_{i_1}^2, w_2 h_2 s_{i_2}^2, ..., w_{N_i} h_{N_i} s_{i_{N_i}}^2)$;
while B_i *is not empty* **do**
  $m = argmax(S_o)$;
  $B_o = B_o \cup \{b_m\}$;
  $B_i = B_i \setminus \{b_m\}$;
  while $b_j \in B_i$ **do**
    if $iou(b_m, b_j) \geq T_{iou} \vee iom(b_m, b_j) \geq T_{iom}$ **then**
      $B_i = B_i \setminus \{b_j\}$;
      $S_o = S_o \setminus \{s_j\}$;
    end
  end
end

to Mask R-CNN implementation developed by Waleed Abdulla from Matterport, Inc. [473] and published under the MIT License[1], then more details can be found in the documentation, if needed. The inference configuration is slightly different, with `IMAGE_RESIZE_MODE = "pad64"` and `RPN_NMS_THRESHOLD = 0.7`.

[1] https://opensource.org/licenses/MIT

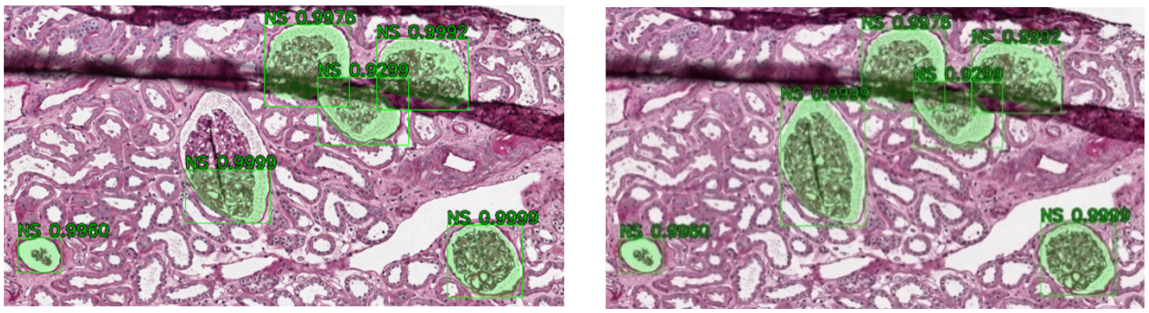


Fig. 3.23 Mask R-CNN predictions, after removal of overlapping bounding boxes with the two considered algorithms: Non-Maximum Suppression (Left) and Non-Maximum-Area Suppression (Right).

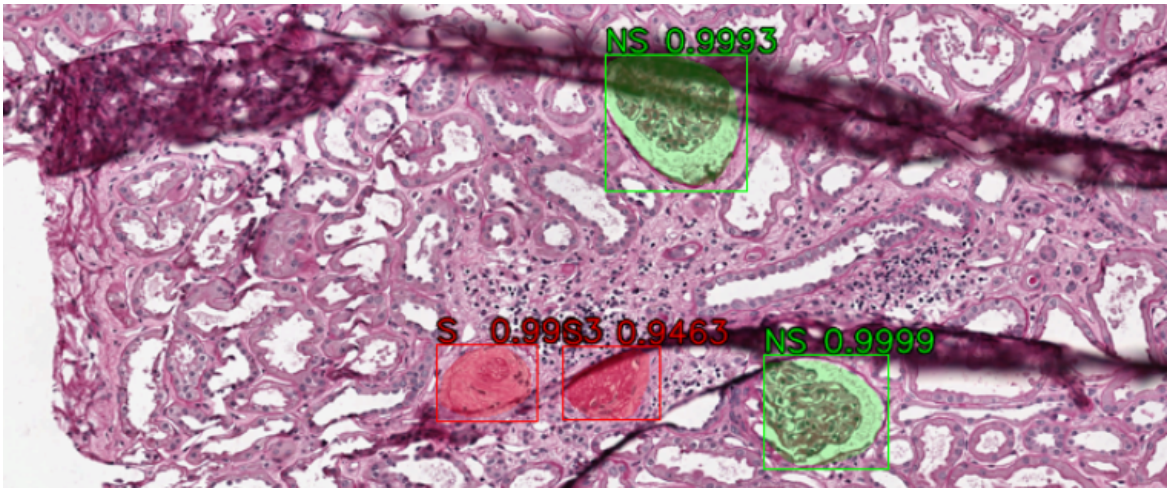


Fig. 3.24 Patch-level detection with Mask R-CNN.

Examples of patch-level and final WSI-level detections can be seen in Fig. 3.24 and Fig. 3.25, respectively. Images clearly show both masks and bounding boxes as expected, since Mask R-CNN allows to solve instance segmentation and object detection tasks at the same time.

3.1.6.2 Results

The results obtained with the Mask R-CNN based approach are reported in Table 3.28 and Table 3.29. Using NMAS instead of NMS for suppressing overlapped bounding boxes leads to an improvement of mAP from 0.881 to 0.902, and of F-measure for non-sclerotic glomeruli from 0.917 to 0.925.

Also in this case, in order to compare the clinical validity of the workflow performance, the corresponding Karpinski score is computed and compared with that of expert pathologists.

Table 3.27 Hyperparameters tuning for Mask R-CNN based detector.

Training Configuration	
Hyperparameter	**Value**
BACKBONE	*resnet50*
RPN_ANCHOR_SCALES	*(32, 96, 160, 200, 256)*
RPN_ANCHOR_RATIOS	*[0.5, 1, 2]*
POST_NMS_ROIS_TRAINING	*800*
POST_NMS_ROIS_INFERENCE	*1600*
RPN_NMS_THRESHOLD	*0.8*
RPN_TRAIN_ANCHORS_PER_IMAGE	*64*
MEAN_PIXEL	*[218.85, 198.25, 207.18]*
MINI_MASK_SHAPE	*(56, 56)*
TRAIN_ROIS_PER_IMAGE	*128*
IMAGE_RESIZE_MODE	*crop*
IMAGE_MIN_DIM	*1024*
IMAGE_MAX_DIM	*1024*
LEARNING_RATE	*0.001*
LEARNING_MOMENTUM	*0.9*
WEIGHT_DECAY	*0.0001*
GRADIENT_CLIP_NORM	*5.0*

Hyperparameters nomenclature refers to Mask R-CNN implementation developed by Waleed Abdulla from Matterport, Inc. [473].

Table 3.28 Object detection confusion matrix with the proposed Mask R-CNN workflow.

		Prediction		
		Sclerotic	Non-Sclerotic	Background
Ground Truth	Sclerotic	*61*	*8*	*18*
	Non-Sclerotic	*0*	*470*	*22*
	Background	*9*	*46*	–

As described in Section 2.4.1.2 and reported in Table 2.7 (the involved table section is

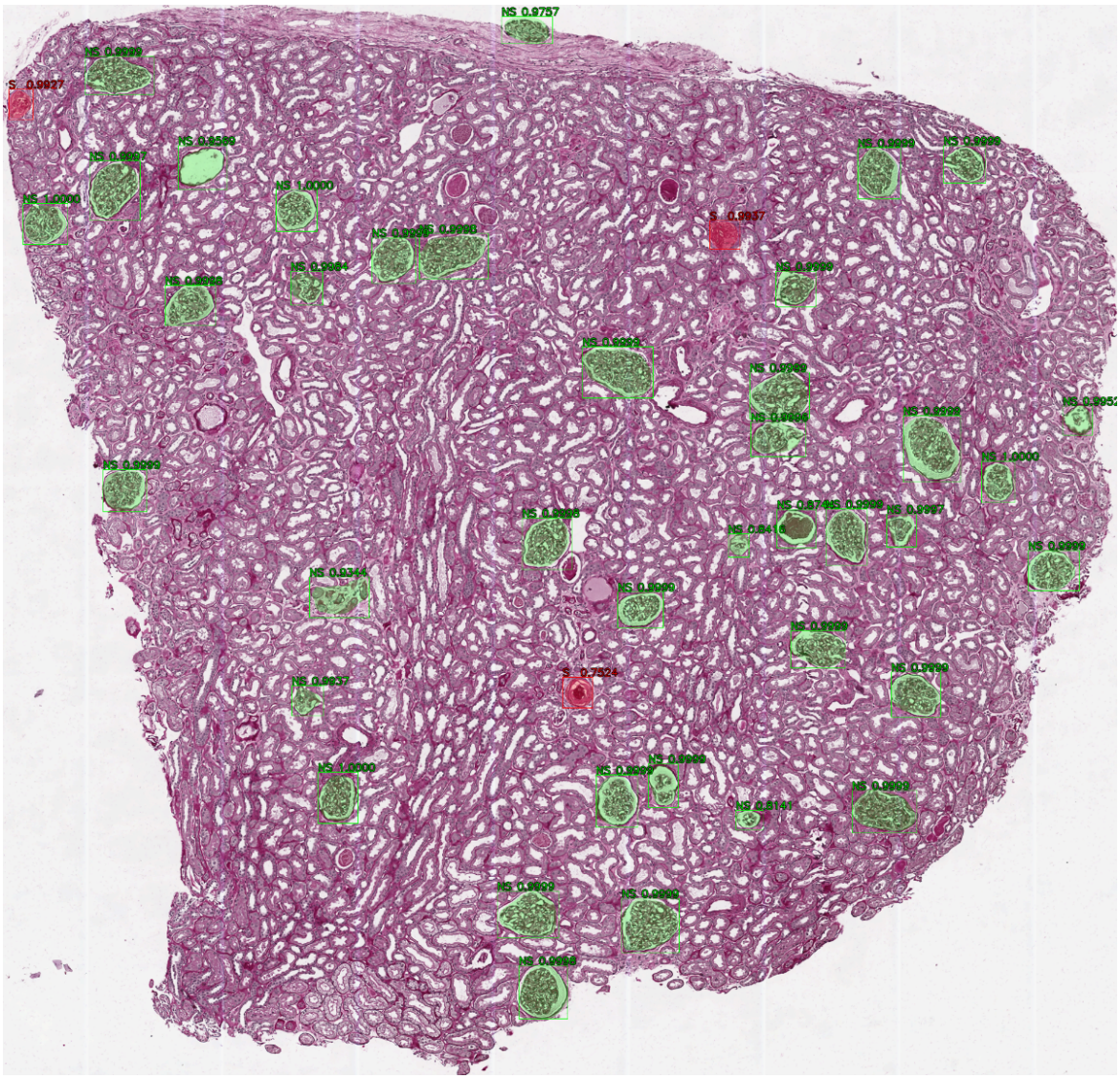


Fig. 3.25 WSI-level detection with Mask R-CNN. Overlapping bounding boxes have been eliminated with the proposed Non-Maximum-Area Suppression algorithm.

Table 3.29 Detection metrics with the proposed Mask R-CNN workflow.

Class	Recall	Precision	F-score
Non-sclerotic	*0.955*	*0.897*	*0.925*
Sclerotic	*0.701*	*0.871*	*0.777*

reported in Table 3.13) the ratio is computed as the number of sclerosed glomeruli divided by the overall number of glomeruli ($Ratio = \frac{S}{S+NS}$), then the score is computed as following: 0, if there are no globally sclerosed glomeruli; 1, if there is $< 20\%$ global glomerulosclerosis; 2, if there is $20-50\%$ global glomerulosclerosis; 3, if there is $> 50\%$ global glomerulosclerosis.

The comparison between Mask R-CNN output and ground truth is shown in Table 3.30; for completeness and ease the comparison between the object detection approach (Section 3.1.4), results reported in Table 3.14 are replicated in Table 3.30 (only Faster R-CNN and Mask R-CNN are compare here because they are natively built for object detection). The results shown that Mask R-CNN approach makes only three errors in assessing the Karpinski score: one time it gives a score of a score of 0 instead of a score of 1; two times it gives a score of 1 instead of a score of 2; Faster R-CNN, instead, makes five errors in assessing the Karpiski score: four times it gives a score of 1 instead of 2, and one time it gives a score of 2 instead of 1.

Table 3.30 Karpinski Score results on hold-out test set. Comparison between Faster R-CNN, Mask R-CNN and ground truth annotations. NS stands for non-sclerotic, S stands for sclerotic. Score belongs to the range $[0-3]$.

Donor	**Kidney**	**Section**	**Mask R-CNN**				**Faster R-CNN**				**Ground Truth**			
			NS	**S**	**Ratio**	**Score**	**NS**	**S**	**Ratio**	**Score**	**NS**	**S**	**Ratio**	**Score**
1	Left	*1*	*30*	*3*	*0.09*	*1*	*31*	*3*	*0.09*	*1*	*30*	*3*	*0.09*	*1*
		2	*30*	*2*	*0.06*	*1*	*32*	*2*	*0.06*	*1*	*30*	*2*	*0.06*	*1*
		3	*31*	*4*	*0.11*	*1*	*29*	*5*	*0.15*	*1*	*28*	*4*	*0.13*	*1*
		4	*29*	*5*	*0.15*	*1*	*31*	*5*	*0.14*	*1*	*25*	*4*	*0.14*	*1*
		5	*32*	*0*	*0.00*	*0*	*30*	*1*	*0.03*	*1*	*31*	*1*	*0.03*	*1*
		6	*31*	*1*	*0.03*	*1*	*35*	*3*	*0.08*	*1*	*31*	*1*	*0.03*	*1*
2	Right	*1*	*11*	*5*	*0.31*	*2*	*9*	*8*	*0.47*	*2*	*10*	*5*	*0.33*	*2*
3	Right	*1*	*41*	*1*	*0.02*	*1*	*40*	*8*	*0.17*	*1*	*38*	*2*	*0.05*	*1*
	Left	*1*	*39*	*3*	*0.07*	*1*	*38*	*3*	*0.07*	*1*	*41*	*4*	*0.09*	*1*
4	Right	*1*	*19*	*4*	*0.17*	*1*	*23*	*7*	*0.23*	*2*	*17*	*5*	*0.23*	*2*
		2	*26*	*3*	*0.10*	*1*	*29*	*4*	*0.12*	*1*	*25*	*3*	*0.11*	*1*
		3	*30*	*2*	*0.06*	*1*	*29*	*5*	*0.15*	*1*	*25*	*3*	*0.11*	*1*
		4	*29*	*5*	*0.15*	*1*	*28*	*9*	*0.24*	*2*	*25*	*5*	*0.17*	*1*
5	Right	*1*	*22*	*4*	*0.15*	*1*	*23*	*3*	*0.12*	*1*	*22*	*4*	*0.15*	*1*
		2	*30*	*5*	*0.14*	*1*	*27*	*3*	*0.10*	*1*	*28*	*5*	*0.15*	*1*
6	Right	*1*	*14*	*4*	*0.22*	*2*	*14*	*3*	*0.18*	*1*	*13*	*6*	*0.32*	*2*
		2	*14*	*4*	*0.22*	*2*	*13*	*3*	*0.19*	*1*	*13*	*6*	*0.32*	*2*
		3	*13*	*4*	*0.24*	*2*	*13*	*3*	*0.19*	*1*	*14*	*5*	*0.26*	*2*
		4	*14*	*2*	*0.13*	*1*	*12*	*1*	*0.08*	*1*	*12*	*2*	*0.14*	*1*
		5	*17*	*5*	*0.23*	*2*	*16*	*4*	*0.20*	*2*	*14*	*6*	*0.30*	*2*
		6	*19*	*4*	*0.17*	*1*	*20*	*4*	*0.17*	*1*	*17*	*10*	*0.37*	*2*

The full comparison among all the deep learning approach analysed in this chapter and the recent research works about the task of glomerular detection is reported in Table 3.31. The proposed model performs well in the detection of non-sclerotic glomeruli, with very high recall and precision values, but metrics for sclerotic glomeruli suffer from a higher number of false negatives.

Table 3.31 Comparison with literature extending the one proposed by Kawazoe *et al.* [198]

Author	Sp	Stain	WSIs	Method	Class	Performances		
						Recall	**Precision**	**F-Measure**
Kato *et al.* [232]	R	D	*20*	R-HOG + SVM	A	*0.911*	*0.777*	*0.838*
				S-HOG + SVM	A	*0.897*	*0.874*	*0.866*
Temerinac-Ott *et al.* [228]	H	M2	*80*	R-HOG + SVM	A	*N/A*	*N/A*	*0.405-0.551*
				CNN	A	*N/A*	*N/A*	*0.522-0.716*
Gallego *et al.* [210]	H	PAS	*108*	CNN	A	*1.000*	*0.881*	*0.937*
Simon *et al.* [233]	M	HE	*15*	mrcLBP + SVM	A	*0.800*	*0.900*	*0.850*
	R	M1	*25*		A	*0.560-0.730*	*0.750-0.914*	*0.680-0.801*
	H	PAS	*25*		A	*0.761*	*0.917*	*0.832*
Kawazoe *et al.* [198]	H	PAS	*200*	Faster R-CNN	A	*0.919*	*0.931*	*0.925*
		PAM	*200*		A	*0.918*	*0.939*	*0.928*
		MT	*200*		A	*0.878*	*0.915*	*0.896*
		Azan	*200*		A	*0.849*	*0.904*	*0.876*
Marsh *et al.* [231]	H	HE	*48*	FCN + BLOB	NS	*0.885*	*0.813*	*0.848*
					S	*0.698*	*0.607*	*0.649*
Proposed Semantic Segmentation (Section 3.1.5)	H	PAS	*26*	SegNet	A	*0.855*	*0.832*	*0.843*
					NS	*0.886*	*0.834*	*0.859*
					S	*0.667*	*0.806*	*0.730*
				DeepLab v3+	A	*0.858*	*0.952*	*0.903*
					NS	*0.913*	*0.935*	*0.924*
					S	*0.471*	*0.976*	*0.636*
Proposed Object Detection (Section 3.1.4)	H	PAS	*26*	Faster R-CNN	A	*0.917*	*0.846*	*0.880*
					NS	*0.941*	*0.870*	*0.904*
					S	*0.701*	*0.635*	*0.667*
Proposed Instance Segmentation	H	PAS	*26*	Mask R-CNN	A	*0.931*	*0.907*	*0.919*
					NS	*0.955*	*0.897*	*0.925*
					S	*0.701*	*0.871*	*0.777*

Stain acronyms: HE - Hematoxylin and eosin, PAS - Periodic acid–Schiff, D - Desmin, M1 - HE/PAS/JS/TRI/CR, M2 - HE/PAS/CD10/SR, JS - Jones silver, TRI - Gömöri's trichrome, CR - Congo red and SR - Sirius red.
Species (Sp) acronyms: H - Human, R - Rat, M - Mouse.
Method acronyms: R-HOG - rectangle-histogram of oriented gradients, S-HOG - segmental-histogram of oriented gradients.
Class acronyms: A - All (no distinction between non-sclerotic and sclerotic glomeruli), NS - non-sclerotic glomeruli and S - sclerotic glomeruli.

3.1.7 Conclusion About Deep Learning Approaches for Glomerulosclerosis Evaluation

Analysing the test performed for the deep learning approaches, it is possible to observe that glomerular detection and classification should be approached as an instance segmentation tasks. Even if object detection approaches can guarantee respectable results, they do not exploit the full mask information in the dataset. Semantic segmentation approaches allow to obtain good results too, but they are slightly worse than instance segmentation ones. Indeed, training a CNN which classifies at pixel-level in a detection task is a less powerful method. Difficulties that occur with semantic segmentation networks include presence of noisy points in the output and lack of distinction between touching objects. Semantic segmentation networks can principally exploit texture information but are less capable to understand concepts as shapes, thus working worse on detection task compared to specialized architectures. Nevertheless, Marsh *et al.* used fully convolutional network, together with BLOB detection as post-processing of semantic segmentation network output, to measure global glomerulosclerosis from kidney biopsies [231]. The proposed Mask R-CNN approach outperforms the one of Marsh *et al.* , improving the F-score for healthy glomeruli from 0.848 to 0.925 and the F-score for sclerosed glomeruli from 0.649 to 0.777. An important reason for these better performances may lie in the choice of a better model, relying on an instance segmentation network instead of a semantic segmentation one. Anyway, it has to be noted that Marsh *et al.* dealt with HE stained biopsies, whereas the dataset adopted for the experiments conducted and described in this thesis, is made up by PAS stained biopsies, which can be a better staining for glomerular recognition tasks. Although, it has to be demonstrated that CNNs work consistently better on PAS compared to HE. It is also worth noting that in the Marsh *et al.* work the unbalancing ratio is lesser, being $3.44 : 1$ compared to $5.48 : 1$, thus allowing a smoother training process for the under-represented class.

Other papers found in literature do not address the task of determining glomerulosclerosis, but focus only on glomerular detection. Though this is a simpler task, they are included in the comparison (Table 3.31), considering healthy and sclerotic glomeruli as a single class. The authors using classical machine learning approaches, obtaining worse results than the proposed deep learning ones [232, 233]. Temerinac-Ott *et al.* compared a machine learning approach, based on HOG (histogram of oriented gradients [240]) feature extraction and an SVM classifier, with a deep learning one with CNN; both obtained lower performances than the proposed end-to-end instance segmentation framework [228]. Gallego *et al.* exploited a CNN for classifying if each patch is a glomerulus or not in a sliding window fashion [210].

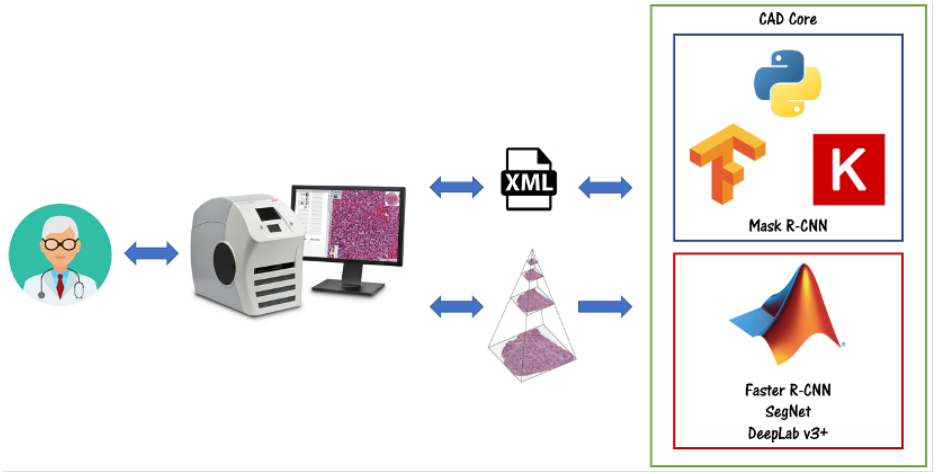


Fig. 3.26 CAD overview. Physicians can annotate WSIs and visualize processing results by using Aperio ImageScope software.

Altough this may look like a more naive approach, if compared with the use of a detector from the R-CNN family (which can also reduce the problem of redundant computation across neighbouring patches), the results obtained in the paper are quite impressive, with a recall of 1. However, it is worth noting that Gallego *et al.* considered only glomeruli with area of at least 200×200 pixel ($> 100\,\mu$m of diameter), whereas glomeruli considered in the dataset used for this thesis have no size constraints, and many of the obtained false negatives are small glomeruli. Furthermore, a precise mask for each glomerulus is found, while Gallego *et al.* can only determine coarse masks composed by the union of rectangular patches. Kawazoe *et al.* used Faster R-CNN for the glomerular detection task, obtaining results comparable with the proposed Mask R-CNN approach, with an F-score of 0.925 (greater than the obtained one of 0.919) [198]. The possibility to use a larger training dataset (200 WSIs instead of 26) can explain why they can get comparable (or even slightly better) results even with a less powerful model. As already noted, discussed Faster R-CNN model performs worse than Mask R-CNN one. In conlusion, at the moment, the proposed frameworks allow to get a reliable estimate of global glomerulosclerosis; then, the pathologists can beneficiate from glomeruli annotations provided by the proposed CAD through an XML interface with the commonly used Aperio ImageScope software, easing the burden of the manual annotation (Fig. 3.26). In the future it could be extended to other kidney biopsies analysis tasks, consenting to define the complete Karpinski histological score.

3.2 Deep Learning in Radiology: Autosomal Dominant Polycystic Kidney Disease Case Study

As discussed in Section 2.4.2.1, innovative approaches based on deep learning strategies have been introduced in recent years for the classification and segmentation of radiologic images. For assessing kidneys growth researchers suggest that MR should be preferred to other imaging techniques [270]. Different research works allowed the estimation of TKV starting from CT images thanks to the higher availability of the acquisition devices and the more accurate and reliable measurement of TKV and cysts volume. On the other side, CT protocols for ADPKD are always contrast-enhanced using a contrast medium harmful for the health of the patient under examination; also, CT exposes the patients to ionising radiations. On these premises, the automatic or semi-automatic segmentation of images from MR acquisitions for improving the TKV estimation capabilities should be further investigated to improve the state-of-the-art performances.

Starting from a preliminary work performed on a small set of patients [297], in the following sections two different approaches based on DL architectures for the automatic segmentation of polycystic kidneys starting from MR acquisitions are presented. The two approaches, based on object detection and semantic segmentation, lead to a fully-automated pipeline for ADPKD evaluation from MR images. Regarding of the application of object detection for ADPKD, due to the presence of cysts in the organs near the kidneys and due to the very similar structures of the surrounding areas, a pre-selection of the region of interest was considered to improve a subsequent semantic segmentation step.

3.2.1 Materials

Data Description Analyses were conducted on MR images acquired from February to July 2017; 18 patients affected by ADPKD (mean age 31.25 ± 15.52 years) underwent magnetic resonance examinations for assessing the TKV. The acquisition protocol was carried out by the physicians of the Department of Emergency and Organ Transplantations (DETO) of the Bari University Hospital. Examinations and images acquisition were performed on a 1.5 Tesla MR device (Achieva, Philips Medical Systems, Best, The Netherlands) by using a four-channel breast coil. The protocol did not use contrast material intravenous injection and consisted of:

- Transverse and Coronal Short-TI Inversion Recovery (STIR) Turbo-Spin-Echo (TSE) sequences ($TR/TE/TI = 3.800/60/165ms$, field of view $(FOV) = 250 \times 450mm$

(AP x RL), matrix 168×300, 50 slices with $3mm$ slice thickness and without gaps, 3 averages, turbo factor 23, resulting in a voxel size of $1.5 \times 1.5 \times 3.0mm^3$; sequence duration of $4.03min$);

- Transverse and Coronal T2-weighted TSE ($TR/TE = 6.300/130ms$, $FOV = 250 \times 450mm$ (AP x RL), matrix 336×600, 50 slices with $3mm$ slice thickness and without gaps, 3 averages, turbo factor 59, SENSE factor 1.7, resulting in a voxel size of $0.75 \times 0.75 \times 3.0mm^3$; sequence duration of $3.09min$);
- Three-Dimensional (3D) T1-Weighted High Resolution Isotropic Volume Examination (THRIVE) sequence ($TR/TE = 4.4/2.0ms$, $FOV = 250 \times 450 \times 150mm(APxRLxFH)$, matrix 168×300, 100 slices with $1.5mm$ slice thickness, turbo factor 50, SENSE factor 1.6, data acquisition time of $1min30s$).

Dataset Creation Although patients undergo the complete protocol, only the coronal T2-weighted TSE sequence was considered. In order to have the segmentation ground truth for all the acquired images, a dedicated framework allowed radiologists to manually draw contours.

3.2.2 ADPKD Study Case: Object Detection

According to Section 2.2.2, DL architectures used for object detection aim to find the location of specific items in images or videos and it is a well-established process in literature employed in different fields [362, 474]. The CNNs for object detection are commonly based on static algorithms, such as EdgeBoxes [475] or Selective Search [156], for the region proposal steps; traditional Region-CNN and Fast R-CNN are the most frequently used techniques [123, 476]. Other applications, such as Faster R-CNN [472], make use of a specific CNN to address the region proposal mechanism, thus joining the region proposal steps in the learning procedure.

Several networks based on Fast R-CNN were investigated to detect areas containing kidneys. To build a dataset suitable for this purpose, the manual contour of each kidney was enclosed in a rectangular bounding box. The considered CNNs were based on layer blocks composed by the concatenation of convolutional and ReLu layers; a max-pooling layer was used between each block for sub-sampling (size $[3 \times 3]$ and stride $[2 \times 2]$). Two fully-connected layers manage the output detection step. The obtained best configurations are reported in Table 3.32.

To help the results evaluation, the Precision-Recall and the Miss Rate plots, that are the precision values varying recall values and the miss rate varying the FP per image, were

Table 3.32 Configurations designed and tested for the object detection CNN. Each layer is a sequence of a convolutional and a ReLu layer.

Network ID	**Number of layers per block**	**Number of convolutional filters per layer**	**Learner**
R-CNN-1	*[3 3]*	*[32 32]*	*SGDM*
R-CNN-2	*[1 1]*	*[16 32]*	*SGDM*
R-CNN-3	*[3 3]*	*[64 32]*	*SGDM*

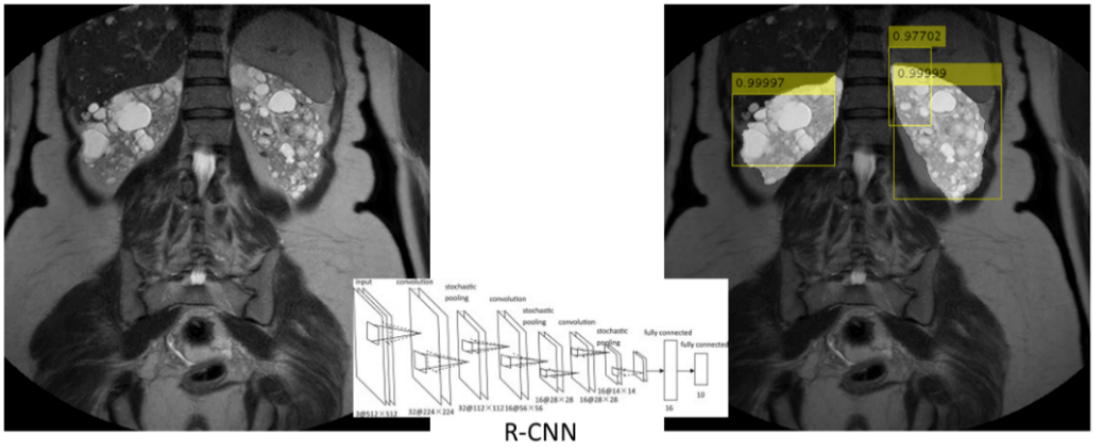
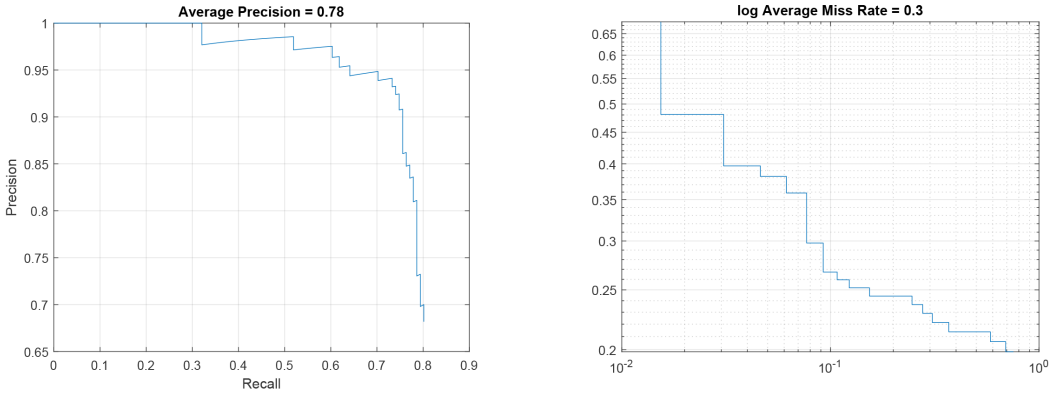


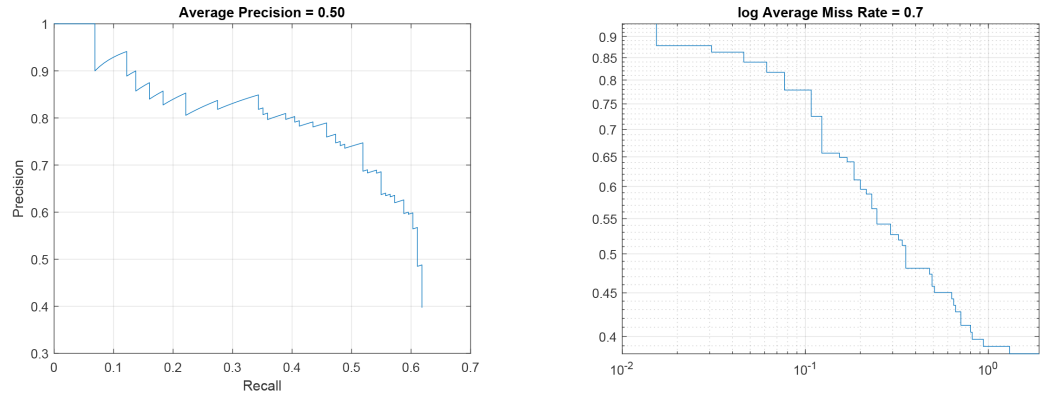
Fig. 3.27 Example results for object detection R-CNN. Input image (left); detected ROIs and relative confidence score (right).

computed for each R-CNN architecture. The plots for R-CNN-1, R-CNN-2 and R-CNN-3 (Table 3.32) are reported in Fig. 3.28a, Fig. 3.28b and Fig. 3.28c, respectively; an example of object detection application is reported in Fig. 3.27.

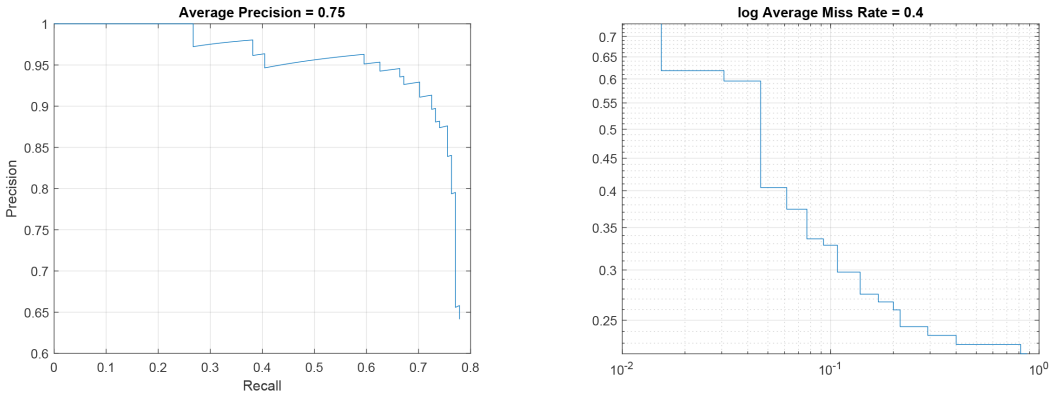
Analysing the plots, it is possible to affirm that R-CNN-1 and R-CNN-3 had an average precision higher than 0.75, with a low log-average miss rate. In particular, R-CNN-1 reached a recall value of about 0.8 with precision higher than 0.65, meaning that the classifier was able to detect 80% of the positive ROIs, accepting a high number of false positives; since the object detections procedure has been designed as a pre-processing step for a following semantic segmentation, R-CNN-1 can be considered as the best candidate among the analysed architectures.



(a)



(b)



(c)

Fig. 3.28 Precision – Recall and log Average Miss rate plots for R-CNN-1 (a), R-CNN-2 (b) and R-CNN-3 (c).

3.2.3 ADPKD Study Case: Semantic Segmentation

According to Section 2.2.3, DL architecture used for semantic segmentation tasks, such as SegNet [160], are commonly based on CNNs featuring an encoder-decoder design (Fig. 2.15). This kind of architecture includes several encoders separated by pooling layers for down-sampling reducing step-by-step the input dimensionality; each encoder includes one or more sequences of convolutional, normalisation and ReLU layers. Based on the encoding part, the decoder are specularly built with up-sampling layers for reconstructing the input size. Finally, pixel-wise scores are computed and a classification criterion labels each pixel of the input image.

Several architectures based on CNNs encoder-decoder topologies were designed and tested, starting from the well-known and general topology of VGG-16 and varying the number of encoders (and decoders), the number of layers for each encoder, the number of convolutional filters for each layer and the learner used during the training (i.e. stochastic gradient descent with momentum (SGDM), or adaptive moment estimation (ADAM) [477]). Several hyper-parameters were fixed allowing to keep unchanged the dimensions of the input across each encoder (kernels size $[3 \times 3]$, stride $[1 \times 1]$ and padding $[1,1,1,1]$); to reduce the features dimensionality between two consecutive encoders, down-sampling was performed with the max-pooling layers with kernel dimension $[2 \times 2]$ and stride $[2 \times 2]$. To reconstruct the original image size, up-sampling was performed between two consecutive decoders. The finale stage allowed the classification of each pixels of the input image into *Kidney* or *Background*. Data augmentation was performed, and the following image transformations were randomly applied:

- horizontal shift in the range $[-200;200]$ pixels;
- horizontal flip;
- scaling with a scale factor ranging in $[0.5;4]$.

The designed and tested configurations, the information about the number of layers per encoder, the number of convolutional filters per layer and the learner employed are reported in Table 3.33. The table reports only the three configurations that reached the highest performance among all the investigated architectures.

Regarding the semantic segmentation performances, the training and testing phases of each CNN architecture were performed using the full slice resolution and the slice obtained as result of the object detection step.

S-CNN-1 was the model achieving better results on full image dataset reaching an accuracy value above 88%; a comparison between the three used models is summarised

Table 3.33 Configurations designed and tested for the semantic segmentation of the full image. Each layer is a sequence of a convolutional layer, a batch normalization layer and a ReLu layer.

Network ID	Number of layers per encoder	Number of convolutional filters per layer	Learner
VGG-16	*[2 2 3 3 3]*	*[64 128 256 512 512]*	*ADAM*
S-CNN-1	*[3 2 3 3 3]*	*[64 128 256 512 512]*	*ADAM*
S-CNN-2	*[3 2 3 3 3]*	*[96 128 256 512 512]*	*ADAM*

Table 3.34 Semantic segmentation results with full image dataset.

Network ID	Mean Accuracy	Weighted IoU	Mean BF Score
VGG-16	*0.88076*	*0.75288*	*0.41117*
S-CNN-1	*0.88359*	*0.76294*	*0.38205*
S-CNN-2	*0.79824*	*0.52781*	*0.38643*

in Table 3.34. The use of an additional layer into the first encoder (S-CNN-1) allowed the creation of a more meaningful features, thus leading to a more accurate classification of the pixels. Conversely, increasing the number of convolutional filters in the first layer of the first encoder (S-CNN-2) did not improve the overall performance. For completeness, the normalised confusion matrices and an example of the semantic segmentation output are reported in Table 3.35 and Fig. 3.29, respectively, whilst the overall work-flow is depicted in Fig. 3.30.

As for the segmentation of the full images, the same architectures were considered for performing the semantic segmentation of the ROIs generated with the R-CNN-1 model; the workflow i reported in Fig. 3.31. Due to the different ROIs size resulting from the object

Table 3.35 Normalized Confusion Matrix for VGG-16, S-CNN-1 and S-CNN-2 computed on full image dataset.

		True Condition					
		VGG-16		**S-CNN-1**		**S-CNN-2**	
		Positive	***Negative***	***Positive***	***Negative***	***Positive***	***Negative***
Predicted Condition	***Positive***	*0.96629*	*0.20477*	*0.96146*	*0.19428*	*0.96595*	*0.21611*
	Negative	*0.03371*	*0.79523*	*0.03854*	*0.80572*	*0.03405*	*0.78389*

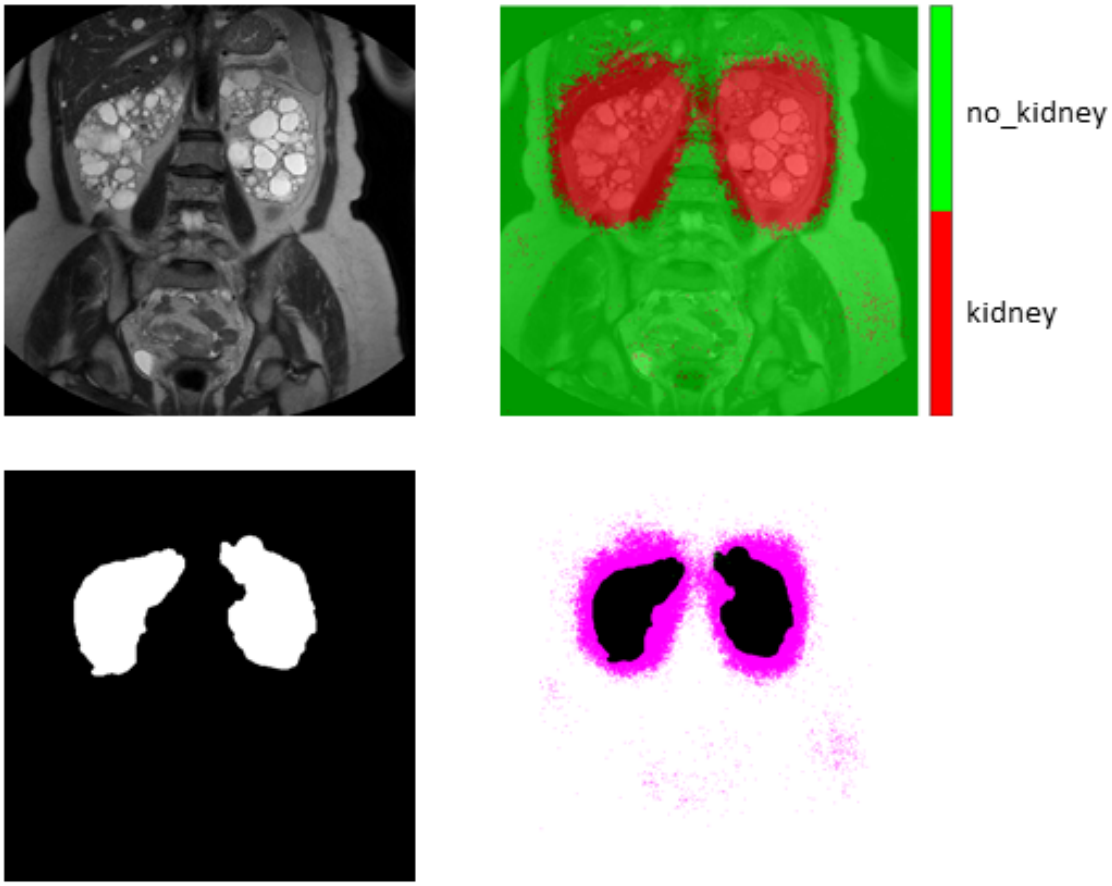


Fig. 3.29 Example of semantic segmentation on full image. Full MR slice (top left); segmentation result (top right); ground-truth mask (bottom left); superimposition of the segmentation result onto the ground-truth mask (bottom right).

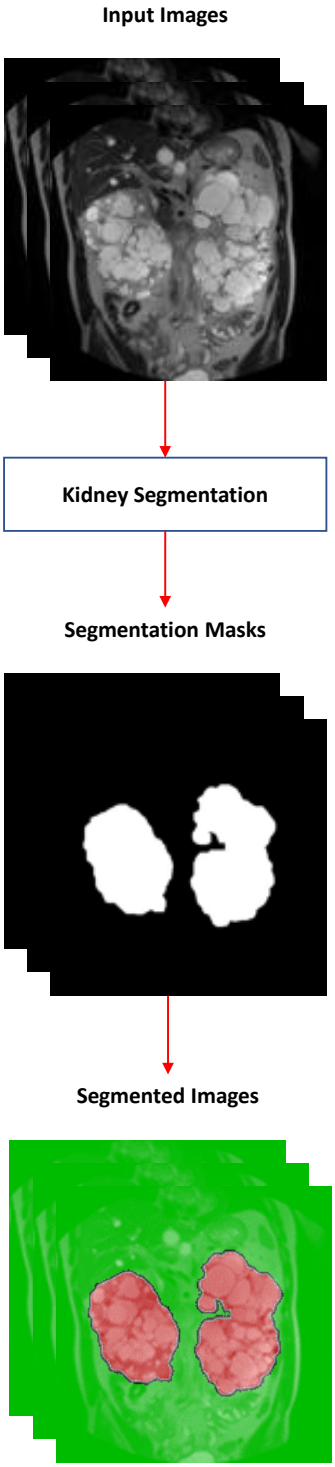


Fig. 3.30 Work-flow for the semantic segmentation starting from the full image.

Table 3.36 Semantic segmentation results with ROIs dataset.

Network ID	**Mean Accuracy**	**Weighted IoU**	**Mean BF Score**
VGG-16	*0.86016*	*0.75426*	*0.34828*
S-CNN-1	*0.8726*	*0.8540*	*0.4332*
S-CNN-2	*0.8550*	*0.82931*	*0.41515*

Table 3.37 Normalised Confusion Matrix for VGG-16, S-CNN-1 and S-CNN-2 computed on ROIs dataset.

		True Condition					
		VGG-16		**S-CNN-1**		**S-CNN-2**	
		Positive	***Negative***	***Positive***	***Negative***	***Positive***	***Negative***
Predicted Condition	***Positive***	*0.88781*	*0.16749*	*0.79742*	*0.05213*	*0.77762*	*0.06762*
	Negative	*0.11219*	*0.83251*	*0.20258*	*0.94787*	*0.22238*	*0.93238*

detection step, a rescaling procedure was performed to adapt all the ROIs to the size required by segmentation; the different magnification factor required the augmentation step to be adapted as follow:

- horizontal shift in the range $[-25;25]$ pixels;
- vertical shift in the range $[-25;25]$ pixels;
- horizontal flip;
- scaling with scale factor ranging in $[0.5;1.1]$.

Performance indices for the semantic segmentation applied on ROIs are summarised in Table 3.36. The final best performances were obtained from S-CNN-1. As in the previous case, normalised confusion matrices and processing example are reported in Table 3.37 and Fig. 3.32, respectively.

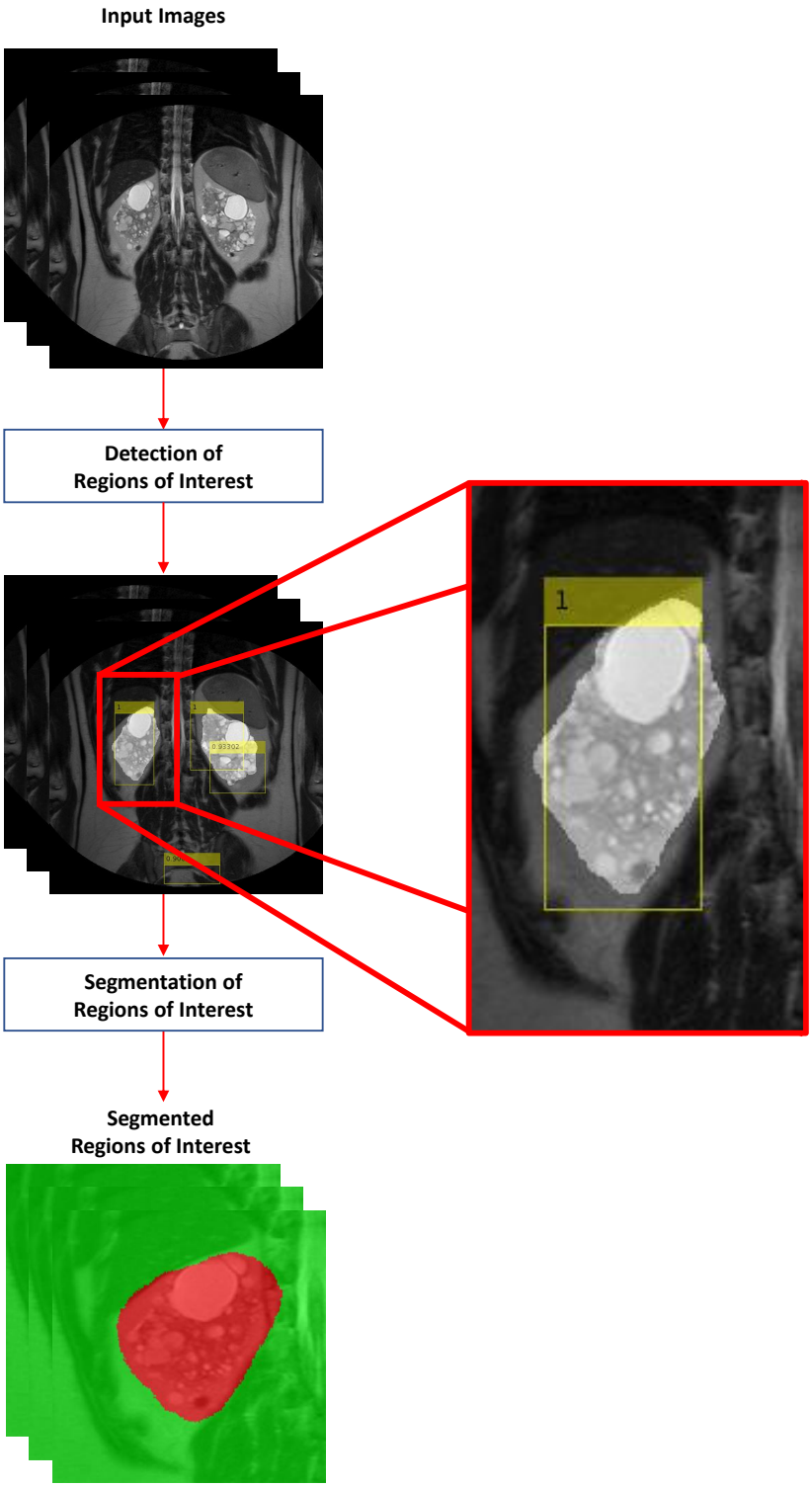


Fig. 3.31 Work-flow for the semantic segmentation starting from ROIs.

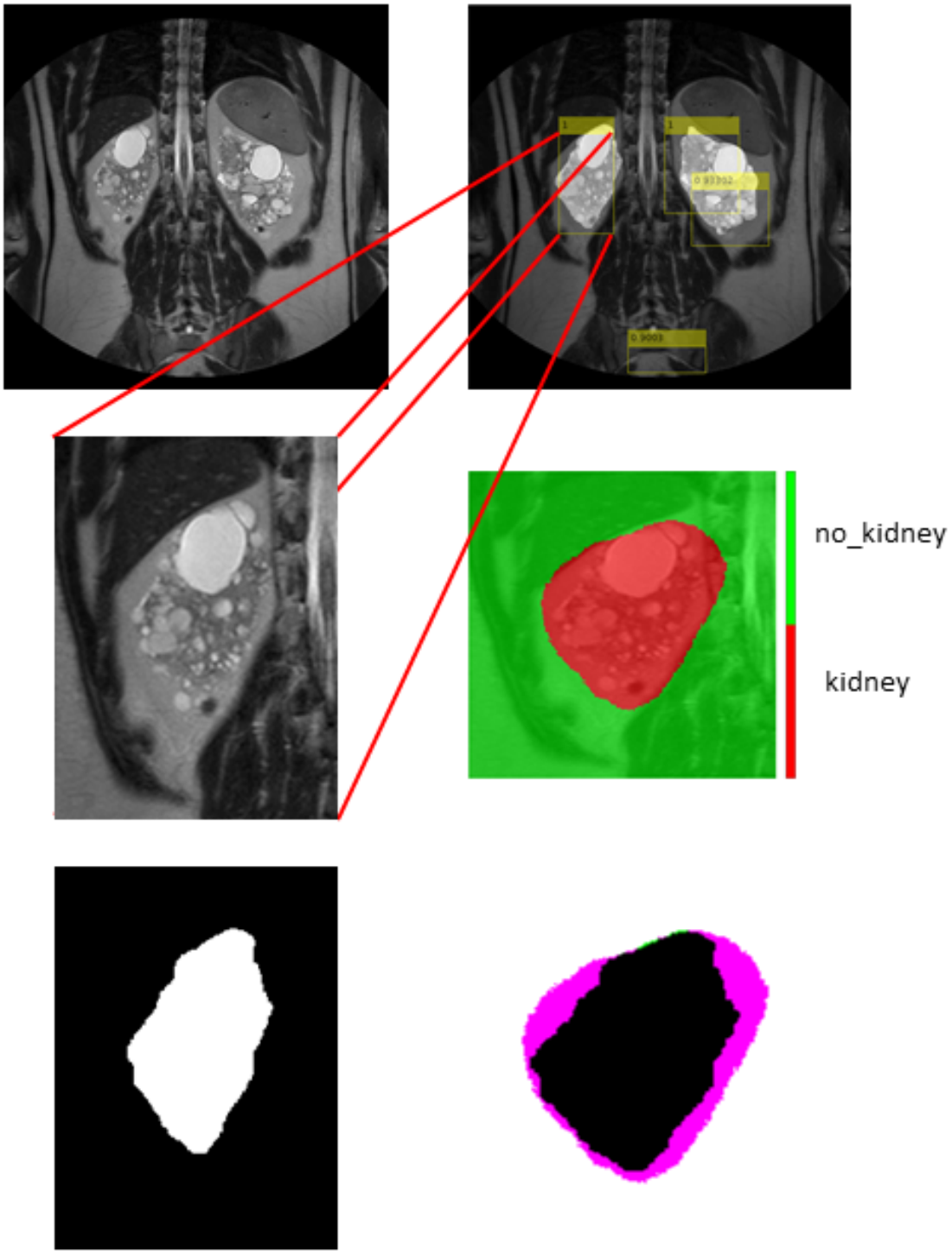


Fig. 3.32 Example result for semantic segmentation applied on ROI. Input MR slice (top left); R-CNN detection result (top right); detected ROIs (middle left); segmentation result (middle right); ground-truth mask for the considered ROI (bottom left); superimposition of the classification result onto the ground-truth mask (bottom right).

3.2.4 ADPKD Study Case: Optimal Topology for Classification and Segmentation

The Achilles' heel of the two methodologies presented before is the detection of slices containing kidneys; a right detection shown to be crucial for improving the final semantic segmentation performance. For this reason in the following paragraph a Mono-Objective Genetic Algorithm (MOGA) has been set-up to find the optimal subset of parameters of a CNN architecture able to classify input slice as "Kidney" or "Not Kidney". The joint application of classification and semantic segmentations steps, allowed the definition of a full segmentation workflow as depicted in Fig. 3.33. In particular, the MOGA algorithm allowed the definition of an optimal CNN classification architecture and the same topology has been used in the subsequent semantic segmentation step (same topology for the encoder and dual topology for decoder).

3.2.4.1 Work-flow Design and Model Configuration

Regarding the MOGA configuration, each genotype was defined to represent a convolutional architectures composed of at least one and up to three encoders; each encoder included up to three block of the following operators: convolution, batch normalisation and ReLU layer. The chromosome also modelled the number of kernels and the filter size for each convolutional layer. Finally, up to two fully connected layers, with a maximum number of 512 neurons, were modelled, and a softmax layer was defined for the final classification between the two classes. The chosen training algorithm for all the CNNs was ADAM [477], due to its more limited memory usage with respect to other algorithms. In addition, a max pooling layer (filter size $[4 \times 4]$, stride $[4 \times 4]$) was included to perform subsampling from one block to the subsequent. The training procedure also included dataset augmentation for improving the overall classification performance and the generalisation capabilities of the classifier [247]. The evolutionary optimisation started from an initial random population of 100 individuals. The crossover probability was 0.8. A simplified schema and the chromosome optimised parameters are reported in Fig. 3.34 and Table 3.38, respectively.

Each individual within the search space was trained, validated and tested on a random permutation of the dataset divided into a train, validation and a test sets with percentages of $60-20-20$, respectively. Each input image was then classified and labelled as positive if it contained at least one pixel belonging to the kidney, negative otherwise. The fitness function for evaluating the genotypes was the classification accuracy computed considering the performance on the validation set.

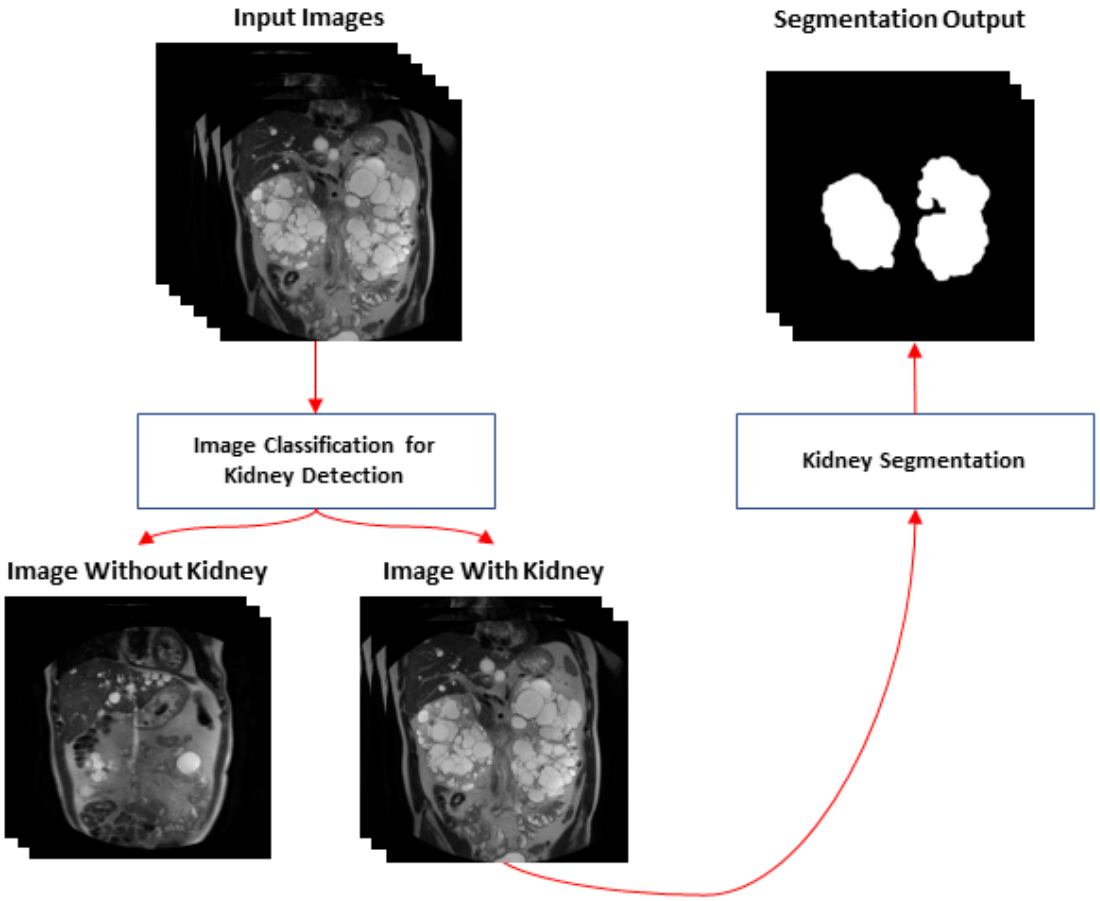


Fig. 3.33 Full workflow for kidney segmentation.

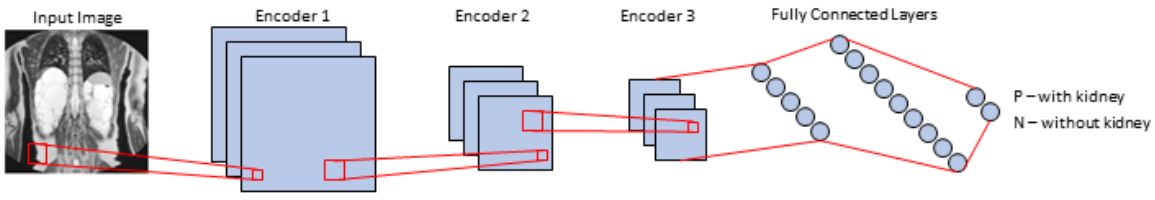


Fig. 3.34 Base schema of a CNN candidate solution by the genetic algorithm.

Table 3.38 Parameters optimised by the GA.

Network Configuration			
Layers	***GA Parameters***		
	Number of Layers	*Number of Filters*	*Filter Dimension*
Encoder 1	[1 – 3]	[8 – 256]	[1 – 7]
Encoder 2	[0 – 3]	[8 – 256]	[1 – 7]
Encoder 3	[0 – 3]	[8 – 256]	[1 – 7]
		Number of Neurons	
Fully-Connected 1		[0 – 512]	
Fully-Connected 2		[0 – 512]	

Table 3.39 Confusion matrix computed on the test set for the best topology found by the genetic algorithm.

		True Condition	
		Positive	***Negative***
Predicted Condition	***Positive***	*69*	*2*
	Negative	*3*	*29*

For designing the CNN for segmentation purposes, the optimised CNN topology obtained in the previous step was replicated in the encoding section of the semantic classifier; then the decoding part was also generated from it. All images containing at least one pixel of the kidney class constituted the dataset used for the semantic segmentation. Specifically, each image was split into a left and a right part, increasing both the number and variety of the samples size. As for image classification, the input dataset was randomly divided into train, validation and a test sets with percentages of $60-20-20$, respectively.

3.2.4.2 Results

The optimised topology found by the application of MOGA algorithm is the following: one block with two convolutional layers with 11 kernels of size $[2 \times 2]$; one block with two convolutional layers with 182 kernels of size $[5 \times 5]$; one block with one convolutional layer with 32 kernels of size $[5 \times 5]$; 56 and 232 neurons for the two fully-connected layers.

The confusion matrix of the classification results on the test set for the best topology is reported in Table 3.39. As shown, it reached an accuracy of 95.15%, a sensitivity of 95.83% and a specificity of 93.55%.

Table 3.40 Normalised confusion matrix of the final segmentation.

		True Condition	
		Positive	***Negative***
Predicted Condition	***Positive***	*0.9441*	*0.1229*
	Negative	*0.0559*	*0.8771*

Regarding the segmentation step, it achieved an accuracy of 91.06%, an IoU of 0.8296 and a BF Score of 0.5234; the normalised confusion matrix is reported in Table 3.40. An example of the application of the full workflow is depicted in Fig. 3.35. The full pipeline allowed to improve the performance achieved with the semantic segmentation only (Section 3.2.3).

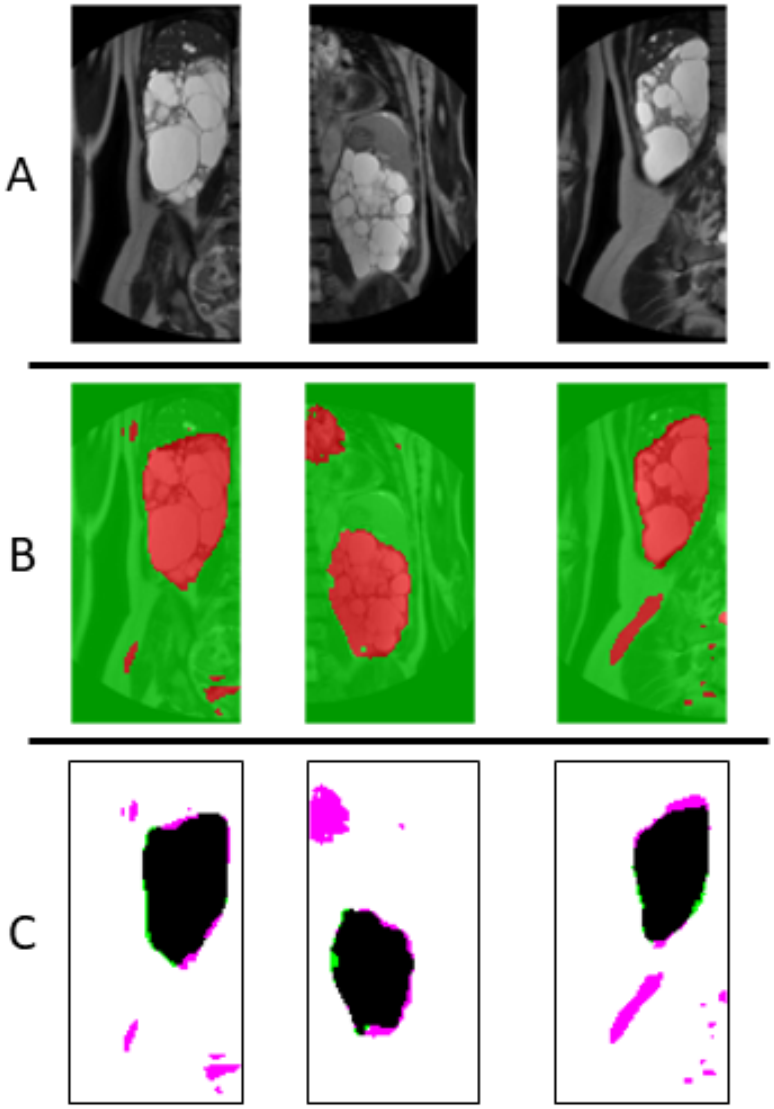


Fig. 3.35 Example of the application of the segmentation work-flow. Input images (A); superimposition of the output of the segmentation onto the input image: the red pixels are "kidney", whereas the green ones belong to the background (B); superimposition of the segmentation output onto the ground: purple pixels are the segmentation output, green pixels are the ground truth and black ones are the true positives (C).

3.3 Deep Learning in Radiology: Liver and Spleen Segmentation

As discussed in Section 2.4.2.2, researchers made considerable efforts in developing semi-automatic or automatic segmentation methods, for the segmentation of those body areas containing organs whose morphology could physiologically vary over time, or due to pathologies, such as kidneys or liver. In this section will be presented and discussed two different approaches based on classic image processing algorithms and deep learning architectures for a semi- and full-automatic spleen, liver and liver vessel segmentation.

3.3.1 A Classic Approach for Liver and Spleen Segmentation

In this section will be discussed a classic approach for segmenting liver and the spleen parenchyma from CT scans. The workflow is based on a modified version of the Region Growing (RG) algorithm; after the selection of the starting seeds, one for each organ, all the other seeds are discovered in a fully automatic way for the remaining slices, minimizing the user interaction. The seed selection phase has been improved by exploiting two utility data structures, called Moving Average Seed Heatmap (MASH) and Area Union Map (AUM), preventing the choice of next seeds from undesired regions of the CT scan.

3.3.1.1 Materials

The liver parenchyma segmentation workflow is based on SLIVER07 dataset [192]. CT scans presents a pixel spacing range in x/y- direction ranging from 0.55 mm to 0.8 mm and a slice distance ranging from 1 mm to 3 mm, depending on the acquisition protocol adopted. The algorithm parameters have been tuned on only 2 CT scans, and the remaining 18 were used for assessing the performances.

The spleen parenchyma segmentation task, instead, is based on the Medical Segmentation Decathlon Task 09 (MSD 09) dataset [478]. In this case the x/y-direction spacing varies from 0.67 mm to 0.97 mm, and the z-direction ranges from 1.6 mm to 8 mm. Only CT scans with slice distance lesser or equal than 3 mm were considered. The algorithm parameters have been tuned on only 1 CT scans, and the remaining 9 were used for assessing the performances.

3.3.1.2 Segmentation workflow

The developed RG algorithm requires the user to chose an initial slice and a seed belonging to the organ region. The initial slice must be chosen among those showing the largest liver or spleen area. Then, the centroid of the segmented area is used as seed for the segmentation of both the previous and the following slices (as suggested by Chen *et al.* [283]). Pre- and post-processing procedures were performed to filter the input data and to improve the segmentation algorithm, respectively. The algorithm work-flow is following reported.

Pre-processing This phase consists of two operations: first, the Hounsfield Unit (HU) values of the images are clipped in the range $[-150, 350]$ to exclude irrelevant organs and tissues. Then, the image is is filtered to reduce noise via a median filter with a 3×3 kernel.

Segmentation algorithm Before launching the segmentation algorithm, the slice is masked with an Area Union Map (AUM) structure. The AUM is initialized with the segmented area of the first segmented slice; then, it is computed as the union of the last ten segmented slices. If there are less than ten slices, AUM is calculated with the available ones only. The AUM allowed to exclude the components having the same intensities range of the considered organ (liver or spleen) but that are not spatially near. For the region growing step, Edman *et al.* ' approach has been investigated; it is based on the creation of a grey-scale map where the pixels with higher values are whose closer to the seed from the intensity and spatial distance point of view [479]. The algorithm searches for structures containing the seed with grey-level in the range described by Equation 3.4, where I_{seed} is the intensity of the seed and D is the maximum interval amplitude.

$$[I_{seed} - i, I_{seed} + i], \forall i = 1, 2, \ldots, D \tag{3.4}$$

The pseudo-code for the grey scale map generation algorithm is reported in Algorithm 4, whereas examples of the ongoing process are shown in Fig. 3.36. Finally, the gray scale map image obtained from the previous step is binarised by using the Otsu thresholding. An example is shown in Fig. 3.37.

Post-processing The obtained mask is refined by applying morphological operators, and in particular: morphological closing with a sphere structuring element with a radius of 4 voxels, hole filling and morphological opening with a sphere structuring element with a radius of 5 voxels. Fig. 3.38 shows an example of post-processing results.

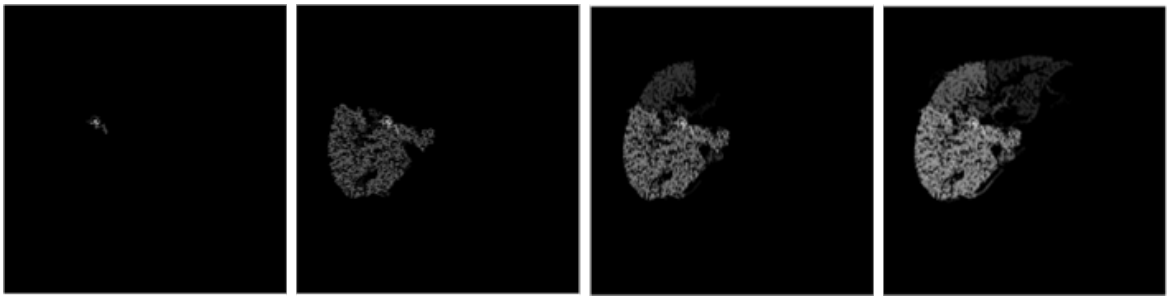


Fig. 3.36 Graphical representation of the grey scale map algorithm generation.

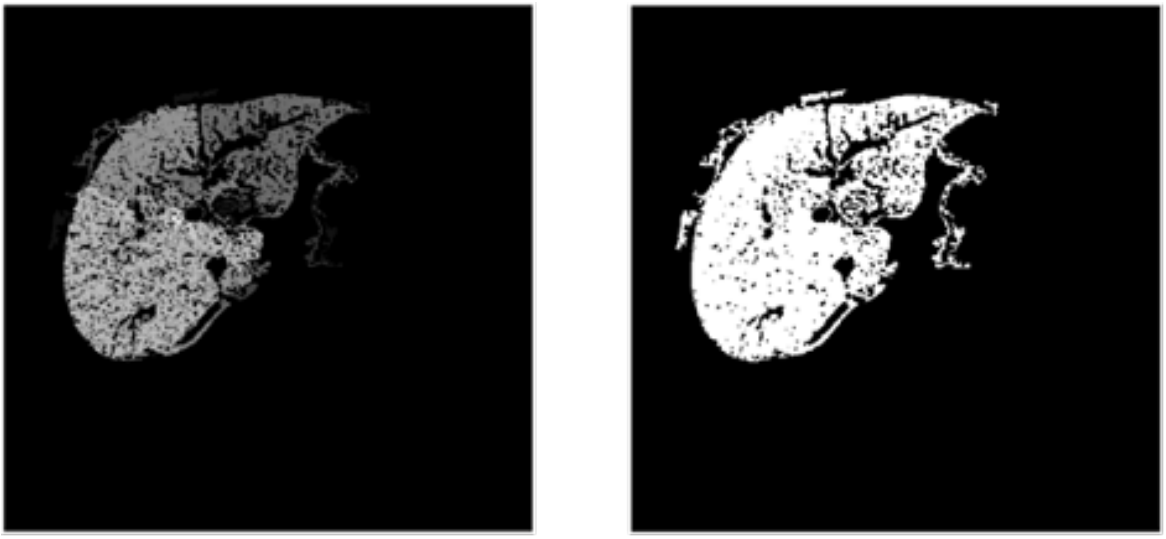


Fig. 3.37 Example of the application of the Otsu thresholding algorithm on the grey-scale map (left) to obtain a binary mask (right).

Algorithm 4: Grey-scale map generation pseudo-code

input : *image*, the input image
seed, the (x,y) coordinate of the seed
D, the maximum interval amplitude
output : *mask*, the final grey-scale map

```
mask = zeros(size(image));
SeedGrayLevel = image[seed];
for i from 0 to D do
    MaskI = (image > SeedGrayLevel − i) ∧ (image ≤ SeedGrayLevel + i);
    MaskILabels = labelsConnectedComponents(MaskI);
    for j from 1 to max(MaskILabel) do
        ConnectedComponents = extractConnectedComponents(MaskILabel, j);
        for k from 0 to size(ConnectedComponents) do
            if ConnectedComponent[k][seed] == 1 then
                mask = mask + ConnectedComponent[k];
            end
        end
    end
end
```

Due to the liver and spleen shape variability among the slices, the generated centroids could fall outside the organ region or, rather, in other undesirable parts (e.g. vessels). The problem is avoidable by computing the Intersection over Union (IoU), or Jaccard Index, defined as in Equation 2.22, between the current and the previous slice. A new seed is generated if the IoU is lesser than a threshold and accepted otherwise. Tests suggested 0.3 as good threshold value.

To choose an appropriate seed, is useful to introduce a data structure named Moving Average Seed Heatmap (MASH), which allows the determination of the most probable areas where to place the new seed, as shown in Fig. 3.39. The procedure is initialised with the area of the first segmented slice and proceed according to Equetion 3.5; γ has been set to 0.6.

$$MASH_i = \begin{cases} Mask_i & i = 1 \\ \gamma MASH_{i-1} + (1+\gamma)Mask_i & i > 1 \end{cases} \tag{3.5}$$

The MASH is updated for all the correctly segmented slices in the volume, that is if the IoU between two consecutive slices is greater than the aforementioned threshold. Otherwise, the MASH is exploited for generating the new seed. In detail, the new seed is chosen among the pixels corresponding to the maximum value in the MASH and whose grey-level deviates at

Fig. 3.38 Example of three consecutive slices before (a) and after (b) the post-processing step. This operation helps to recover from errors during the segmentation phase.

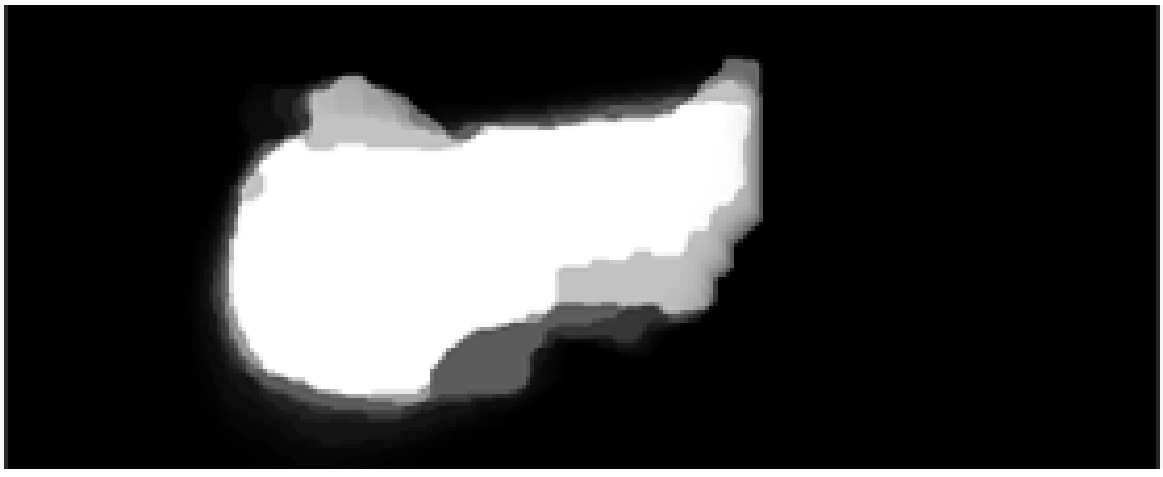


Fig. 3.39 Example of a MASH: pixel intensities are correlated with the probability of the corresponding pixel to be selected as seed point for the corresponding slice.

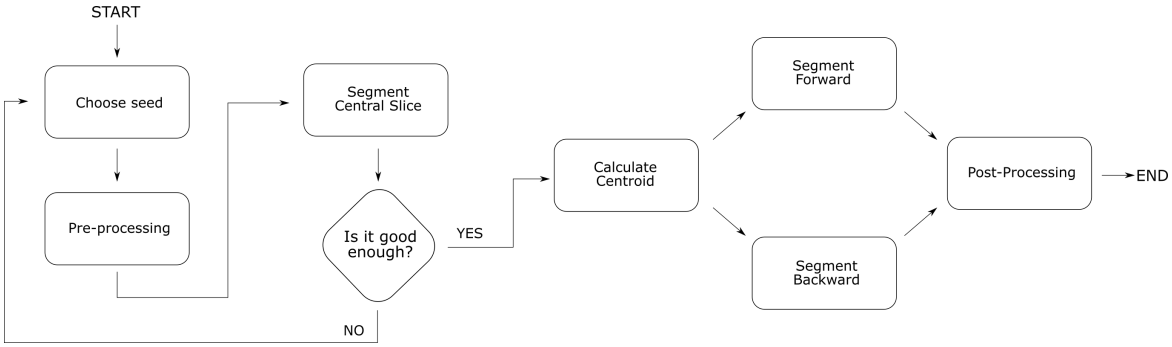


Fig. 3.40 Algorithm workflow.

Table 3.41 Proposed method for liver parenchyma segmentation. Results are expressed as "mean ± standard deviation".

Model	VOE [%]	DSC [%]	RVD [%]	MSSD [mm]	ASSD [mm]
RG (D=20)	*16.16 ± 9.20*	*90.91 ± 5.91*	*-12.18 ± 11.16*	*48.94 ± 20.70*	*2.97 ± 1.83*
RG (D=25)	*11.32 ± 4.10*	*93.95 ± 2.37*	*-3.97 ± 6.86*	*40.13 ± 17.83*	*1.95 ± 0.94*
RG (D=30)	*12.56 ± 7.30*	*93.13 ± 4.48*	*-1.12 ± 10.94*	*39.26 ± 17.05*	*2.17 ± 1.46*

most 0.01% from the mean value of the area segmented in the previous slice. This process is repeated up to 200 times and stops when the IoU between the considered slices reaches the threshold. If this condition is not met, the seed is chosen among those pixels which allowed to obtain the highest value of IoU.

The region growing stops when the IoU between the area of the slice and the mean of the 5 previous segmented ones is below the threshold or the current segmented area is lesser than a certain threshold. In the liver case it has been set to 2500 pixels, whereas for the spleen it has been set to 1000 pixels. These two values have been found by performing empirical tests. The difference in the threshold areas is due to the different sizes of the two organs. The steps of the previous algorithm for selecting the seed points are represented in the flow-charts reported in Fig. 3.40 and Fig. 3.41.

3.3.1.3 Results

The same metrics defined in the SLIVER07 challenge (Equations 2.33, 2.34, 2.36, 2.39 and 2.38) and discussed in Section 2.3.3 are used. In Table 3.41 and Table 3.43 are reported the results for liver and spleen parenchyma segmentation, respectively. Table 3.42 and Table 3.44, instead, report the results from existing literature.

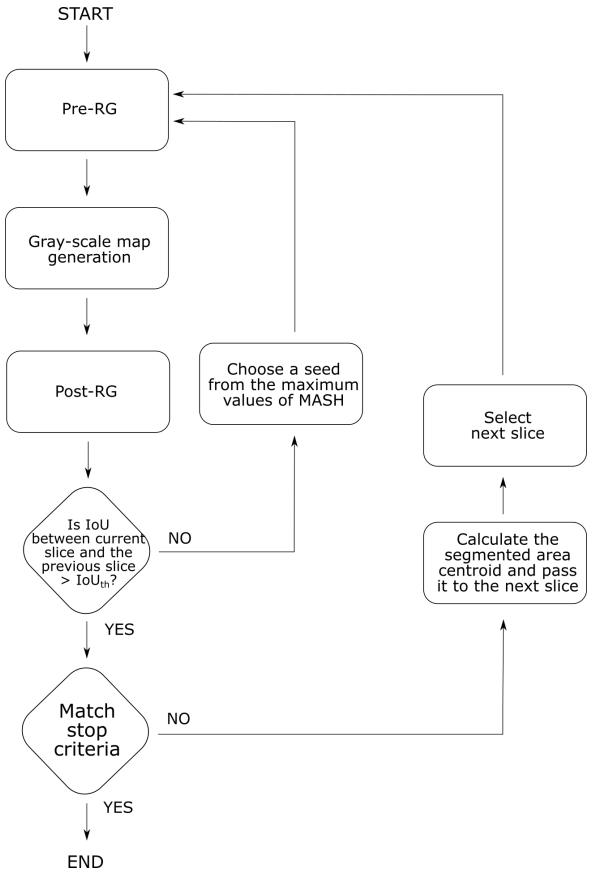


Fig. 3.41 Slice-wise segmentation flow chart.

Table 3.42 Literature overview for liver parenchyma segmentation. Results are expressed as "mean ± standard deviation" (if available).

Model	VOE[%]	DSC[%]	RVD[%]	MSSD[mm]	ASSD[mm]
Arjun *et. al.* [283]	*N/A*	*86.5*	*N/A*	*N/A*	*N/A*
Bevilacqua *et. al.* [288]	*N/A*	*90.6 ± 2.6*	*N/A*	*N/A*	*N/A*
Rafiei *et. al.* [281]	*N/A*	*92.56*	*N/A*	*N/A*	*N/A*
Czipczer *et. al.* [287]	*9.51*	*95*	*0.50*	*23.94*	*1.85*
Mostafa *et. al.* [275]	*N/A*	*96.04*	*N/A*	*N/A*	*N/A*
Lakshmipriya *et. al.* [280]	*6.11*	*97.04*	*N/A*	*N/A*	*N/A*
Kumar *et. al.* [277]	*N/A*	*97.58 ± 0.50*	*N/A*	*N/A*	*N/A*
Xu *et al.* [290]	*2.05 ± 1.30*	*N/A*	*0.66 ± 2.04*	*6.88 ± 6.13*	*0.94 ± 0.48*

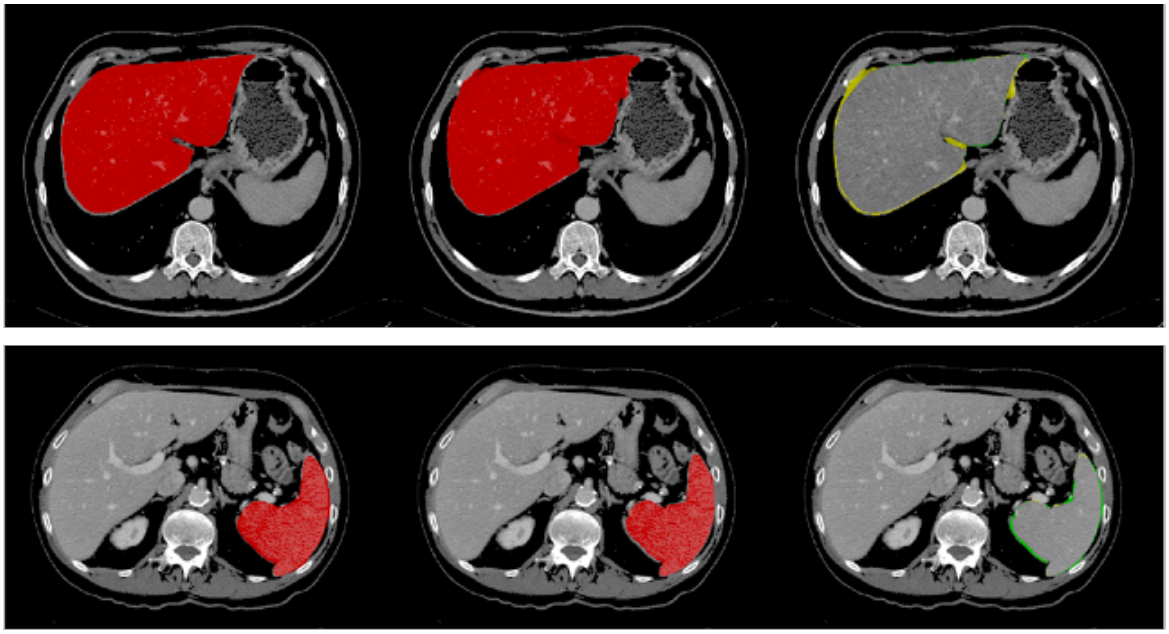


Fig. 3.42 Example of liver (top) and spleen (bottom) parenchyma segmentation: (left) ground truth; (center) prediction; (right) difference between prediction and ground truth, where the green label indicates the false negatives and the yellow label indicates the false positives.

Table 3.43 Proposed method for spleen parenchyma segmentation. Results are expressed as "mean ± standard deviation".

Model	**VOE [%]**	**DSC [%]**	**RVD [%]**	**MSSD [mm]**	**ASSD [mm]**
RG (D=20)	*16.94 ± 3.98*	*90.69 ± 2.30*	*-14.54 ± 4.40*	*11.93 ± 4.76*	*1.27 ± 0.49*
RG (D=25)	*14.95 ± 3.60*	*91.88 ± 2.10*	*-11.33 ± 3.89*	*9.79 ± 2.99*	*1.02 ± 0.37*
RG (D=30)	*14.50 ± 3.60*	*92.14 ± 2.09*	*-8.70 ± 3.97*	*11.75 ± 5.54*	*1.05 ± 0.43*

Table 3.44 Literature overview for spleen parenchyma segmentation. Results are expressed as "mean ± standard deviation" (if available).

Model	**DSC [%]**	**ASSD [mm]**
Mihaylova *et. al.* [291]	*79.83*	*N/A*
Behrad *et. al.* [293]	*93.97 ± 2.27*	*N/A*
Gauriau *et. al.* [295]	*87 ± 15*	*2.6 ± 3*
Reza Soroushmehr *et. al.* [296]	*97.3*	*N/A*

For the liver parenchyma segmentation, the proposed algorithm reached the best performance with $D = 25$ obtaining the best DSC, VOE, MSSD and ASSD results. For the spleen parenchyma segmentation, the best configuration among those proposed is that with $D = 30$, which has a mean DSC value greater than 92%. In both cases, the results are comparable with the literature, some of which present a more complex segmentation process. For the spleen parenchyma segmentation, the proposed approach consents to improve the existing results for ASSD, by obtaining a mean ASSD of $1.02mm$, with a standard deviation of $0.37mm$. Fig. 3.42 shows some examples for both liver parenchyma and spleen parenchyma segmentation task.

3.3.2 A Deep Learning Approach for Liver and Liver Vessels Segmentation

In this section the liver segmentation problem will be faced from the deep learning perspective, allowing the creation of a full automatic pipeline and extending the segmentation granularity to liver vessels.

3.3.2.1 Materials

Liver Parenchyma Segmentation The proposed automatic liver parenchyma segmentation approach, is based on two different dataset; the training dataset is the training set of the Liver Tumor Segmentation (LiTS) Challenge, containing CT abdominal acquisitions of 131 subjects [191], while the test dataset is the same SLIVER07 dataset used in Section 3.3.1.1. In the CT scans of LiTS Challenge, pixel spacing ranges from $0.56mm$ to $1.0mm$ in x/y-direction, whilst slice from $0.45mm$ to $6.0mm$, with the number of slices varying from 42 to 1026. All the images were pre-processed by windowing the HU values in the range $[-150, 350]$. For the CNN model, the values were then scaled in the range $[0, 1]$.

Liver Vessels Segmentation 20 CT scans of 3D-IRCADb dataset[2] were used for training and internally cross-validating the CNN model, and a private dataset provided by the Polyclinic of Bari composed of 4 CT scans was used as an independent test set for external validation. The 3D-IRCADb dataset contains CT scans whose axial-plane resolution ranges from $0.56mm$ to $0.81mm$, whilst resolution along the z-axis spans from $1mm$ to $4mm$. The number of slices ranges from 74 to 260. The 20 CT scans of 3D-IRCADb come from 10 women and 10 men, and the number of patients with hepatic tumours is 75% of the overall

[2]https://www.ircad.fr/research/3d-ircadb-01/

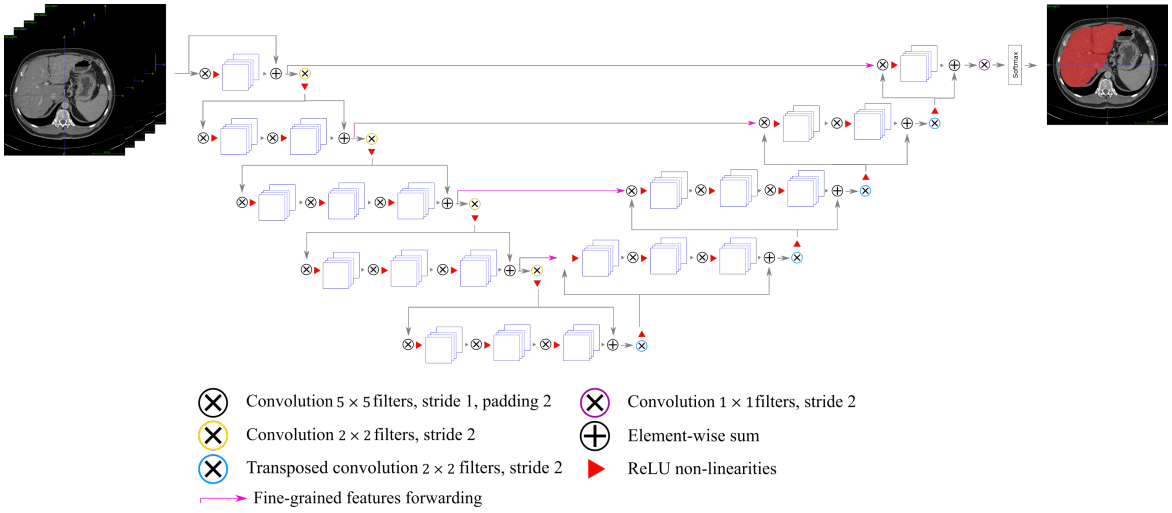


Fig. 3.43 Proposed 2.5D V-Net architecture.

dataset. The dataset from the Polyclinic of Bari contains 4 CT scans with an axial plane resolution varying from 0.76*mm* to 0.86*mm*, and a z-axis resolution spanning from 0.7*mm* to 0.8*mm*. The number of slices is between 563 and 694. Pre-processing adopted for liver vessels segmentation was the same used for liver parenchyma segmentation.

3.3.2.2 Segmentation workflow

In the biomedical image segmentation field, a well-known and widely used semantic segmentation CNN architecture are U-Net [171] and its extensions [172, 173]

Starting from the V-Net architecture proposed by Milletari *et al.* [173], a 2.5D variant is proposed. In detail, all the 3D layers are replaced by the corresponding 2D ones, with a first layer which processes 5 slices as 5 channels. In the adopted architecture, *down* convolutional layers have kernel size 2×2 and stride 2×2; normal convolutional layers have kernel size 5×5, and transposed convolutional layers used as *up* convolutions have 2×2 kernels. Moreover, a batch-normalization layer is added after each convolutional layer. The use of batch-normalization has been taken into consideration by the same authors of the original V-Net [480]. Standard ReLu is used instead of PReLu non-linearity. The 2.5D variant of the V-Net architecture is depicted in Fig. 3.43.

For both the considered tasks, it was trained a 2.5D V-Net by taking random patches of 5 slices from the training set, assigning a higher probability to take a patch containing at least one voxel belonging to the liver or to the vessels. The optimiser for the training process was Adam [477], with a starting learning rate of 0.01. The network is trained for 2001 epochs,

reducing the learning rate by 10 every 333 epochs. 200 samples were processed per each epoch. The batch size has been set to 4.

In order to ensure lower convergence time, it is crucial to select a proper loss function, such as Binary Cross-Entropy (BCE) loss function, reported in Equation (3.6), or the Weighted BCE (WBCE), reported in Equation (3.7).

For the following definitions, let $p_i \in P$ be the probability of the ith voxel to belong to the liver and $g_i \in G$ its binary label with $i = 1,\ldots,N$; P and G respectively being the predicted segmented volume and the ground truth volume.

$$BCE = -\frac{1}{N}\sum_{i=1}^{N}(g_i \cdot log(p_i) + (1-g_i)\cdot log(1-p_i)), \tag{3.6}$$

$$WBCE = -\frac{1}{N}\sum_{i=1}^{N}(\omega_1 \cdot g_i \cdot log(p_i) + \omega_0 \cdot (1-g_i)\cdot log(1-p_i)). \tag{3.7}$$

In Equation 3.7, ω_1 and ω_0 are introduced to give a different weight for positives and negatives. These functions act as a proxy for the optimization of the true measures used later for the evaluation, which usually include the Dice Coefficient (Eq. 3.8). Thus, another plausible choice for the optimization function consisted of directly adopting an objective function based on the Dice Coefficient [173], or, more generally, the Tversky index. Salehi *et al.* exploited the Tversky Loss function for lesion segmentation by means of 3D CNNs [481]. The used implementation is reported in Equation 3.9, where p_{0i} is the probability of the ith voxel to be positive, g_{0i} its binary label (i.e., 1 for positives and 0 for negatives), p_{1i} its probability of being a negative, g_{1i} its negated binary label, directly obtained applying logical NOT to g_{0i} (i.e., 0 for positives and 1 for negatives).

$$D = \frac{2\cdot\sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} g_i} \tag{3.8}$$

$$T_{\alpha,\beta} = \frac{2\cdot\sum_{i=1}^{N} p_{0i} g_{0i}}{\sum_{i=1}^{N} p_{0i} g_{0i} + \alpha\sum_{i=1}^{N} p_{0i} g_{1i} + \beta\sum_{i=1}^{N} p_{1i} g_{0i}} \tag{3.9}$$

Tversky index-based loss function is used due to the unbalanced voxels problem. In fact, the voxels belonging to the liver region are only a fraction of the whole CT scan. In the LiTS dataset, the unbalancing ratio is approximately 40 : 1 in favour of negative voxels, becoming more relevant for the vessels segmentation (about 200 : 1 in favour of negative voxels). Dice Loss does not give different weights to false positives and false negatives, thus it does not focus the learning on the maximization of the recall of the voxels of interest. With

a Tversky Loss, thanks to the α and β coefficients, it is possible to give a major weight on false negatives.

Different data augmentation techniques are considered, as slice-wise right-left flipping of volume patches, gaussian blur, elastic transform with $\alpha = 2$ and $\sigma = 3$, multiplicative noise, random rotations in the range $[-10°, 10°]$, random brightness and contrast perturbations.

In the inference phase, instead, the volumetric images is processed in a 3D sliding window fashion, processing sub-volumes of $512 \times 512 \times 5$ voxels. Due to the adoption of a 2.5D approach, the five processed slices were used for predicting the central one only. In order to create patches of five slices also at the begin and at the end of the CT scans, the first and last slices were replicated.

Morphological opening operator is used as post-processing in order to separate, if needed, the liver and the spleen. In fact, due to the similarity between spleen and liver intensity values and texture, the two organs could be misclassified. Then, connected components labelling is applied retaining only the largest one, since the liver is the largest organ in the abdomen. Finally, morphological closing and morphological hole-filling are applied to the segmented masks. A similar procedure has been carried out for the vessels segmentation, without the morphological and connected components labelling post-processing.

3.3.2.3 Results

To evaluate the performance of the implemented segmentation algorithms, the same indexes (Equations 2.33, 2.34, 2.36, 2.39 and 2.38) adopted in the SLIVER07 and LiTS challenges [191, 192] and discussed in Section 2.3.3 are used. Performances indexes, such as accuracy (Eq. 2.13), recall (Eq. 2.15) and specificity (Eq. 2.16) are also computed, considering as positives the voxels belonging to the liver or vessels (depending on the segmentation task) and negatives the others.

The results obtained for the liver parenchyma and liver vessels segmentation are reported in Table 3.45 and Table 3.46, respectively. The 2.5D V-Net trained with the Dice loss allowed to obtain a mean Dice Coefficient of 96.13% and a mean MSSD of 140.42*mm* for the liver parenchyma segmentation task. Since in a surgical planning setup is crucial to reduce the Hausdorff distance, has been observed that the adoption of the post-processing is very beneficial. It permitted to reduce the mean MSSD to 31.50*mm*, and its standard deviation from 86.96*mm* to 12.05*mm*. A so large reduction of the MSSD can be explained by the fact that sometimes the liver CNN model also segments the spleen region, as previously stated. The proposed approach outperforms other methods proposed in the literature, as reported in Table 3.45. Note that, since the liver segmentation task is not too unbalanced, the

Table 3.45 Liver parenchyma segmentation results expressed as "mean ± standard deviation".

	Model	DSC [%]	VOE [%]	RVD [%]	ASSD [mm]	MSSD [mm]
Proposed 2.5D V-Net	DL	*96.13 ± 2.45*	*7.35 ± 4.31*	*1.13 ± 6.24*	*4.18 ± 7.95*	*140.42 ± 86.96*
	DL, PP	*96.52 ± 1.84*	*6.66 ± 3.32*	*4.26 ± 3.95*	*1.44 ± 0.89*	*31.50 ± 12.05*
	TL ($\alpha = 0.3$,$\beta = 0.7$)	*96.27 ± 1.57*	*7.16 ± 2.86*	*1.24 ± 4.03*	*3.56 ± 7.48*	*141.04 ± 84.54*
	TL, PP ($\alpha = 0.3$,$\beta = 0.7$)	*96.26 ± 1.18*	*7.19 ± 2.18*	*5.11 ± 3.24*	*1.50 ± 0.61*	*31.26 ± 12.18*
	TL ($\alpha = 0.1$,$\beta = 0.9$)	*94.21 ± 3.01*	*10.81 ± 5.16*	*7.05 ± 5.90*	*6.03 ± 8.56*	*158.37 ± 65.09*
	TL, PP ($\alpha = 0.1$,$\beta = 0.9$)	*93.88 ± 2.40*	*11.45 ± 4.12*	*10.99 ± 6.15*	*2.43 ± 1.11*	*33.95 ± 13.01*
Lu *et al.* [21]		*N/A*	*9.21 ± 2.64*	*1.27 ± 3.85*	*1.75 ± 1.41*	*36.17 ± 15.90*
Rafiei *et al.* [22] - 3D-2D-FCN + CRF		*93.52*	*N/A*	*N/A*	*N/A*	*N/A*
Kim *et al.* [23] - 3D U-Net		*95.9 ± 1.8*	*N/A*	*N/A*	*0.71 ± 0.30*	*8.93 ± 6.30*

Acronyms: DL - Dice Loss, TL - Tversky Loss, PP - post-processing.

Table 3.46 Liver vessels segmentation results, expressed as “mean ± standard deviation”.

	Model	Accuracy [%]	Recall [%]	Specificity [%]	ASSD [mm]
Proposed 2.5D V-Net	Dice Loss	*99.94 ± 0.02*	*56.67 ± 20.13*	*99.96 ± 0.03*	*9.51 ± 0.52*
	Tversky Loss	*99.93 ± 0.03*	*62.62 ± 17.01*	*99.94 ± 0.03*	*14.84 ± 2.93*
	Tversky Loss	*99.91 ± 0.04*	*68.92 ± 19.13*	*99.92 ± 0.05*	*15.79 ± 2.21*
Goceri *et al.* [29] - AVS		*89.57 ± 0.57*	*N/A*	*N/A*	*23.1 ± 16.4*
Chi *et al.* [30] Context-Based Voting		*98 ± 1*	*70 ± 1*	*99 ± 1*	*2.28 ± 1.38*
Zeng *et al.* [31] Oriented Flux Symmetry and Graph Cuts		*97.7*	*79.8*	*98.6*	*N/A*

V-Net trained with Tversky loss with $\alpha = 0.3$, $\beta = 0.7$ leads to results comparable with the network trained with the Dice Loss, whereas the V-Net trained with the Tversky loss with $\alpha = 0.1$, $\beta = 0.9$ leads to worse overall results, since the coefficients penalize too much the false negatives. Regarding the vessels segmentation task, the proposed method shows high accuracy (higher than 99%), also compared to other approaches proposed in the literature, as can be seen in Table 3.46. The adoption of the Tversky loss to penalize false negatives more than false positives yields to progressively higher recall values when α is decreased and β is increased, with a best recall of 68.92 in the configuration with $\alpha = 0.1$, $\beta = 0.9$. Thus, in case of extremely unbalanced dataset, the use of the Tversky loss function becomes appreciable. However, it is important to note that the used test set is very small, and the results suffer from high variability. Examples of liver segmentation results obtained with the proposed method are depicted in Fig. 3.44, whereas examples of vessels segmentation results are reported in Fig. 3.45 and Fig. 3.46.

Comparing the deep learning based liver segmentation with the classic approach of Section 3.3.1, on the common SLIVER07 dataset, the proposed CNN pipeline allowed to obtain higher voxel-level performances, with Dice Coefficient greater than 0.96 and VOE lesser than 0.07; the region growing approach, instead, allowed to get a Dice Coefficient greater than 0.93 and VOE lesser than 0.12. At surface-level, the CNN approach obtained a MSSD of 31.50*mm* and an ASSD of 1.44*mm*, whereas the region growing algorithm reached MSSD of 39.26*mm* and ASSD of 1.95*mm*. Furthermore, the CNNs deep learning pipeline reached more stable result with lower standard deviation and, really important, it is fully automatic methodology but, as a drowback, its workflow is more complex than the semi-automatic procedure implementing the region growing algorithm. The obtained results show that the 2.5D V-Net, trained with a Tversky Loss, is a very promising approach for the liver and liver vessels segmentation in CT scans, allowing to obtain accurate volumetric reconstructions of the segmented regions.

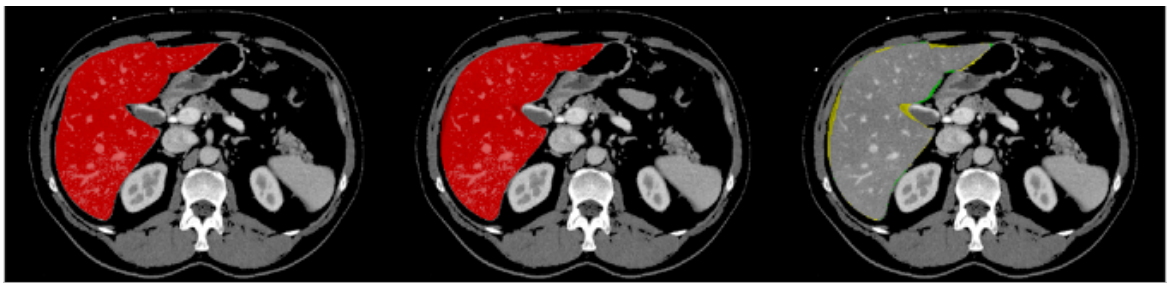


Fig. 3.44 Example of slice obtained with the liver segmentation pipeline: (left) ground truth; (center) Dice Loss-based 2.5D V-Net prediction; (right) difference between ground truth and prediction, where false negatives and false positives are respectively evidenced in green and in yellow.

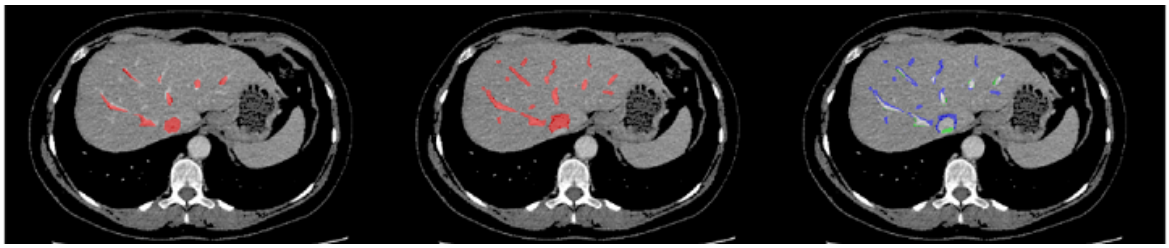


Fig. 3.45 Example of slice obtained with the liver vessels segmentation pipeline: (left) ground truth; (center) Tversky Loss-based 2.5D V-Net prediction; (right) difference between ground truth and prediction where false negatives and false positives are respectively evidenced in green and in blue.

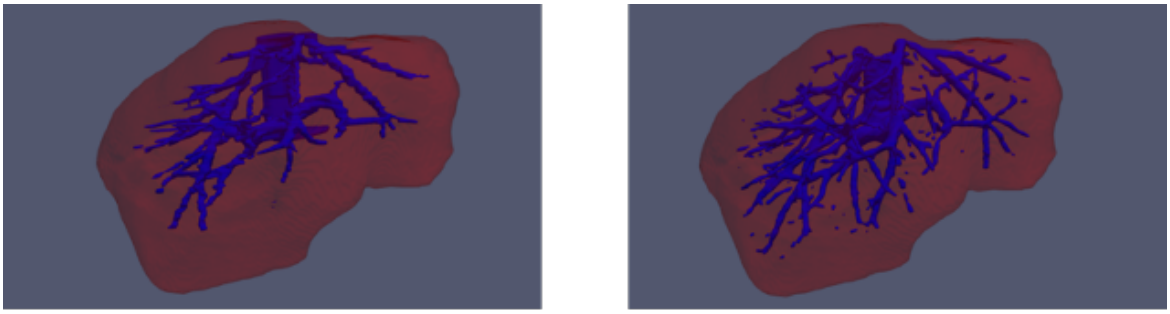


Fig. 3.46 Example of mesh obtained by the liver vessels segmentation: (left) ground truth; (right) Tversky Loss-based 2.5D V-Net prediction.

Chapter 4

Machine Learning and Deep Learning for Signals Processing

The first insight of autoencoder architecture on electromyographic signals is based on the application of an undercomplete autoencoder to extract spatial muscle synergies. The presented bio-inspired autoencoder topology (Section 4.2) has been trained to extract muscles synergies from EMG signals of the main upper-limb muscles, acquired during isometric reaching tasks. The extracted synergy activations have been also used to estimate the moments applied to the shoulder and elbow articulations. The experimental results were compared with the standard NNMF algorithm used in muscle synergy extraction. A second application (Section 4.3), starting from the results of the first one, make use of a customised autoencoder-based neural model able to extract the muscle synergy patterns simultaneously considering the performance in the task space (i.e., estimation of moments/forces exerted by the human upper limb). Specifically, the model builds its synergy code considering both the EMG signals reconstruction performance and the estimation quality of the upper limb moments computed as a linear combination of the synergy activation signals, thus allowing a task-oriented synergy extraction. The creation of a more complex model is due to the assumption that a direct integration of task-space constraints in the algorithm used to extract the synergies, could produce a better task-space variable estimation, thus leading to a new class of optimized myo-controllers and, perhaps, providing a deeper understanding of the hypothetical modularity of the central nervous system and its relationship with the motor learning.

Part of this chapter has been published in international conferences and journals [114, 115, 332, 337].

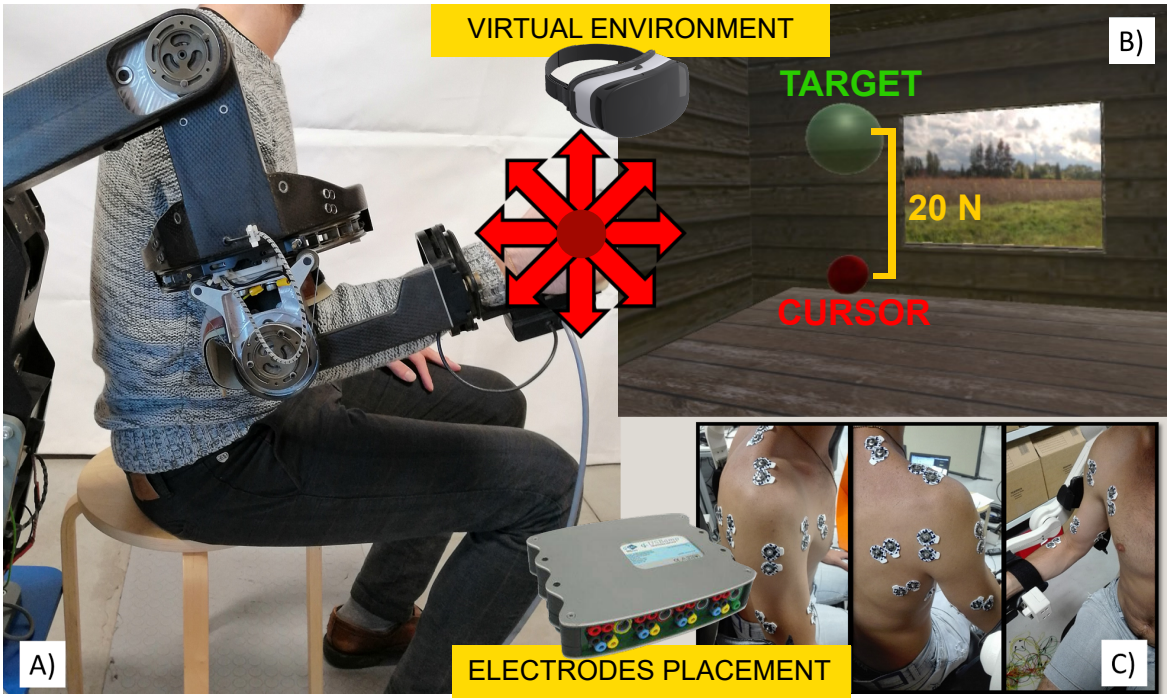


Fig. 4.1 The experimental set-up. The subject that is wearing the upper limb L-Exos exoskeleton (A). The virtual environment showing the cursor and the target sphere (B). The surface EMG electrode placement (C).

4.1 Materials

For the research purpose analysed in this section, two dataset have been considered.The first study involved six healthy subjects, while the second extends the first with a larger number of participant (nine right-handed healthy subjects, age 27.7 ± 4.9 years, weight 74.1 ± 9.1 kg). The experiments were conducted in accordance with the WMA Declaration of Helsinki and all subjects provided written consent to participate.

Experimental Setup The setup was designed for measuring the subject upper-limb muscle EMG signals and forces exerted at the hand level during a set of isometric contractions (Fig. 4.1). An electromechanical upper-limb exoskeleton, designed for upper-limb rehabilitation, namely L-Exos, was used for acquiring the interaction force between the subject's hand and the exoskeleton's cylindrical handle featuring a triaxial force sensor. The L-Exos has been designed as a wearable haptic interface, capable of providing a controllable force at the center of user's right hand palm, oriented along any direction of the space [482]. The L-Exos has four actuated DOFs for supporting elbow and shoulder movements: shoulder

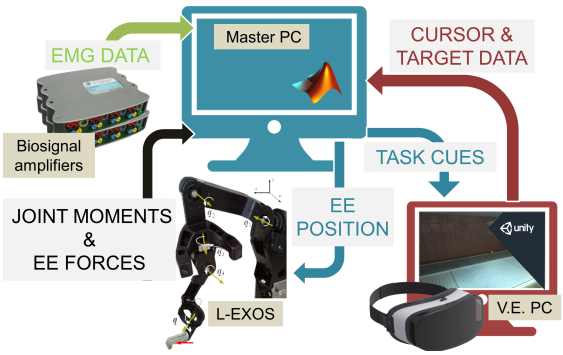


Fig. 4.2 Main architecture of the acquisition system.

adduction/abduction; shoulder flexion/extension; shoulder internal/external rotation; elbow flexion/extension, and one passive DOF used for measuring the wrist pronosupination angle. All the motors of the exoskeleton have been located on the fixed frame. For each actuated DOF, the torque is delivered from the motor to the corresponding joint by means of steel cables and a reduction gear integrated at the joint axis. All actuated joints are driven with a proportional-derivative control strategy with gravity compensation. The force sensor readings have been then used to estimate the articulation moments.

Concerning the EMG acquisition system, two bio-signals amplifiers (g.USBamp, gTec, Austria) were included in the setup to record the activity of 13 muscle heads: biceps short head, biceps long head, brachioradial, triceps long head, triceps lateral head, deltoid anterior head, deltoid posterior head, trapezius, pectoralis major, teres major, infraspinatus, latissimus dorsi and rhomboid. Disposable Ag/AgCl surface electrodes were placed by following the SENIAM[1] recommendations, after a skin cleaning process, and the ground electrode attached to the right elbow. All the surface EMG signals were acquired at $1200Hz$ sampling frequency and filtered by the amplifier with a $5-500Hz$ band-pass filter and a $50Hz$ notch filter.

In order to make a more intuitive and easy experimental session, the subject was immersed in a virtual environment (VE) by wearing a head mounted display (Oculus Rift HMD, Oculus) to receive visual feedback. The force sensor measurements, VE signals and EMG data were synchronized on a Master PC (Fig. 4.2), featuring Microsoft Windows 10 (64 bit), Intel i7 1.6 GHz, 8 Gb RAM and Matlab (Release 2018b). The Master PC has been also used to generate commands for driving the exoskeleton and the VE, according to the acquisition routine.

Data Acquisition Protocol Before starting the acquisition routine, subjects were invited to sit on a chair and wear the exoskeleton using the flip-off arm bands. By using stacked

[1] http://www.seniam.org/

hard plastic layers under the chair, the height of the seat was adjusted in order to align the centres of rotation of the subject's and exoskeleton shoulder joint. At the beginning of the experiment the exoskeleton joint angles were automatically fixed to a pre-defined angles set: shoulder abduction/abduction angle equal to 0 degrees, shoulder internal rotation angle equal to 0 degrees, shoulder elevation angle equal to 10 degrees and elbow angle equal to 90 degrees. After the surface EMG electrodes were placed on the targeted muscles, elastic bands were used to keep electrodes and wires firmly attached to the body in such a way that the exoskeleton handle was easily reachable. Then, subjects were asked to perform 16 isometric virtual reaching tasks along 8 directions (two trials per direction) on the sagittal plane, equally spaced at 45 degrees and randomly sorted. Isometric contractions were achieved through the exoskeleton end effector position control, keeping the subjects upper-limb pose fixed. In the virtual environment, the subjects hand position corresponds to a red sphere (cursor) and the task target is represented as a green sphere. The distance between the two spheres is covered applying the target force of 20 $kg \cdot m/s^2$ on the sensor and the radius difference allowed a maximum positioning error equal to 3 $kg \cdot m/s^2$ ($1\ N = 1\ kg \cdot m/s^2$). Each virtual reaching task consists of: positioning the cursor inside the target, holding it in place for 2 s and then relaxing to move the cursor back to the rest position. The cursor position is driven by a spring model $P_c = K * F_{EE}$ where P_c is the 3D cursor position, F_{EE} is the applied isometric force vector and K is the elastic constant of the virtual spring.

4.2 An Undercomplete Autoencoder for Muscle Synergies Extraction

Among the several kinds of AE families cited in Section 2.2.1.2, an undercomplete autoencoder has been chosen and firstly used to extract the spatial muscle synergies of the human upper limb while executing an isometric reaching task in a bi-dimensional space. In a second step, the extracted muscles synergy signals have been used to predict the movement intention by estimating the shoulder and elbow articulation moments with a linear combination.

Given the input vector $I(t) = [m_1(t), m_2(t), \ldots, m_N(t)]$, where $m_i(t)$ represents the pre-processed activation of the i-th muscle and N is the number of considered muscles, the autoencoder has the objective to extract the activations of the muscle synergies s_i. The pre-processing phase considers the following steps for each raw EMG signal: high-pass filtering (20 Hz second-order Butterworth); rectification and low-pass filtering (5 Hz second-

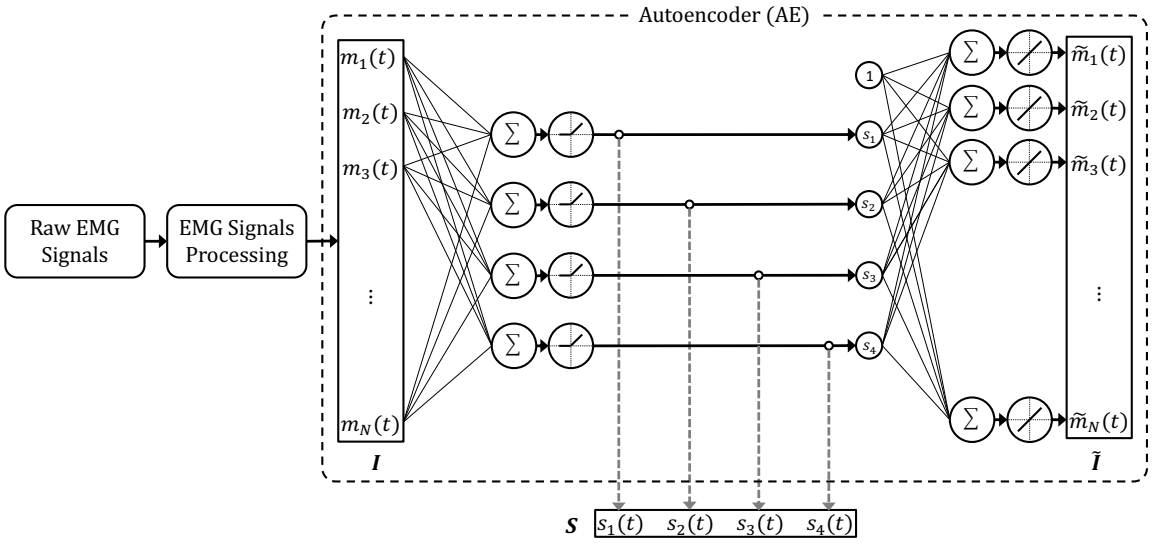


Fig. 4.3 Architecture of the undercomplete autoencoder for synergies extraction.

order Butterworth); normalization over the maximum value computed during the calibration procedure.

4.2.1 Model Design and Configuration

The structure of the designed autoencoder is shown in Fig. 4.3. The considered topology has one hidden layer with four positive linear neurons that produce the synergy activations named $s_1(t)$, $s_2(t)$, $s_3(t)$ and $s_4(t)$. Such configuration replicates the physiological model of the spatial muscles synergies reported by Berger *et al.* [402]. The problem of selecting the best number of the hidden neurons was not faced with the topology definition, i.e. the number of extracted muscles synergies, but a fixed number (4) of hidden neurons has been chosen since some studies have reported that the isometric activation of the human upper limb muscles can be accurately described by four muscle synergies [335, 402]. Considering the definition of autoencoders, the output layer has the same dimension as the input layer. As suggested by Goodfellow *et al.* , a simple linear decoder with biases is sufficient to avoid the copying task without extracting useful information, caused by too much learning capacity [116].

The model has been implemented with the Neural Network toolbox of Matlab (Release 2018b) and trained using a gradient descent with momentum and adaptive learning rate algorithm and considering 1000 training epochs. Given a single train set, the training was repeated 20 times with different initial weights. Among the 20 models, the best one has been chosen as the model featuring the minimum correlation index among the four synergy

activations. Such index has been computed as the sum of the elements of the absolute upper triangular matrix extracted by the correlation matrix of $s_1(t)$, $s_2(t)$, $s_3(t)$ and $s_4(t)$. Considering a train set composed by about 4000 time point, the training process of each autoencoder lasts about 1.5 seconds. All the trainings have been run on a PC featuring two Intel XEON E5 2630 v3 CPUs and 64 GB of RAM.

4.2.2 Joint Moment Estimation Based on Muscle Synergies: Comparison with the State-of the Art

For this study, the bi-dimensional motion intention estimator that takes the AE-extracted muscles synergies as input has been compared with other methods already proposed in literature and based on the same model described by the Equation 2.42. Each model is able to estimates the shoulder and elbow moments. In detail:

- **Model** Hm: $T = H \cdot m$, where m is the muscle activation signal vector processed as the input data of the AE (Fig. 4.3) [402];
- **Model** $HWW^{+}m$: $T = H \cdot W \cdot W^{+} \cdot m$, where H is exactly the same EMG-to-moment matrix extracted for the Model Hm [402] and W is the synergy matrix extracted with the NNMF by using the Matlab function $nnmf(\dots)$;
- **Model** $\hat{H}c$: $T = \hat{H} \cdot c$, where c is extracted with the NNMF for the model calibration and computed as reported in Equation 2.41 for the model evaluation [335];
- **Model** AE: $T = H_{AE} \cdot S$, where S is the synergy activation vector extracted by the autoencoder.

It is worth noting that the matrix H has dimension equal to $2 \times N$ (two is the number of moment components: shoulder and elbow joint moments), whereas the matrices $\hat{H}$ and H_{AE} have size 2×4 (four is the number of considered muscle synergies).

4.2.3 Model Calibration and Performance Metrics

Each subject-specific model has been independently trained with 256 (2^8) different train sets, where 2 is the number of reaching trials executed for each of the target sphere positioned in the 8 directions. Hence, a single train set contains the EMG and shoulder/elbow moment data acquired in one trial along all directions. Fixed a single train set, then all models have been evaluated on the complementary test set, that contains the data acquired during the contractions that have not been considered for the calibration.

The multivariate R^2 index has been computed for each test set in order to evaluate the synergy extraction performance of both the NNMF and AE. The multivariate R^2 index represents the fraction of total variation accounted by the synergy reconstruction and then is a global indicator of the goodness of reconstruction. The R^2 has been computed as follows [415]:

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_s \sum_{k=1}^{k_s} \|m_s(t_k) - m_s^r(t_k)\|^2}{\sum_s \sum_{k=1}^{k_s} \|m_s(t_k) - \bar{m}\|^2} \tag{4.1}$$

where SSE is the sum of the squared errors, and SST is the sum of the squared residual from the mean activation vector $\bar{m}$, i.e. the total variation multiplied by the total number of samples $K = \sum_s k_s$.

The shoulder and elbow articulation moment reconstruction by the models presented above has been evaluated computing the root mean square error (E_{RMS}) between the measured and estimated torques.

4.2.4 Statistics

In order to compare the proposed methods, the average values of the R^2 and E_{RMS} among the 256 test sets for each subject have been computed. The two synergy extraction methods, i.e., AE-based and NNMF, have been compared with the Wilcoxon test. The four moment estimator models, i.e. Hm, HWW^+m, $\hat{H}c$ and AE-based model, have been compared running the Friedman test and the Dunn's pairwise post-hoc tests with Bonferroni correction. The significance level has been set to 0.05. Non-parametric tests were adopted since the assumptions underlying parametric tests resulted to be violated for all sets of data. All the analyses have been performed using the SPSS[2] software (Version 21).

4.2.5 Results

In order to evaluate the ability of the proposed autoencoder to extract representative muscles synergies, the NNMF method and the autoencoder were compared in terms of the multivariate R^2 index computed between the processed measured muscle activations m_i and the reconstructed $\tilde{m}_i$ computed as $\tilde{m}_i = W \cdot W^+ \cdot m_i$.

Fig. 4.4 reports the average R^2 among the several test sets for each subject. It resulted that the R^2 of the muscle activation reconstruction based on the AE is significantly higher than the NNMF R^2 ($z = -2.201$, $p = 0.028$).

[2]https://www.ibm.com/analytics/spss-statistics-software

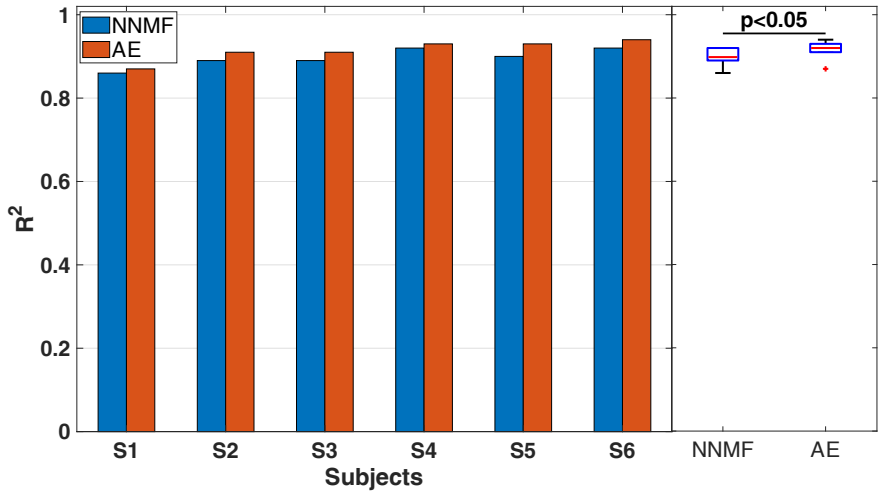


Fig. 4.4 Quality index of the muscle activation reconstruction.

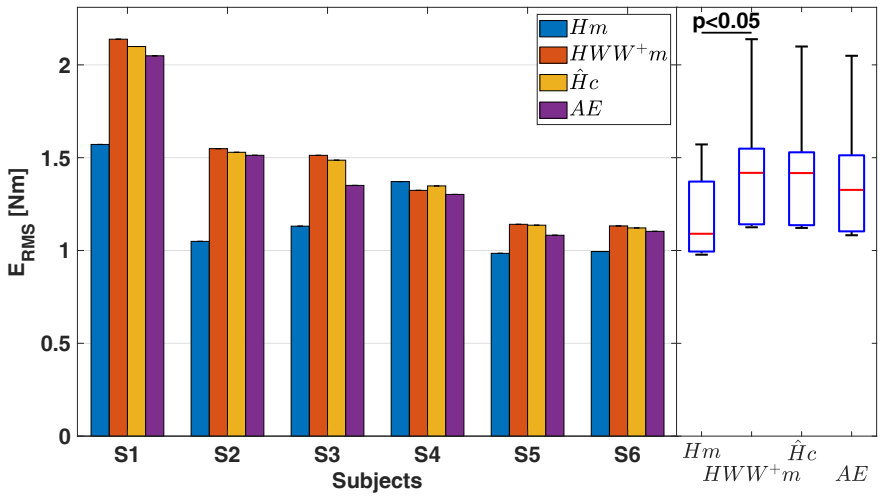


Fig. 4.5 Shoulder moment estimation error.

To validate the estimation ability of the method based on the muscle synergies extracted with the proposed autoencoder, the four models were compared in terms of E_{RMS} between the moments predicted by the model and reference (or measured) joint moments.

Fig. 4.5 and Fig. 4.6 report, for each subject, the averaged E_{RMS} among the several test sets. Concerning the shoulder joint moment estimation, the E_{RMS} of the four methods are significantly different ($\chi^2(3) = 11$, $p = 0.012$). Dunn tests with the Bonferroni correction were used to follow up this finding. It appeared that there is only a significant difference between the Model Hm and the model HWW^+m ($T = -2.167$, $p = 0.022$). Regarding the elbow joint moment estimation, no significant difference between the E_{RMS} of the four methods has been found ($\chi^2(3) = 7.8$, $p = 0.050$).

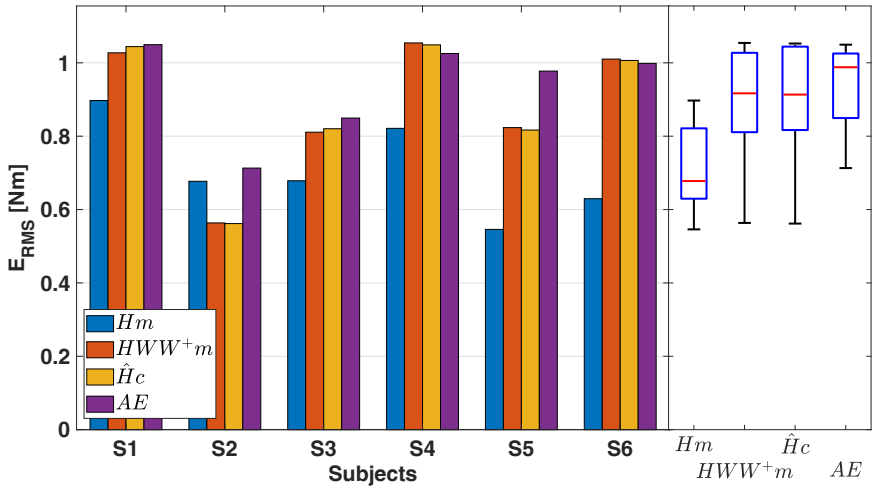


Fig. 4.6 Elbow moment estimation error.

Table 4.1 E_{RMS} performance Bonferroni corrected post-hoc comparisons for the shoulder joint.

Pairwise comparison	T Statistics	p-value
Hm - $HWW^{+}m$	-2.167	**0.022**
Hm - $\hat{H}c$	-1.500	0.265
Hm - AE	-0.333	1.000
$HWW^{+}m$ - $\hat{H}c$	0.667	1.000
$HWW^{+}m$ - AE	1.833	0.083
$\hat{H}c$ - AE	1.167	0.705

As reported in Fig.4.4, comparing the averaged R^2 values computed on the test sets, it resulted that the autoencoder performs better than the NNMF (Wilcoxon test, z=-2.201, p=0.028). This means that the AE generates synergy activations that better reconstruct the original muscle activation signals. It also worth noting that the AE and the NNMF have not been tested on the reconstruction of the same EMG signals used to calibrate the synergy model, but on different muscle activations acquired in the same condition, i.e. the same upper limb pose.

In conclusion, summarising the results, the statistical analysis revealed that the only found significant difference in estimating the shoulder moment is between the models Hm and $HWW^{+}m$ (Dunn test with the Bonferroni correction, $T = -2.167$, $p = 0.022$). The clear messages that arises from the statistical analysis, Fig. 4.5 and Fig. 4.6 are that: the Hm model is better than the $HWW^{+}m$, $\hat{H}c$ and AE models, even if such difference is not significant in the case of elbow moment estimation; the three methods $HWW^{+}m$, $\hat{H}c$ and AE performs

similarly. This result is reasonable since the Model *Hm* is not synergy based, i.e. it does not introduce any loss of EMG signal information, and it uses a bigger *H* matrix allowing a better learning. However this results are limited by the fact that the estimator has been calibrated and tested in the same upper limb pose. A synergy-based method should achieve better performances when a complex calibrated model is tested in different conditions [335].

4.3 Autoencoder for Task-Oriented Muscle Synergy Extraction

Even though a certain number of procedures for muscles synergy extraction has been proposed in the literature [407, 418], the main drawback of the existing approaches, such as the one presented before, concerns the fact that muscle synergies are estimated by analysing recorded muscle activities without having any information about neither the underlying task nor the final application. This means that the synergy extraction procedure considers the total variance reconstruction rate of the EMG signals as the only performance index to be optimized. Cristiano *et al.* [407] reported: *"We suggest that synergy extraction methods should explicitly take into account task execution variables, thus moving from a perspective purely based on input-space to one grounded on task-space as well ... In conclusion, we believe that the evidence reviewed here provides support for the existence of muscle synergies. However, many issues are still unresolved. A deeper investigation of the relationship between synergies and task variables might help to address some of the open questions"*. Few works have investigated the concept of functional synergies that are an initial attempt to link muscle synergies with task variables [407, 483–485]. However, as deeply discussed by Barradas *et al.* [486] and Cristiano *et al.* [407], functional synergies present some issues and limitations. After an extensive argumentation, Cristiano and his colleagues state that a novel required technique for muscle synergy extraction *"... should optimize the reconstruction error of the EMG signals, and constrain a good fit of the task-variables"*.

4.3.1 Model Design and Configuration

Starting from the architecture proposed in Section 4.2, a novel architecture that is able to learn the optimised muscles synergies patterns that lead to the optimized muscle synergy-based movement intention detection has been designed. The structure of the presented model (Fig. 4.7) is composed of two main blocks: an undercomplete autoencoder for muscle

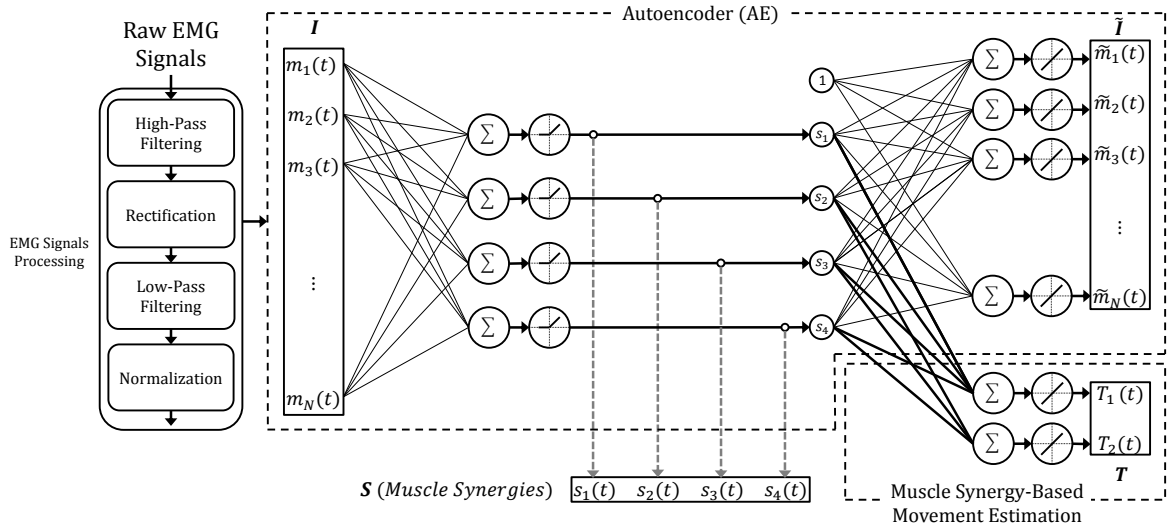


Fig. 4.7 The proposed extended autoencoder model.

synergies extraction and a feed-forward layer for movement estimation based on muscle synergy activations.

Referring to the Fig. 4.7, part of the modified autoencoder, works to learn and extract the muscles synergies patterns; in fact, given the input feature vector $I(t) = [m_1(t)m_2(t)\dots m_N(t)]$, where $m_i(t)$ indicates the pre-processed activation of the i-th muscle and N is the number of considered muscles, the AE preserves the objective to extract muscle synergy activations s_i. The topology has one hidden layer with four positive linear neurons that encode the muscle activations into synergy activations named $s_1(t)$, $s_2(t)$, $s_3(t)$ and $s_4(t)$. As before, the number of hidden neurons is set to four, and the overall configuration has been chosen with aim to replicate the physiological model of the spatial muscles synergies reported and deeply discussed in the work of Berger *et al.* [402]. The same three step pre-processing routine is also executed on each raw electromyographic signal: high-pass filtering (20 Hz second-order Butterworth); rectification and low-pass filtering (5 Hz second-order Butterworth); per-channel normalization over the maximum value computed during the calibration procedure.

Differently from the previous Section 4.2, the synergy-based movement intention detection has been achieved by adding a feed-forward block on top of the encoding hidden layer of the AE. By adding such block, the proposed neural model is able to compute the best muscle-synergy patterns that leads to the best trade-off between muscle activation reconstruction and movement intention estimation, that is, hand forces or articulation moment predictions. From the study of literature, this is the first attempt to build a model that is able to extract muscles synergies considering the performance into the task space, i.e. movement. Regarding the

activation function, the layer considers a linear function and no bias has been added. Such configuration allows for the computation of the forces/moments as a linear combination of the synergy activations. In detail, the output vector $T(t) = [T_1(t), T_2(t)]$ represents the estimated moments. It is important mentioning that the moment components $T_1(t)$ and $T_2(t)$ have been normalized to range within the interval $[-0.5, 0.5]$.

Network Training The network has been implemented using the Neural Network toolbox of Matlab (Release 2018b), and trained using a gradient descent with momentum and adaptive learning rate algorithm for 1000 epochs. Given a single training set, the training of the neural model has been repeated 10 times considering different initial weights [320], then the model featuring the best performance has been considered for the next analysis. Considering a training set composed of about 1000 time points, the training process of the model lasts about 4.5 *s*. All the training sequences have been run on a PC featuring two Intel XEON E5 2630 v3 CPUs and 64 GB of RAM.

Joint moment estimation based on muscle synergies: comparison with the state of the art. As the previous application, the performance of the bi-dimensional motion intention estimator has been compared with the same standard methods. In detail:

- **Model** Hm: $T = H \cdot m$, where m is the muscle activation signal vector processed as the input data of the AE (see Fig. 4.7) [402];
- **Model** $HWW^{+}m$: $T = H \cdot W \cdot W^{+} \cdot m$, where H is exactly the same EMG-to-moment matrix extracted for the model Hm [402] and W is the synergy matrix extracted with the NNMF by using the Matlab function $nnmf(\dots)$ (Release 2018b);
- **Model** $\hat{H}c$: $T = \hat{H} \cdot c$, where c is extracted with the NNMF for the model calibration and computed as reported in Equation (2.41) for the model evaluation [335];
- **AE-based model**: $T = H_{model} \cdot S$, where S is the synergy activation vector extracted by the autoencoder and H_{model} are the weights of the model block devoted to the synergy-based movement intention detection.

As before, the matrix H has dimension equal to $2 \times N$ (two is the number of moment components: shoulder and elbow joint moments), whereas the matrices $\hat{H}$ and H_{model} have size 2×4 (four is the number of considered muscle synergies). All methods have been tested using the same set of muscle activation recordings.

4.3.2 Results

In order to compare the proposed methods, the same model calibration, performance metrics and statistic methodologies used before were replicated. The proposed neural model has been evaluated both in terms of joint moment estimation and sEMG signal reconstruction. Fig. 4.8 (top-left and bottom-left) and Fig. 4.8 (top-right) report the mean value of the E_{RMS} and the mean R^2 values among all test sets for each subject, respectively. Table 4.2 reports the E_{RMS} and multivariate R^2 values averaged among all subjects for each compared methodology.

The Friedman test revealed that there is a significant difference among the four investigated techniques in terms of E_{RMS} relative to both shoulder moment prediction ($\chi^2 = 18.733$, $p < 0.001$) and elbow moment prediction ($\chi^2 = 15.000$, $p = 0.002$). Dunn test with Bonferroni correction was then used to perform the post-hoc tests (see Table 4.3). The results of the post-hoc analysis showed that the shoulder moment $\mathrm{E_{RMS}}$ error observed with AE-based model is significantly lower than both the errors obtained by the $HWW^{+}m$ model ($Z = 2.556$, $p < 0.001$) and $\hat{H}c$ model ($Z = 1.778$, $p = 0.021$). No significant differences were found between the AE-based model and Hm model ($Z = 1.222$, $p = 0.268$). Similar results were found analyzing the moment elbow $\mathrm{E_{RMS}}$ errors. In detail, the elbow moment $\mathrm{E_{RMS}}$ error observed with AE-based model is significantly lower than both the errors obtained by the $HWW^{+}m$ model ($Z = 2.111$, $p = 0.003$) and $\hat{H}c$ model ($Z = 1.778$, $p = 0.021$). No significant differences were found between the AE-based model and Hm model ($Z = 0.778$, $p = 1.000$).

The Friedman test also revealed that there is a significant difference among the four investigated techniques in terms of multivariate R^2 index between the measured and predicted joint moments, $\chi^2 = 21.400$, $p < 0.001$. The post-hoc analysis has reported that there is a significant difference between three pairs of models (see Table 4.3): the R^2 index of the AE-based model is higher than both the R^2 index of the $HWW^{+}m$ model ($Z = -2.667$, $p < 0.001$) and $\hat{H}c$ model ($Z = -1.889$, $p = 0.011$), respectively; the Hm model outperforms the $HWW^{+}m$ model ($Z = 1.667$, $p = 0.037$); then no significant difference was found between the AE-based model and the Hm model ($Z = -1.000$, $p = 0.602$).

The difference between the autoencoder and the NNMF algorithm were also assessed in terms of sEMG signals reconstruction quality by comparing the multivariate R^2 index between the acquired and reconstructed EMG signals (see Fig. 4.8 (bottom-right) and Table 4.4). The Wilcoxon test results showed that the NNMF achieved a significant higher R^2 index value than the autoencoder ($Z = -2.666$, $p = 0.008$).

To summurise, the statistical analysis on the data acquired from nine healthy subjects and the Fig. 4.8 revealed that: the proposed method outperforms the two synergy-based

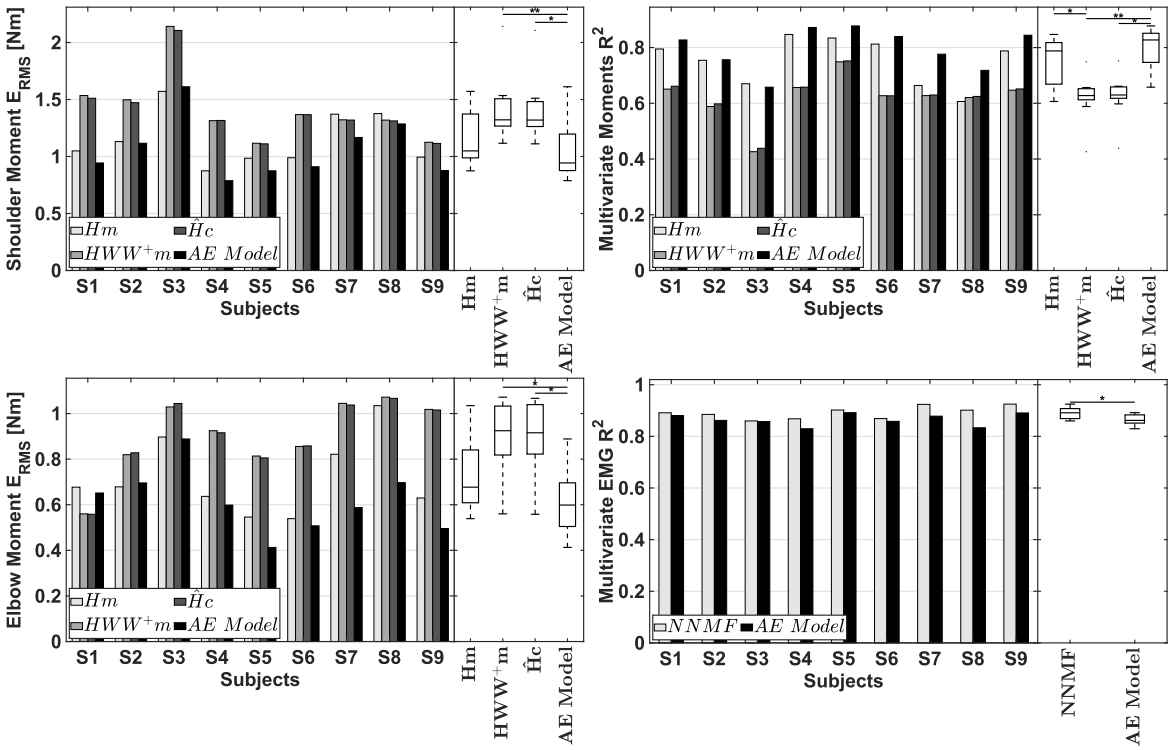


Fig. 4.8 Results averaged among the test sets for each subject and each compared technique/model. Shoulder moment E_{RMS} errors (top-left). Elbow moment E_{RMS} errors (bottom-left). Moment Multivariate R^2 index values (top-right). sEMG Multivariate R^2 index values (bottom-right).

Table 4.2 Means and standard deviations of the shoulder/elbow moment RMS errors and multivariate R^2 index values among all subjects.

Model	**Shoulder Moment RMS Error [Nm] (M $\pm$ SD)**	**Elbow Moment RMS Error [Nm] (M $\pm$ SD)**	**Shoulder and Elbow Moment Multivariate R^2 (M $\pm$ SD)**
Hm	1.15 ± 0.24	0.72 ± 0.17	0.75 ± 0.09
HWW^+m	1.42 ± 0.31	0.90 ± 0.16	0.62 ± 0.09
$\hat{H}c$	1.40 ± 0.30	0.90 ± 0.16	0.63 ± 0.08
AE model	1.06 ± 0.26	0.62 ± 0.14	0.80 ± 0.07

approaches HWW^+m and $\hat{H}c$ and such difference is statistically significant; no statistical difference has been found between the proposed method and the Hm model that considers a direct mapping between the EMG signals and the joint moments. Such findings seem promising since the proposed method is able to achieve the comparable performance of

Table 4.3 Results of the post-hoc analysis about the joint moments (p-values lower than 0.05 are in bold text).

	Shoulder Moment RMS Error		**Elbow Moment RMS Error**		**Moment Multivariate R^2**	
Pairwise Comparison	**Z**	***p*-Value**	**Z**	***p*-Value**	**Z**	***p*-Value**
$Hm - HWW^{+}m$	-1.333	0.171	-1.333	0.171	1.667	**0.037**
$Hm - \hat{H}c$	-0.556	1.000	-1.000	0.602	0.889	0.865
$Hm - AE\,model$	1.222	0.268	0.778	1.000	-1.000	0.602
$HWW^{+}m - \hat{H}c$	0.778	1.000	0.333	1.000	-0.778	1.000
$HWW^{+}m - AE\,model$	2.556	**<0.001**	2.111	**0.003**	-2.667	**<0.001**
$\hat{H}c - AE\,model$	1.778	**0.021**	1.778	**0.021**	-1.889	**0.011**

Table 4.4 Means and standard deviations of the sEMG multivariate R^2 index values among all subjects. Results of the Wilcoxon Test about the comparison between the non-negative matrix factorization (NNMF) and autoencoder.

Model	**sEMG multivariate R^2**	
	M $\pm$ SD	**Wilcoxon Test**
NNMF	0.89 ± 0.02	$Z = -2.666, p = 0.008$
AE	0.86 ± 0.02	

the Hm model even if introduces some loss in the EMG signal information due to the AE bottleneck.

About the quality of the muscle activity reconstruction, as reported in Fig. 4.8, it turned out that the proposed AE-based model has shown slightly lower performance than the NNMF (Wilcoxon test, $z = -2.666$, $p = 0.008$). This means that the NNMF generates synergy activations that better reconstruct the original muscle activation signals. This finding is not a big surprise since, differently from the NNMF, the proposed neural model has simultaneously focused on the reconstruction of both the EMG signals and joint moment. It is also worth noting that the AE and the NNMF have not been tested on the reconstruction of the same EMG signals used to calibrate the synergy model, but on different muscle activations acquired in the same condition (i.e., the same upper limb pose).

This work does not address the study of the relationship between the model accuracy and the number of considered muscles [401]. All the main superficial upper limb muscles that contribute to the shoulder and elbow moment generation have been considered [402]. Clearly,

a reduction in the number of considered muscles would lead to a loss of model accuracy, and such loss would be related to the functional contribution of the specific excluded set of muscles. A further study could investigate the role of the considered muscles in moment estimation when using the proposed approach. However, it is important to remark that the main goal was to propose a general methodology. The specific set-up (i.e. considered muscles, task-space variables and acquisition procedure) needs to be customized case by case.

In conclusion, the proposed work represents a first attempt to develop a muscle synergy-based myo-controller that is tailored to the specific subject by simultaneously considering the synergy extraction and the mapping between the synergy activations and the variables used in the task space, i.e. forces or moments. Concerning the specific experimental setup used in this study, the obtained results have clearly showed that the proposed model has lead to a better moment estimation when compared with other synergy-based models. However, at the same time, the quality of the EMG signals reconstruction was slightly degraded. This finding demonstrated that a trade-off between the capability of the extracted muscle synergies to better describe the EMG signals variability and the task performance in terms of force reconstruction might exist and can be exploited to develop more intuitive myo-controllers that are mainly evaluated in the task space [407, 486].

Chapter 5

Machine Learning and Deep Learning for Movement Disorder Analysis

This chapter will focus on the application of machine learning and deep neural networks for the development of CAD system pipelines able to objectify and follow-up movement disorder diseases. All the proposed solution were tested and validated on real clinical scenarios.

The following two section will focus on the analysis and development of two prototyped CAD system aiming to support the clinicians during the examination of patients affected by Blepharospasm (a focal dystonia) and Parkinson' disease. In detail, the blepharospasm CAD solution aims to help the standardisation of the disease assessment carried out by physician; the CAD developed for Parkinson' disease evaluation, instead, investigates the possibility to use handwriting analysis as a methodology to help physician with the assessment and grading of the disease.

Part of this chapter has been published in international conferences and journals [85–87, 89, 347, 487].

5.1 Deep Neural Networks for Blepharospasm Evaluation

In this section will be presented a CAD tool for blepharospasm evaluation. The proposed software try to overcome the main limitations of a preliminary approach conducted before this thesis research [439]. The analysis is based on standard video-recordings from commonly available video cameras, and it allows not only to measure the percentage time of eye closure, but also to recognise blinking, brief and prolonged spasms, which are the typical facial movements that take place in patients with blepharospasm. The proposed software is a practical system very suited to the clinical context where the environmental conditions cannot

be easily standardised; it is a promising tool for supporting/assisting physicians to rate blepharospasm severity according to the BSRS scale.

5.1.1 Materials

Nine patients with BSP were recruited (3 women and 6 men, average age 69.55 ± 8.94 years) to be recorded with a digital video camera[1] at 29.97 frames per second. An experienced neurologist reviewed the video-recordings identifying dystonic spasms and blinks, and evaluated the overall Severity Index (SIn) of the recruited patients by applying the BSRS scale. The neurologist classified the BSP symptoms observed in the recruited patients as follows (Fig. 5.1).

1. A sudden Orbicularis Oculi (OO) muscle contraction causing lowering of the eyebrow and narrowing/closure of the eyelid rim was classified by the neurologist as a spasm. In turn, the spasm was classified as:
   (a) brief spasm, i.e. a spasm inducing a brief eyelid closure lasting $0.3 - 3$ s;
   (b) prolonged spasm, i.e. a spasm inducing a prolonged eyelid closure with a duration greater than 3 s.
2. A bilateral, synchronous, short duration (< 1 s) OO muscle contraction causing a transient eyelid drop without any lowering of the eyebrow was classified by the clinician as a blink.
3. A delay in reopening the eyelids after involuntary closure associated with no overt OO contraction, or raising of the eyebrow above the superior orbital margin was classified as apraxia of eyelid opening (Fig. 5.1).

Data acquisition. The following clinical procedure was performed to determine the severity index SIn, according to the BSRS scale, of the nine recruited patients. The participants were seated on a chair placed in front of the video camera with their feet resting on the floor and their hands on their knees (Fig. 5.2). The camera objective was zoomed in so that the resulting field of view was entirely occupied by the patient's head and her/his shoulders. The video-recordings last approximately 5 minutes and were performed according to the protocol described in the preliminary study [431]. In the first three minutes, a training phase for the clinician took place; the neurologist asked the patient to perform some tasks and made

[1] Canon, Tokyo (Japan), Legria HFM306, 3.3 MP Full HD CMOS, HD Video Lens (up to 18× zoom), DIGIC DV III

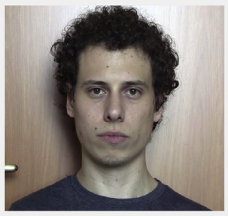
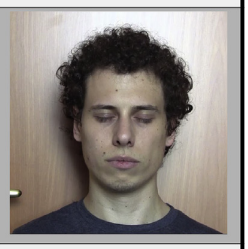
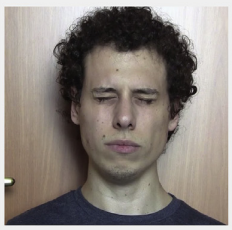
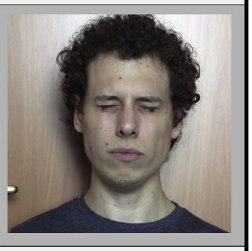
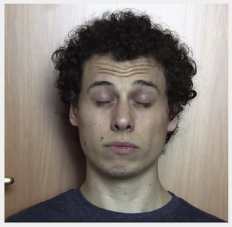
Symptom	Duration	Features	
Open eyes	-	-	
Blink	< 1s	transient eyelid drop without any lowering of the eyebrow	
Brief spasm	0.3 to 3 s	lowering of the eyebrow and narrowing/closure of the eyelid rim	
Prolonged spasm	> 3 s	lowering of the eyebrow and narrowing/closure of the eyelid rim	
apraxia of eyelid opening		raising of the eyebrow above the superior orbital margin	

Fig. 5.1 Example of typical symptoms observed in patients with blepharospasm.

careful observations, thus acquiring a deep knowledge of the modalities with which the BSP symptoms took place in the patient. During the last two minutes of the test, the patient was asked to remain at rest with their eyes open and fixed on a specific point located in front of her/him. During this time interval, the clinician recognises and counts blinks, brief and prolonged spasms, and apraxia of the eyelid opening.

In detail, the adopted clinical test is articulated in the following steps:

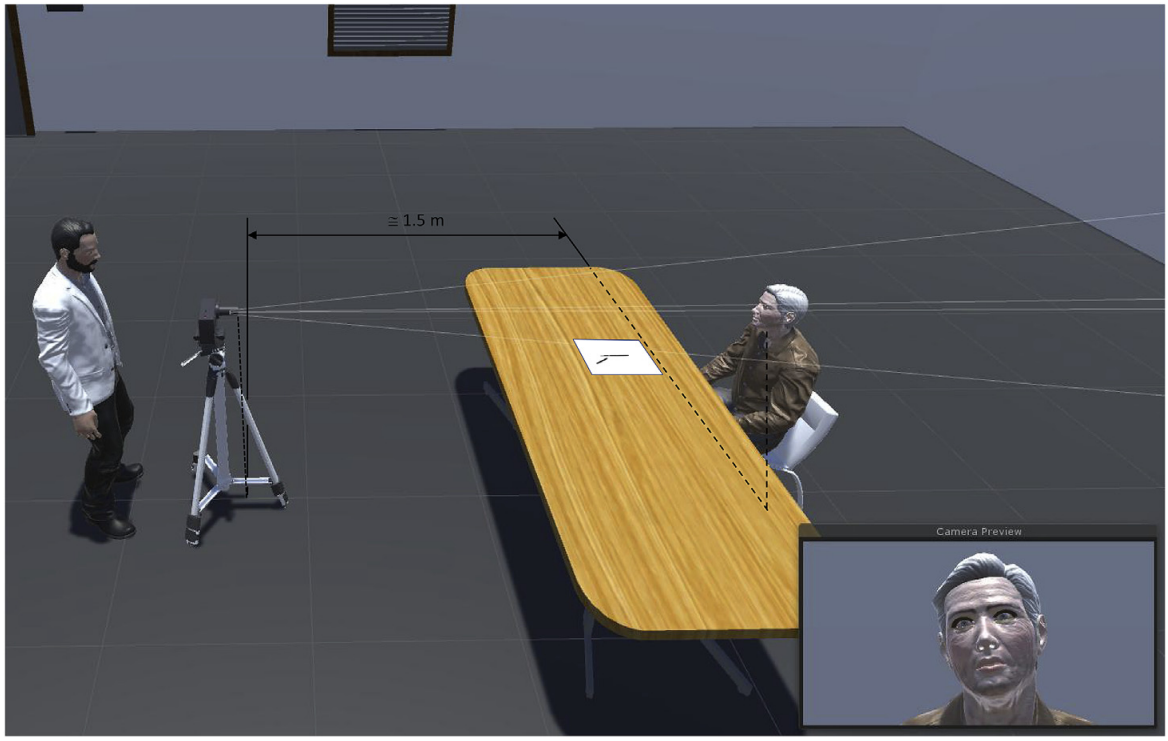


Fig. 5.2 Set-up utilised to acquire the facial expressions of the patient during the clinical test.

(i) the patient rests with open eyes for 10 *s*;

(ii) the patient is asked to voluntarily and forcefully close and open eyes five times, approximately one cycle per second (the time necessary to perform the requested task depends on the patient and the severity of the dystonia);

(iii) the patient rests with open eyes for 10 *s*;

(iv) the patient is asked to voluntarily and gently close and open eyes five times, one cycle per second;

(v) the patient rests with open eyes for other 10 *s*;

(vi) the neurologist poses the following questions: Are you able to avoid closing your eyes? How? With sheer will? Or, do you need to touch your eyes, face, or neck?

(vii) the patient has 50 *s* to reply to the posed questions;

(viii) the patient is asked to write three times on a sheet of paper a stereotyped sentence (e.g., "Today is a nice sunny day");

(ix) the patient remains at rest for at least 150 *s*, with eyes open and fixed on a specific point located in front of her/him. In the last 120 *s*, the neurologist "manually" counts the number of blinks, brief and prolonged spasms, and apraxia of eyelid opening.

Patients were instructed to avoid antagonistic gestures to not falsify the evaluation of the proposed software (i.e., manoeuvres voluntarily adopted by the patient to minimise the spasm effects, such as touching the eyelid, the temple, etc.).

The approximately first three minutes of the test included steps from (i) to (viii), whereas during the last approximately two minutes, only the step (ix) was performed. It is worth noting that the tasks performed in steps (ii) and (iv) are of crucial importance from the neurologist point of view. In fact, a voluntary and forceful closure of the eyes can provide useful information about how a spasm occurs in the patient. Similarly, a voluntary and gentle closure of the eyes is a type of simulation of a blink and, therefore, can instruct the clinician on how this event can take place in the patient.

5.1.2 Work-flow Design and Model Configuration

This section will reports all the steps designed and developed for the blepharospasm CAD evaluation tool; the extracted information allows to measure the duration of the time interval during which the patient's eyes were closed and to automatically recognise three of the four BSP symptoms described in Section 5.1.1, namely: blinks (2), brief (1a) and prolonged (1b) spasms; the last symptom, the apraxia of eyelid opening, was neglected in this study phase. The schematic overview about the principal steps followed to develop and validate the software are depicted in Fig. 5.3.

5.1.2.1 Face Detector and Face Pose Estimator

The developed software tool is based on the dlib library[2] and implements face detector and face pose estimator algorithms as pre-processing steps to reduce the region of interest and detect useful face landmarks.

Face detector algorithm. For each acquired frame (during the clinical test, Fig. 5.3, *Blocks 1*, *2* and *3*), the face detector algorithm, based on the traditional Histogram of Oriented Gradients (HOG) descriptor combined with a linear classifier in a sliding window detection scheme, identifies the bounding box in which the patient's head can be inscribed (Fig. 5.3, *Block 4*). Preliminary analyses revealed that the face detector was significantly robust with respect to head movements and lighting conditions. For each of the nine recruited patients, the percentage of frames ε_{FD}, in which the face detector algorithm was capable of reliably

[2]http://dlib.net/

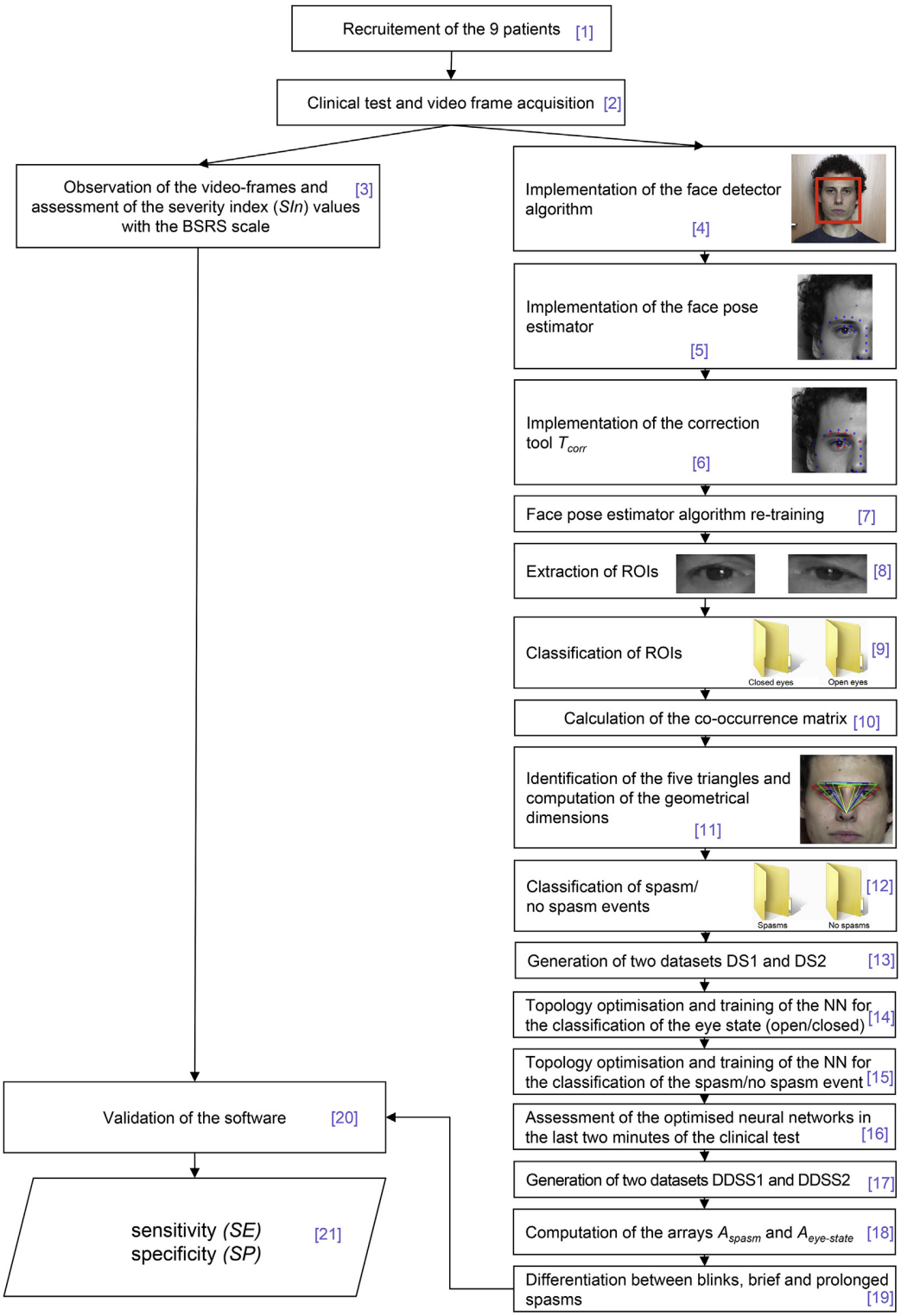


Fig. 5.3 Schematic of the steps followed to develop and validate the proposed software.

identifying the face of the patient, was computed with respect to the total number of frames acquired for the same patient. The average value of ε_{FD} computed over all the patients is 99.960%, whereas the lowest is 99.948%. Table 5.1 lists, for each of the nine patients, the number of frames acquired during the last two minutes of the clinical test, (i.e. during the approximative time interval in which the patient remained at rest with the eyes open and fixed on a point) and the corresponding value of ε_{FD}.

Table 5.1 Acquired frame and face detector results. ε_{FD} is the percentage of frames acquired during the last two minutes of the clinical test with a successful face detection.

Patient	**Total number of analysed frames**	**Number of frames in which the patient's face was detected**	$\varepsilon_{\mathbf{FD}}$
P1	*3897*	*3897*	*100.000*
P2	*3927*	*3925*	*99.949*
P3	*3927*	*3925*	*99.949*
P4	*3837*	*3835*	*99.948*
P5	*3867*	*3865*	*99.948*
P6	*3897*	*3895*	*99.949*
P7	*3927*	*3926*	*99.974*
P8	*3867*	*3865*	*99.948*
P9	*3897*	*3896*	*99.974*
	Average		***99.960***

Face pose estimator algorithm. The face pose estimator detects the position of 68 facial landmarks distributed at different points on the patient's face, such as the edges of the mouth, the eyes, the eyebrows, the nose, etc. (Fig. 5.3, *Block 5*; Fig. 5.4(a)). In detail, the pose estimator algorithm identifies firstly the position of specific face points that allow the definition of the principal facial features, then predicts the location of the 68 facial landmarks in near real-time. The pose estimator was created by using dlib's implementation of the study by Kazemi *et al.* [488] and was trained on the iBUG 300-W face landmark dataset [489].

The two algorithms described above were used to process the video frames acquired from all the patients. For each analysed frame, the face of the patient was firstly detected via the face detector algorithm, then the location of the 68 facial landmarks using the face pose estimator was extracted. Preliminary analyses revealed that, after a training phase, the face pose estimator algorithm was always capable of identifying the location of all the 68 landmarks in all the recorded videos. However, due to head movements and variable lighting

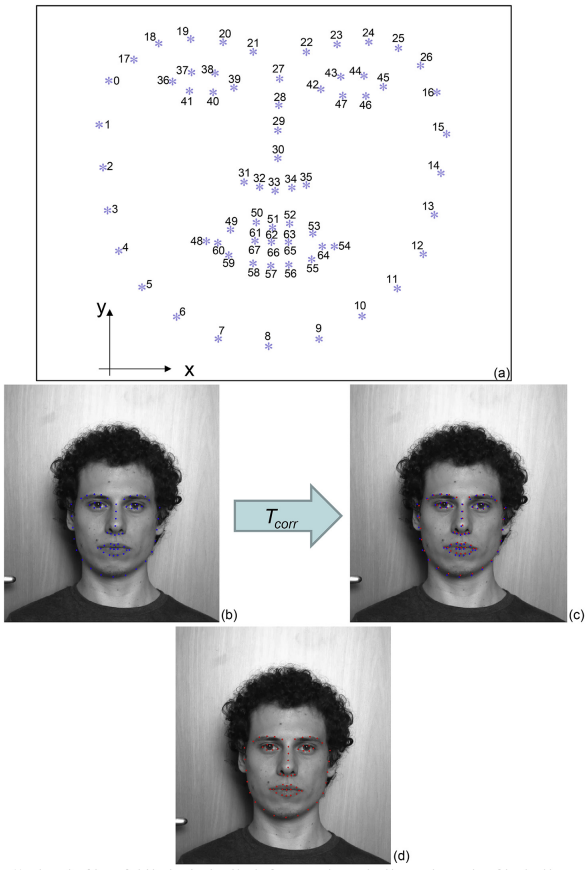


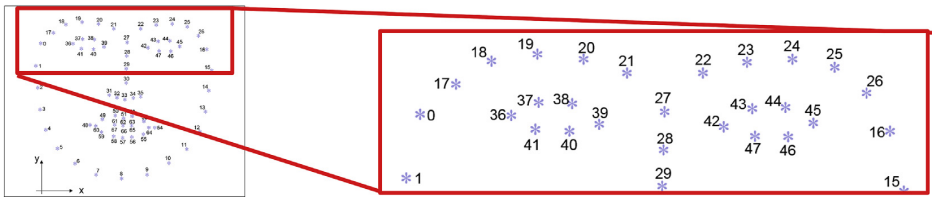
Fig. 5.4 Example of the application of the correction tool to improve the face landmarks stability. Schematic of the 68 facial landmarks placed by the face pose estimator algorithm (a). Implementation of the algorithm on one of the acquired frames before (b) and after (c) the application of the correction tool. The final results (d) shown more stable and more correctly positioned facial landmarks. The red and blue points represent the facial landmarks correctly and not correctly positioned, respectively.

conditions, the face pose estimator could not ever predict with precision the location of some facial landmarks (Fig. 5.4(b)). A correction tool, T_{corr}, was hence developed to allow the re-training of the pose estimator utilising the facial landmarks correctly re-located (Fig. 5.3, *Block 6*). The core of the tool is *imglab*, a dlib simple graphical tool for annotating images with object bounding boxes. The tool requires the clinician to choose a random number of frames (at least thirty), recorded during the first three minutes of the clinical test, showing the patient with eyes in several states (closed eyes during a spasm, closed eyes during a blink, and open eyes). For each frame the clinician can drag and drop, in a more correct position, all the facial landmarks not correctly positioned. The use of the tool allowed an overall better

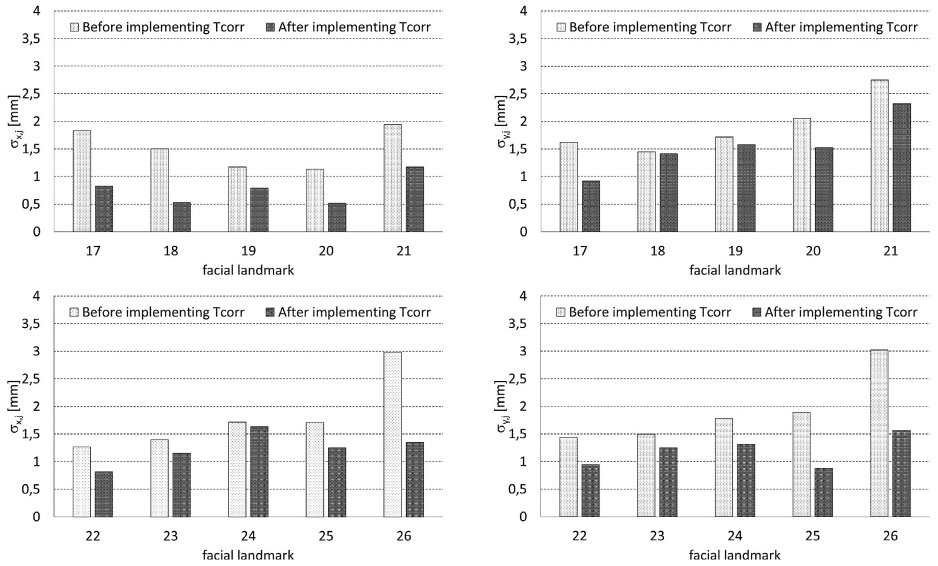
face pose estimator algorithm, thus making it capable of predicting the location of the 68 facial landmarks with higher accuracy (Fig. 5.3, *Block 7*; Fig. 5.4(b), (c), and (d)).

In order to test the reliability of the proposed correction tool, the coordinates x and y of the ten landmarks located on the two eyebrows were extracted (i.e., points marked in Fig. 5.5a from 17 to 21 for the right eyebrow and from 22 to 26 for the left one). For each patient, 10 clips with a duration of 0.3 *s* were considered. Therefore, the total number of frames considered for each patient and included in each portion of video was 9 (29.97 (acquisition frequency) $\times$ 0.3 (duration in seconds of the video portion)). For all the nine patients, the standard deviation of the x and y coordinates of points $17-21$ and $22-26$ detected in the 9 frames included in each portion of video was computed.

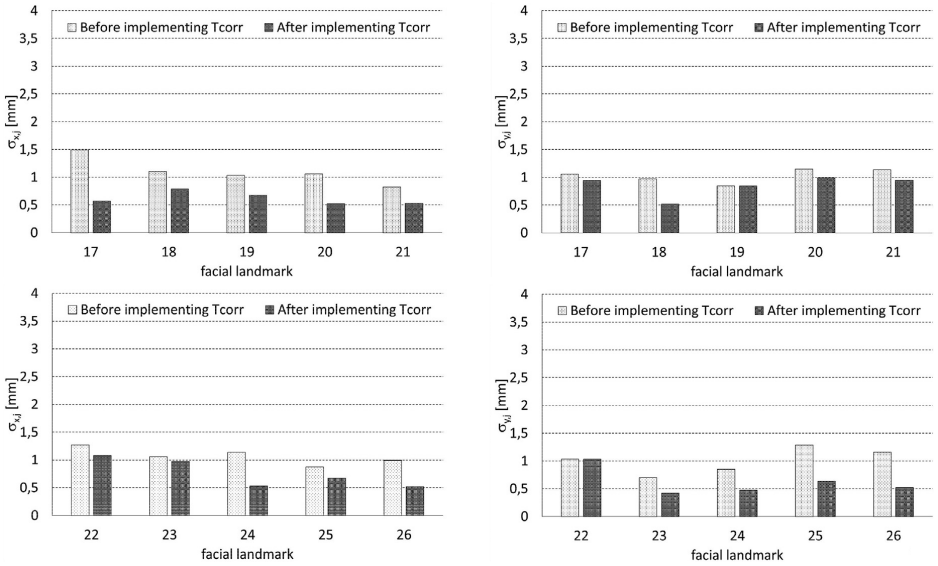
The duration of only 0.3 *s* was considered since it is the shortest time interval in which blinks of patients with blepharospasm can occur (brief and prolonged spasms have a longer duration). Electromyographic analyses revealed, in fact, that bursts of co-contracting activity in the facial muscles accompanying the involuntary movements, take place in patients with BSP and last from $200-300$ *ms* to many seconds [490]. Therefore, it is possible to state that in this short time, very small facial movements can occur, and hence, any changes experienced by the facial landmarks in this time interval can be justified only by the stability/accuracy of the face pose estimator. In other words, if the coordinates of a given landmark change in this short time interval, it will be due to the accuracy with which the face pose estimator algorithm predicts the landmark location and only marginally to the possible BSP symptoms occurring in this same time interval. In general, the larger the standard deviation, the larger the changes in the landmark coordinates and hence the lower the stability/accuracy of the face pose estimator algorithm. Conversely, the smaller the standard deviation, the higher the accuracy with which the algorithm predicts the location of the landmarks. Interestingly, the values of the standard deviation computed after implementing the correction tool are significantly lower than those obtained without its use (Fig. 5.5b), which demonstrates that it actually increases the predictive power of the face pose estimator algorithm. The values of standard deviation shown in Fig. 5.5b refer to video frames acquired in the first three minutes of the clinical test with and without the correction tool; the standard deviation computed on portions of video frames registered in the last two minutes of the clinical test lead to similar results (Fig. 5.5c). It is possible to conclude that the correction tool allows to increase the accuracy and the predictive power of the face pose estimator.



(a)



(b)



(c)

Fig. 5.5 Evaluation of the landmarks correction tool. Values of standard deviations of the x and y coordinates of the points $17-21$ and $22-26$ (a) computed in the first three minutes (b) and in the last two minutes (c) of the clinical test.

5.1.2.2 Features Extraction

Starting from the informations extracted with the algorithms presented before, eyes ROIs were subsequently defined to focus the BSP symptoms detection. Specific facial landmarks, predicted by the facial pose estimator, were selected and utilised to define the rectangular bounding box delimiting each eye (Fig. 5.3, *Block 8*). Concerning the right eye, points 36 and 39 were used to determine the horizontal dimension of the rectangle (Fig. 5.6). In detail, the horizontal dimension of the ROI goes from point $36'$, placed 30 pixels to the left of point 36, to point $39'$, placed 30 pixels at the right of point 39 (Fig. 5.6(a)). The same offset was used for the definition of the vertical dimension starting from points 38 and 41 to obtain points $38'$ and $41'$. The extraction of right eye ROI follows an analogue procedure by using points 42 and 45, and 44 and 46 for horizontal and vertical dimensions, respectively (Fig. 5.6(b)). For each frame the two extracted ROIs were resized to 64×32 pixels and saved in gray scale. In general, the dimensions of each extracted ROI depends on the eye size and, therefore, is patient-dependent. The resizing procedure enables all the ROIs to have the same dimensions, which is an essential requirement for the successive computation procedures described below.

The algorithm to crop the ROI frame by frame was implemented on the video registered from all the patients during the first three minutes of the clinical test, i.e., during the time interval in which the clinician asks the patient to perform some tasks and deeply observes her/his "response". Two categories were defined to gather the ROIs with open and closed eyes (Fig. 5.3, *Block 9*; Fig. 5.6 (c) and (d)). Then, the same neurologist that assessed the severity index SIn (according to the BSRS scale) of the nine recruited patients, manually labelled the obtained ROIs. It is worth noting that, although this classification must be performed for all the extracted ROIs, the described procedure is rather easy to accomplish and requires relatively little time. In the first three minutes of the clinical test, the patient gently or forcefully closes their eyes; then, most of the frames with open (or closed) eyes are close in time and, therefore, it is rather easy for the neurologist to gather frames with the same eye state.

For all the extracted and labelled ROIs, the co-occurrence matrices of oriented gradients were computed for the classification of the eye state (Fig. 5.3, *Block 10*). Indeed, the typical descriptive feature implemented in the computer vision for the eye state classification is represented by the histogram of oriented gradients (HOG), which is a useful and commonly utilised tool but suffers the limitation of local gradient information. The co-occurrence matrix of oriented gradients has been proven to enhance the capability to describe the global gradient information of eye images, thus allowing classification of the eye state with a higher accuracy than the classical HOG [491]. The matrix is a 4D array with dimensions given by

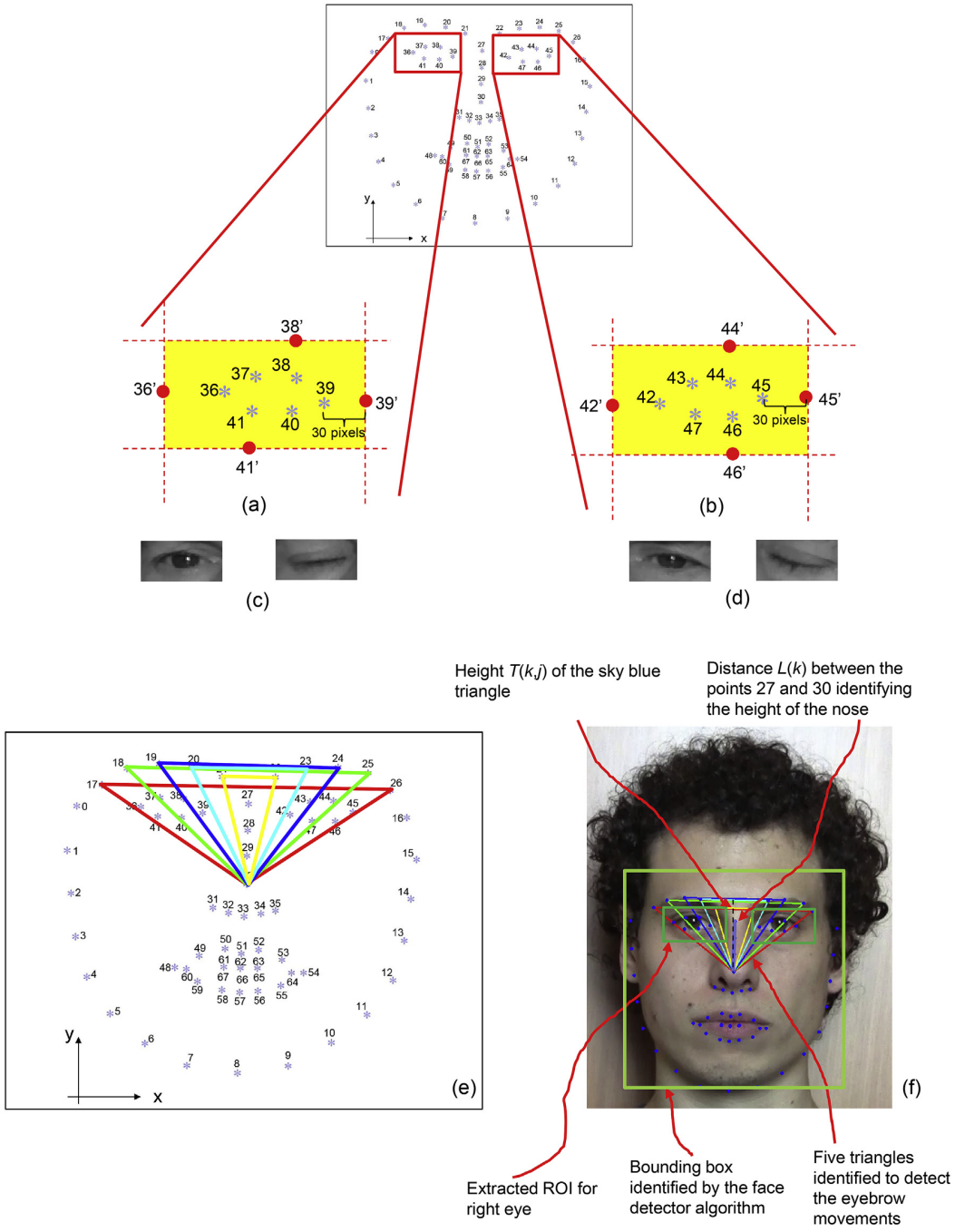


Fig. 5.6 Informations extracted from the face landmarks. Regions of interest extracted around the right (a) and the left (b) eye. Examples of ROIs extracted for the right (c) and the left (d) eye (open and closed). The five triangles with one of the vertices on the tip of the nose and the others defined by the facial landmarks located on the eyebrows identified to detect the eyebrow movements (schematic (e), over-imposition on an example face (f)).

gray levels $\times$ *gray levels* $\times$ *number of distances* $\times$ *number of angles*. In detail, a value at the coordinates xx, yy, zz, and ww, is the number of times the grey level yy occurs at the distance zz and at the angle ww starting from the grey level xx. Following Zhang *et al.* [491], the number of grey levels was set to 8, the distance to 1 pixel, and the angle to 0 radians; further details on the computation of the co-occurrence matrix can be found in Zhang *et al.* [491]. Therefore, the dimensions of the computed co-occurrence matrices, which represent the number of features that will be given in input to the artificial neural network described below, are: $8 \times 8 \times 1 \times 1$ for a total number of features $n_{Feye-state}$ equal to 64.

In order to detect the eyebrow movements related to the spasm events, an algorithm was developed to measure the height of the five triangles defined by the vertices on the tip of the nose (i.e. point 30) and the pairs of facial landmarks symmetric with respect to the sagittal plane and located on the eyebrows (Fig. 5.3, *Block 11*; Fig. 5.6 (e) and (f)). To avoid sudden changes in the height of the triangle due to possible rotations of the patient's head, the height was normalised with respect to that of the nose defined as the distance between points 27 and 30 (Fig. 5.6(f)). In detail, for each acquired frame k, the normalised height $Y(k,j)$ of the jth triangle ($j = 1,2,\ldots,5$), the average normalised height value $\overline{Y}(k)$ and the standard deviation $\sigma(Y(k))$, were computed as:

$$Y(k,j) = \frac{T(k,j)}{L(k)},\ j = 1,2,\ldots,5, \tag{5.1}$$

$$\overline{Y}(k) = \frac{1}{5}\sum_{j=1}^{5} Y(k,j), \tag{5.2}$$

$$\sigma(Y(k)) = \sqrt{\frac{\sum_{j=1}^{5}(Y(k,j) - \overline{Y}(k))^2}{5}}, \tag{5.3}$$

where, $T(k,j)$ is the height of the jth triangle and $L(k)$ is the distance between points 27 and 30. Fig. 5.7 and Fig. 5.8 show the average normalised height of triangles ($\overline{Y}(k)$) typically registered during a blink and a spasm, respectively. It is interesting to note how in the case of blinking (Fig. 5.7), the average height of the triangles remains practically constant, whereas a large decrease in $\overline{Y}(k)$ occurs during a spasm (Fig. 5.8).

The total number of features given in input to the artificial network described below is $n_{Fspasm} = 7$: the five normalised heights $Y(k,j)$, the average normalised height $\overline{Y}(k)$ and the standard deviation $\sigma(Y(k))$.

For the 'Spasms/No spasms' classification two category were additionally defined (Fig. 5.3, *Block 12*). According to its definition, the spasm event requires the eyes to

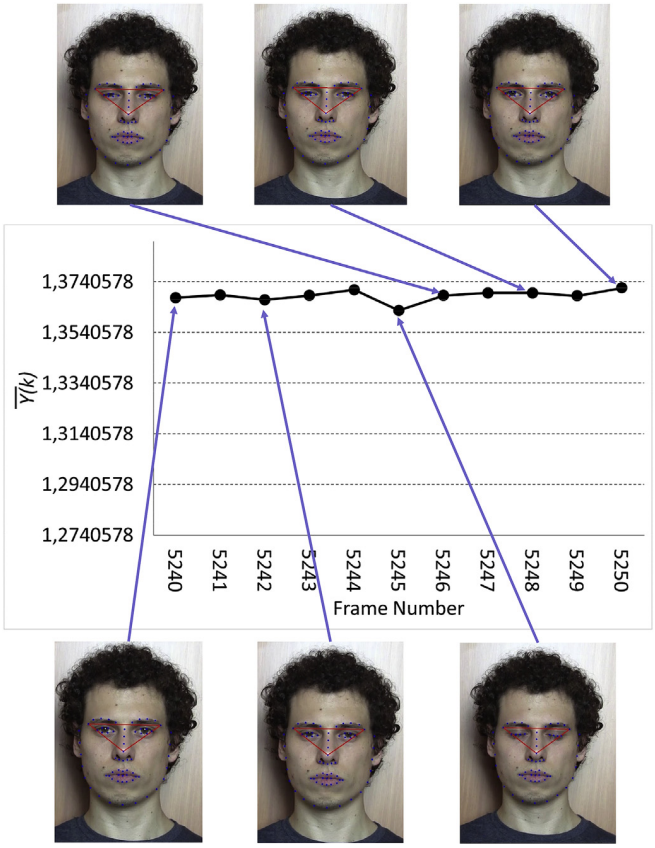


Fig. 5.7 Typical values of the average normalised height $\overline{Y}(k)$ of triangles registered during a blink.

be closed, then all the frames presenting closed eyes and acquired in step (ii), in which the patient was asked to voluntarily and forcefully close and open eyes five times, were automatically labelled as 'spasm'; on the contrary, the frames acquired before or after step (ii) presenting open eyes were labelled as 'no spasm'. The strategy of including in the category 'No spasms' the frames with open eyes is justified by the fact that when the patient has eyes open, the height of the five triangles does not change over time (which is the basic requirement for having the 'no spasm' event). It is worth noting that preliminary analyses revealed that during step (iv), when patients are asked to voluntarily and gently close and open their eyes five times, as a blink simulation, the height of the triangles can change significantly. Some patients, in fact, due to the pathology, were not able to "gently" close and open the eyes and, due to difficulties in the re-opening phase, often moved their eyebrows. To avoid these issues, the category 'No spasms' included only frames recorded in the first three minutes of the clinical test with the eyes open.

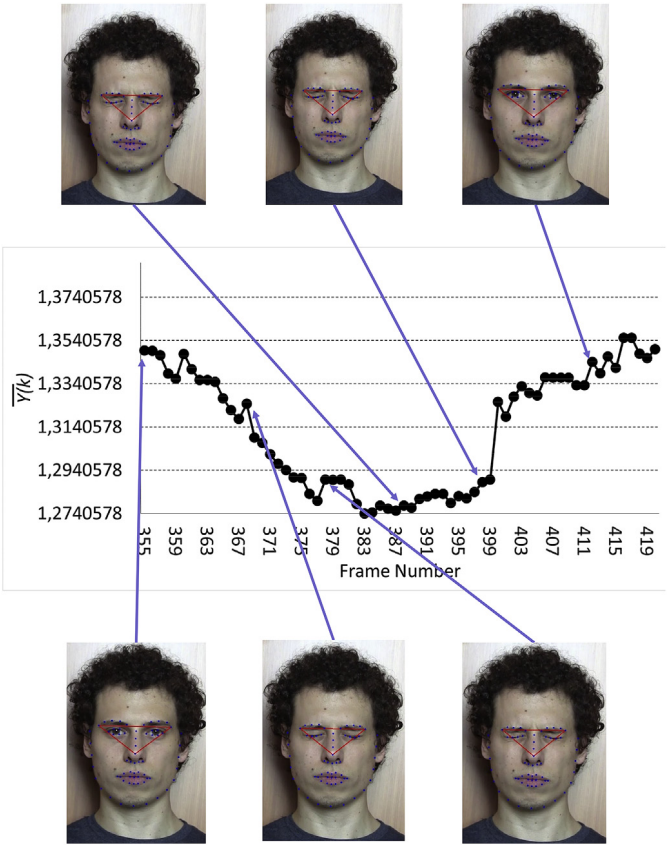


Fig. 5.8 Typical values of the average normalised height $\overline{Y}(k)$ of triangles registered during a spasm.

Two datasets were finally generated (Fig. 5.3, *Block 13*): the first one (DS1) includes 30160 entries and regards the classification of the eye state (16576 entries were labelled as 'closed eyes' and 13584 as 'open eyes'); the second one (DS2) includes 11266 entries and regards the classification of spasm/no spasm events (4474 entries were labelled as 'spasm', the remaining 6792 as 'no spasm'). The 64 $n_{Feye-state}$ and the 7 n_{Fspasm} features described before were used for DS1 and DS2, respectively. Table 5.2 lists, for each patient, the number of entries obtained for each of the two datasets.

5.1.2.3 Deep Neural Network Design

The two datasets DS1 and DS2 were given as input to two deep neural networks. The models were utilised to automatically classify blinks, brief and prolonged spasms observed in the nine recruited patients with BSP (Fig. 5.3, *Blocks 14* and *15*). In general, the performance exhibited by such classifiers is strictly dependent on their topology expressed in terms of number of hidden layers, neurons per layer and activation function per layer. Identifying the

Table 5.2 Dataset entries extracted from each patient and used as model input.

Patient	Entries 'closed eyes'	Entries 'open eyes'	Entries 'spasm'	Dataset eye state: Total entries 'closed eyes' + 'open eyes'	Dataset spasm/no spasm: Total entries 'spasm' + 'open eyes'
P1	*2666*	*1950*	*978*	*4616*	*1953*
P2	*1544*	*1634*	*421*	*3178*	*1238*
P3	*2930*	*848*	*788*	*3778*	*1212*
P4	*1666*	*2112*	*785*	*3778*	*1841*
P5	*2634*	*784*	*341*	*3418*	*733*
P6	*1162*	*1596*	*301*	*2758*	*1099*
P7	*1068*	*1750*	*279*	*2818*	*1154*
P8	*1736*	*1022*	*325*	*2758*	*836*
P9	*1170*	*1888*	*256*	*3058*	*1200*
		Total		***30160***	***11266***

optimal topology is a task of crucial importance, indeed, incoherent choices in the design phase can lead to unstable classification models with limited performance [78].

As for the study case presented in Section 3.2, the optimal neural network topology was designed by using an evolutionary algorithms. According to Bevilacqua *et al.* [353], a mono-objective genetic algorithm (MOGA) can be used as an optimisation strategy to design ANNs with optimal topologies. For this purpose, a binary chromosome of 30 bits was assembled to describe the following features characterising the topology of an artificial neural network:

- first hidden layer: number of neurons ranging in the interval $[1,256]$, coded with 8 bit;
- second and third hidden layers: number of neurons ranging in the interval $[0,255]$ (0 means no-layer), coded with 8 bit for each layer;
- first, second, and third hidden layers: activation function, coded with 2 bit for each activation function of each layer.

The four activation functions coded in the chromosome were: log-sigmoid (logsig), hyperbolic tangent sigmoid (tansig), pure linear (purelin) and symmetric saturating linear (satlins), whereas the activation function utilised in the output layer was the softmax function (softmax). The model hyper-parameters not encoded in the chromosome were fixed; the training algorithm for weights (W, Fig. 5.9) and bias (b, Fig. 5.9) update was the resilient backpropagation algorithm [492]. The parameters used in the genetic algorithm were: initial population with 100 randomly generated individuals, where each individual corresponds, practically, to a candidate ANN topology; crossover with two points and probability of 0.8; mutation with a probability of 0.2; elitism as selection system. The solution computed with

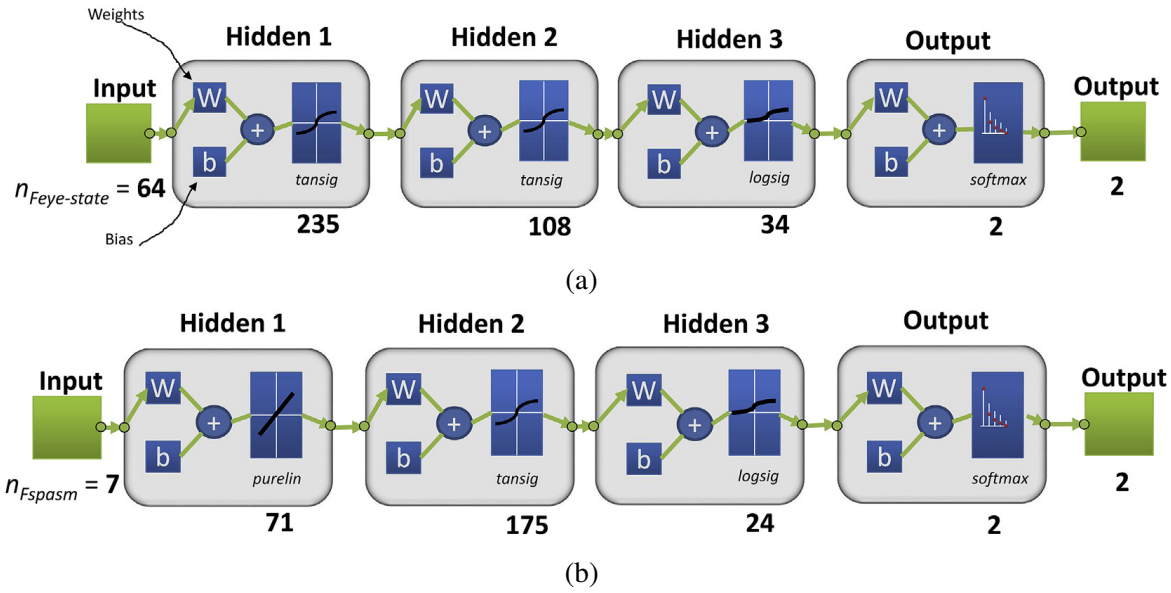


Fig. 5.9 Neural networks with optimised topology utilised to classify the open/closed eye state (a) and the spasm/no spasm event (b).

the genetic algorithm was the optimal ANN topology, which, after training, validation, and testing for a given number of iterations of different permutations of the input dataset, showed the highest mean accuracy.

The number of iteration was set to 200 and train, validation, and test datasets were obtained from the two input datasets as: 60% of the samples for training, 20% for validation, and 20% for the test.

The genetic algorithm described above was implemented to determine the optimal topology of two networks: the first for the classification of the eye state and the second for the spasm/no spasm event. For both of them, the genetic optimisation algorithm predicted an optimal topology with three hidden layers (Fig. 5.9). In detail, the optimal topology computed by the genetic algorithm for the eye state classifier included (Fig. 5.9a) 235, 108, 34, and 2 neurons for the first three hidden layers and the output one, respectively. The optimal activation functions were: tansig for the first and the second hidden layer and logsig for the third hidden layer. For the output layer, as stated above, the function was not optimised and was set to softmax. The optimal topology of the neural network for the classification of the spasm/no spasm event included (Fig. 5.9b) 71, 175, 24, and 2 neurons for the first three hidden layers and the output one, respectively. The activation functions were purelin, tansig, and logsig, for the first, the second, and the third hidden layer, respectively. Again, softmax was utilised for the output layer.

Table 5.3 Performance indexes over the 200 iterations. Results are expressed as mean ± standard deviation.

Datasets	Accuracy	Specificity	Sensitivity
Eye State (DS1)	*0.9641 ± 0.0015*	*0.9643 ± 0.0002*	*0.9637 ± 0.0026*
Spasm/No Spasm Event (DS2)	*0.9290 ± 0.0051*	*0.9507 ± 0.0007*	*0.8743 ± 0.0136*

5.1.2.4 Model Inference Criteria

The evaluation of the two optimised neural networks was performed with the frames acquired in the last two minutes of the clinical test when patients were asked to remain at rest with their eyes open and staring at a specific point located in front of them (Fig. 5.3, *Block 16*). For each acquired frame, the ROIs were cropped and the co-occurrence matrix and the heights of the five triangles were computed.

Two different subsets were created (Fig. 5.3, *Block 17*): the first dataset (DDSS1) was given in input to the neural network for the classification of the eye state, the second (DDSS2) to the neural network for the classification of the spasm/no spasm event. DDSS1 included a number of entries equal to twice the number of frames acquired in the two minutes (for each frame, two ROIs can be extracted), and DDSS2 equals to the number of frames acquired in the two minutes. Again, each entry of DDSS1 included $n_{Feye-state} = 64$ features, whereas each entry of DDSS2 included $n_{Fspasm} = 7$ features. Then, the following work-flow was implemented:

- giving DDSS1 as input to the first optimised neural network for the classification of the eye state, the frames containing closed eyes were first identified. Then, for each frame, the co-occurrence matrix, for both the right and the left eye, was computed and if one of the two matrices was predicted to be 'closed eye', the other one was automatically hypothesised to be the same. This is since BSP is a focal dystonia with bilateral and synchronous symptoms that simultaneously affect both eyes [428–430]. Therefore, the output of this first classification was an array $A_{eye-state}$ with length equal to the number of frames acquired in the two minutes of the clinical test. The output was codified as 0 in the case of open eyes and 1 in the case of closed eyes (Fig. 5.3, *Block 18*; Fig. 5.10);
- giving DDSS2 as input to the second optimised neural network, the frames where a lowering of the eyebrows took place were detected. The output of this second classification was the array A_{spasm} with the same length of $A_{eye-state}$ array, and assuming value 1 if an eyebrow narrowing occurs, 0 otherwise (Fig. 5.3, *Block 18*; Fig. 5.10);

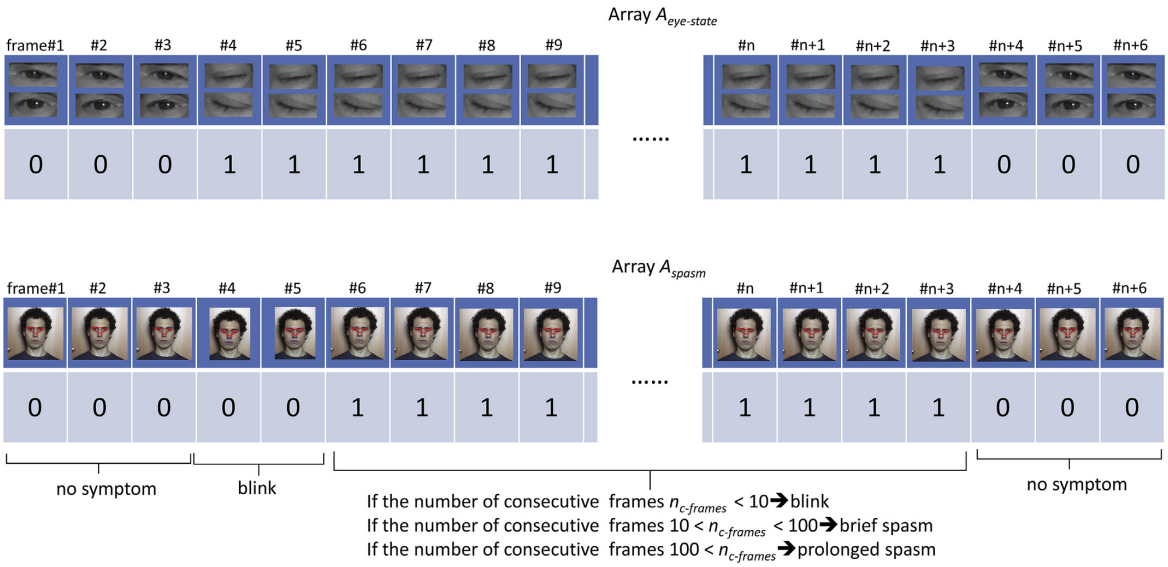


Fig. 5.10 Computation of the arrays $A_{eye-state}$ and A_{spasm}, and classification of the blepharospasm symptoms.

- the final step use the two generated arrays to distinguish spasms from blinks. The requirements for a symptom to be classified as spasm are eye closure, lowering of the eyebrows and a duration of at least 300 *ms*, which corresponds to the time necessary to acquire approximately 10 frames with the used camera. A symptom with a shorter duration can not be classified as a spasm but instead as a blink [490]. Furthermore, as stated before, a spasm lasting less than 3 *s*, (i.e. about 100 frames) must be classified as a brief spasm, and a spasm lasting longer than this as a prolonged spasm. Therefore, the frames where $A_{eye-state}$ assumes the value 1 (i.e. 'eyes closed') were considered and, in correspondence of these frames, the values of A_{spasm} were observed. If a set of less than 10 consecutive frames is characterised by $A_{eye-state} = A_{spasm} = 1$, then the entire set is classified as a blink. Instead, if a set includes more than 10 and less than 100 consecutive frames with $A_{eye-state} = A_{spasm} = 1$, then the set is classified as a brief spasm. If the condition $A_{eye-state} = Aspasm = 1$ is satisfied for a number of consecutive frames higher than 100, then the set is classified as a prolonged spasm. Finally, all the sets of consecutive frames with $A_{eye-state} = 1$ and $A_{spasm} = 0$ were classified as a blink (Fig. 5.3, *Block 19*; Fig. 5.10).

5.1.2.5 Validation Procedure

The last two minutes of the videos registered for the nine patients during the clinical test were manually segmented in clips lasting $10-20$ s. Each clip was trimmed to include just one of the following events: brief spasm, prolonged spasm, blink, or no involuntary eye closure (Fig. 5.3, *Block 20*). The segmentation was carried out by an independent expert neurologist who did not participate to the other steps of the study. Four category were then created: 'blinks', 'brief spasms', 'blinks + brief spasms', 'prolonged spasms', and clips showing no involuntary eye closure were also included. Therefore, for instance, the blinks category included all the clips of blinks as well as some of those showing no involuntary eye closures. Similarly, the category 'blinks + brief spasm' included the clips of brief spasms, blinks and some of those with no involuntary eye closures. The selected clips were then evaluated by the proposed software and the expert neurologist that participated to the firsts steps of the study (i.e. the neurologist that determined the severity index SIn of the nine recruited patients). The severity index evaluated by the proposed software was correlated with the eye closure time and with the severity index measured by the expert neurologist. In particular, the Spearman rank correlation coefficient was computed, which is a non-parametric measure of rank correlation. It allows the evaluation of how well the relation between two variables can be described using a monotonic function. Whereas Pearson's correlation assesses linear relationships, Spearman's correlation assesses, in general, monotonic relationships (that can be either linear or non-linear). Furthermore, the Spearman correlation is particularly suited to evaluate relationships involving ordinal variables (such as, the severity index SIn).

The sensitivity and the specificity (Fig. 5.3, *Block 21*) of the software were also computed for each category.

5.1.3 Results

The values of sensitivity and specificity computed for each detected symptoms are summarised in Fig. 5.11.

The clinimetric properties of the proposed software were assessed. For each patient, all the frames recorded in the last two minutes of the clinical test when the eyes were closed, were considered. Therefore, to determine the percentage of closure time for the investigated symptoms, the frames with closed eyes were distinguished depending on the symptom and counted. If f_{blink}, f_{bsp}, and f_{psp} are the numbers of frames showing blinks, brief spasms, and prolonged spasms, respectively, and f_{tot} the total number of frames registered in the last two

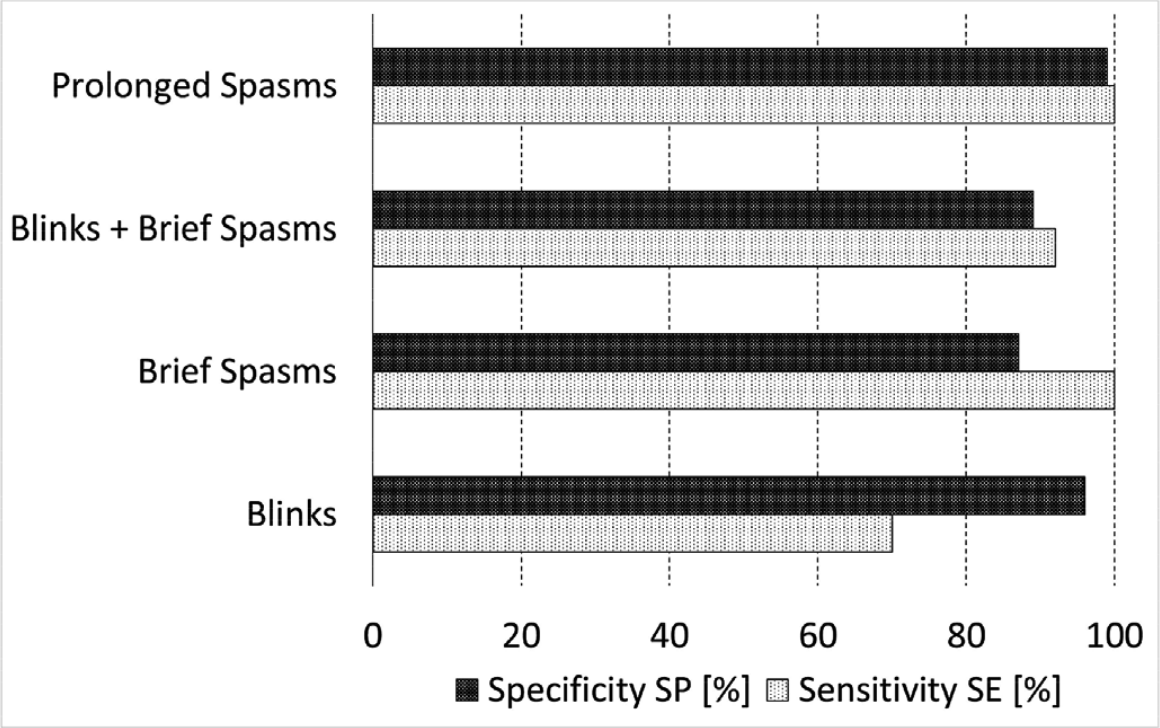


Fig. 5.11 Values of sensitivity and specificity obtained with the proposed software for the different investigated symptoms.

minutes, the percentage of closure time for blinks t_{blink}, brief spasms t_{bsp}, and prolonged spasms t_{psp} can be computed as follows:

$$t_{blink} = \frac{f_{blink}}{f_{tot}} \times 100 \tag{5.4}$$

$$t_{bsp} = \frac{f_{bsp}}{f_{tot}} \times 100 \tag{5.5}$$

$$t_{psp} = \frac{f_{psp}}{f_{tot}} \times 100 \tag{5.6}$$

The values of percentage of closure time were computed for all the patients and correlated with the severity index values SIn evaluated by the expert neurologist according to the BSRS scale (Fig. 5.12a). The values of the severity index were also correlated with the percentage of total closure time. If f_{totce} is the total number of frames registered in the last two minutes

Table 5.4 Spearman correlation coefficients computed between the scores extracted by the software and those determined by the expert neurologist.

BSRS Item	Features	Spearman rho	p-value
A1	Type of eyelid spasm	*0.793*	*0.019*
A2*	Apraxia of eyelid opening	*N/A*	*N/A*
A3*	Spasms occurring during the writing of the stereotyped sentence	*N/A*	*N/A*
A4	Average duration of the prolonged spasms	*0.806*	*0.009*
B1	Frequency of blinks + brief spasms	*0.676*	*0.046*
B2	Frequency of prolonged spasms	*0.756*	*0.030*
Total 'measurable' severity index SIn_m	$SIn_m = S(A1) + S(A4) + S(B1) + S(B2)$	*0.863*	*0.003*

*Items A2 and A3 were not measurable by the proposed software.

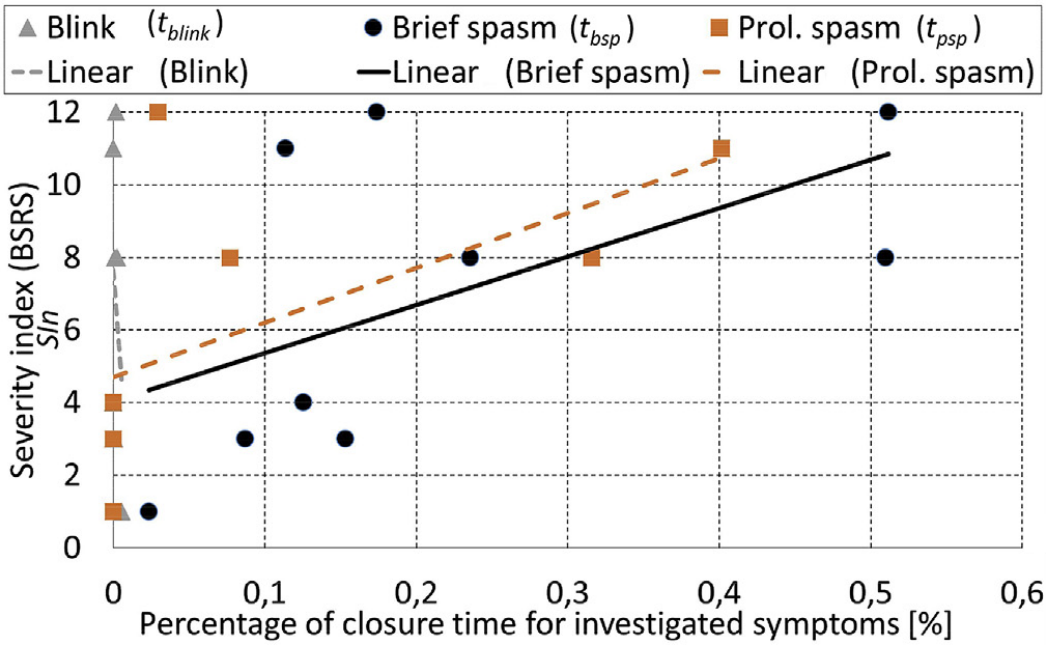
of the clinical tests and characterised from having closed eyes, the percentage of total closure time t_{tot} can be computed as:

$$t_{tot} = \frac{f_{totce}}{f_{tot}} \times 100 \tag{5.7}$$

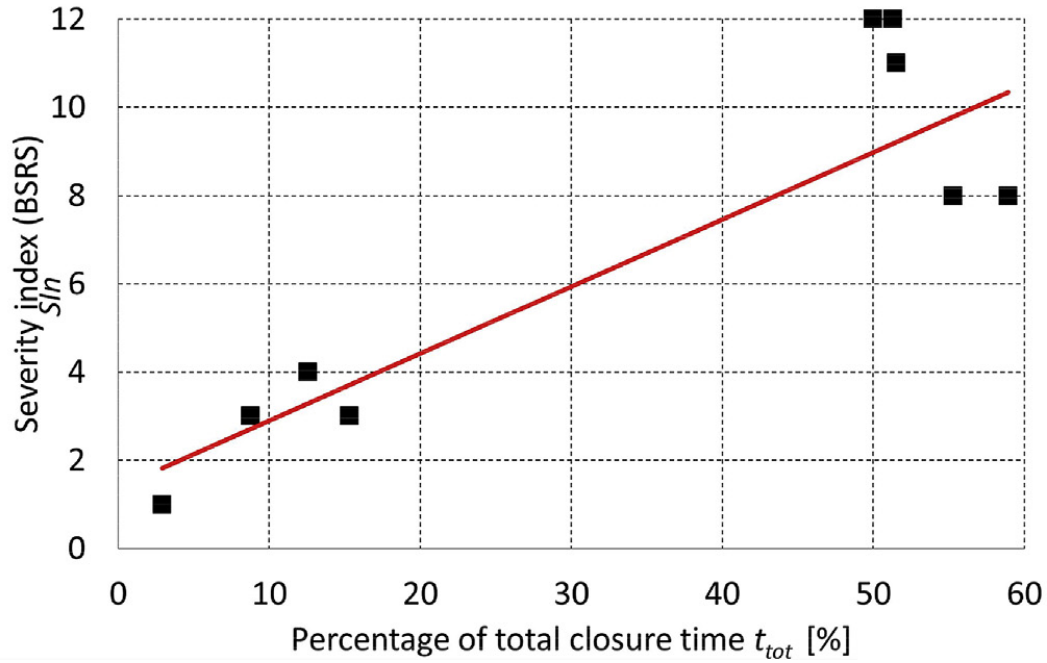
The values of the severity index were reported as a function of t_{tot}, and a linear regression line was also included in the graph of Fig. 5.12b.

It is worth noting that the proposed software is not capable of evaluating the complete BSRS scale. The BSRS scale, in fact, includes six items, for each of which a score S must be assigned according to specific criteria (further details on the BSRS scale can be found in Section 2.4.4.1). Among others, the BSRS includes items regarding apraxia of eyelid opening (A2) and spasms occurring during the writing of the stereotyped sentence (A3). Due to how the system has been designed, it is not possible to assigning a score for the items A2 and A3. However, considering only the remaining items and summing up the scores given to each of them, the severity index SIn_m (measurable) computed by the software shows consistency with the corresponding values determined by the expert neurologist (Spearman rho 0.863, p-value 0.003) (Fig. 5.13). Significant values of the Spearman correlation coefficients can also be found considering the score given to individual items computed by the software and the score determined by the expert neurologist (Table 5.4).

Analysing the results it is possible to affirm that the software shown high sensitivity for prolonged spasms and lower but satisfactory for brief spasms (Fig. 5.11). Lower values



(a)



(b)

Fig. 5.12 Correlation graphs between: the severity index SIn determined by the expert neurologist and the percentages of closure time for the investigated symptoms (a); the severity index SIn determined by the expert neurologist and the total closure time (b).

of sensitivity were found in the case of blinks; a confusion related to the imperceptible difference between blinks and brief spasms was probably responsible for this result. Proof

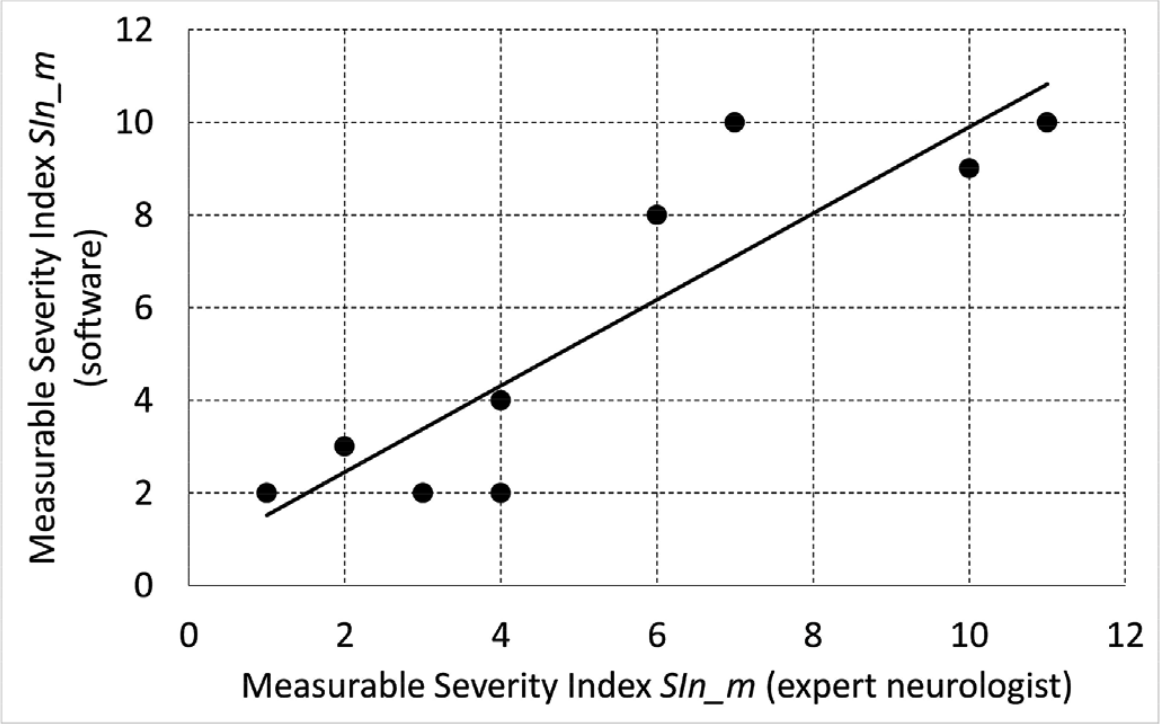


Fig. 5.13 Correlation between the measurable severity index SIn_m computed by the software and that determined by the expert neurologist.

of this is given by the satisfying level of sensitivity computed in the case of blinks and brief spasms combined in the same category. The high levels of specificity demonstrate the capability of the proposed software to distinguish the non-pathological conditions.

High Spearman correlation coefficients were computed for brief (Spearman rho 0.684, p-value 0.042) and prolonged (Spearman rho 0.783, p-value 0.022) spasms (Fig. 5.12a). A very low correlation coefficient was, by contrast, found in the case of the blinks, which indicates that no clear correlation exists between t_{blink} and the severity index SIn values. Considering the relative smaller importance that blinks have in this severity scale rather than the other symptoms (only item B1 partially depends on the number of blinks), it is possible to guess that blinks affects marginal the severity index values, leading to a lower correlation. A significant correlation coefficient was found between the severity index SIn values and t_{tot} (Spearman rho 0.735, p-value 0.038) (Fig. 5.12b).

The proposed software tool presents some limitations. First, although the software exhibits high values of sensitivity in distinguishing brief spasms, prolonged spasms, and blinks + brief spasms, the value computed for blinks is small. This can be justified by the fact that the software sometimes confuses blinks with brief spasms. However, it is worth noting that the difference between the symptoms of the blink and the symptoms of the brief spasm

is subtle and, as often occurs in clinical practice, the task of distinguishing the two symptoms is complex even for neurologists. This is true especially in the case of brief spasms that have a duration close to the threshold value of 300 *ms* [490] that separate brief spasms from blinks. During the clinical evaluation, the neurologist does not physically measure the time, thus, increasing the probability of confusing the two symptoms. The proposed software instead takes into account the exact number of frames included in the set showing the symptom under investigation and can measure the time with an accuracy of approximately 0.03 s. Furthermore, the distinction of blepharospasm symptoms is often subtle and imperceptible. It is commonly known, in fact, that different expert neurologists can assign different severity index SIn values to the same patients. Further investigations should be carried out on this topic. The second limitation of the study is represented by the time necessary to use the software on the specific patient, which includes: 1) the time necessary to utilise the correction tool T_{corr} (the neurologist is asked to drag and drop the incorrect facial landmarks on at least 30 frames); 2) the time necessary to re-train the face pose estimator after implementing T_{corr}; 3) the time necessary to extract, from the acquired frames, all the entries to give in input to the neural networks. Preliminary investigations revealed that for an experienced neurologist, all these tasks require approximately and on average 30 minutes. However, considering that an experienced neurologist spends approximately $60-90$ minutes per patient observing all the acquired video clips, it is possible to conclude that implementing the software allows to save time ensuring the reproducibility of the results. Furthermore, the proposed software was not conceived to 'replace' the neurologist but to 'assist/support' in defining the severity of BSP. From this point of view, the limitation of the time necessary to use the software is relevant but it is abundantly counterbalanced by the important advantage of making the process of evaluating the BSP symptoms objective. The third limitation of the study is represented by the fact that the datasets given in input to the deep neural networks for the optimisation of their topology are unbalanced (Table 5.2). A measure of the quality of binary classifications useful in this situation is the Matthews Correlation Coefficient (MCC) already defined (Eq. 2.20) and used in the previous study cases. The average MCC obtained for the two optimised models are: $MCC = 0.9612 \pm 0.0016$ for the classification of the eye state; $MCC = 0.8481 \pm 0.0055$ for the classification of the spasm/no spasm event.

In conclusion, despite these limitations, it is possible to state that a correlation exists between the total closure time t_{tot} and the severity index values SIn (Fig. 5.9), which is consistent with the results of Peterson *et al.* [435]. Furthermore, a very high level of ε_{FD}, the percentage of frames with face detected by the face detector algorithm, was found. The lowest percentage of face-found frames was $\varepsilon_{FD} = 99.948\%$, which is higher than Peterson

et al. results (93%). Finally, it is worth noting that currently, the only computerised and automatic system capable of rating the blepharospasm severity is represented by the toolbox CERT [435], which is capable of measuring the eyes closure time but cannot recognise, and hence count, the specific BSP symptoms. The developed software, instead, allowed a separate evaluation of the contribution of the individual symptoms to the global severity index, thus opening up new perspectives in the problem of evaluating/measuring BSP symptoms; it is an automatic tool capable of making the 'measurement' of BSP symptoms objective and, hence, assisting/supporting the neurologist in rating the severity of the dystonia.

5.2 Deep Neural Networks for Biometric Handwriting Analysis to Support Parkinson's Disease Assessment and Grading

In this section will be investigated the possibility to use handwriting analysis as a methodology to help physician with the assessment and grading of Parkinson' disease.

Two main study, and as many CAD tool, will be presented investigating how the joining of dynamic features extracted from sEMG and handwritten text/drawing features, can be used for patients evaluation. The first study make use of features extracted from scanned paper sheets analysed by exploiting vision-based features, whilst the second one investigates advantages and drawbacks of features generable by using tablets as input device.

5.2.1 A model-free technique based on computer vision and sEMG for classification in Parkinson's

A preliminary research conducted by Loconsole *et al.* proposed a reduced set of features, extracted by exploiting surface electromyography signal processing and computer vision techniques applied on the scan of common paper sheets, to differentiate PD patients from healthy subjects [493]. To advance in the proposed direction, a larger number of features (also considering the muscular activity) were addressed to improve classification performance, and allowed to investigate five specific research scientific questions related to the handwriting analysis capacity of discriminate healthy subject from patients suffering from Parkinson's disease: (1) *which are the most representative features?*; (2) *which is(are) the best writing pattern(s)?*; (3) *which is the best AI-based classification approach between ANN optimal topology and SVM in terms of accuracy?*; (4) *which is the best AI-based classification*

approach between ANN optimal topology and SVM in terms of repeatability of the result?; (5) *which is the effect of considering the most representative features in the classification process?*

The results allowed to infer some important properties on writing patterns, classification approaches and the role of muscular activities on the handwriting analysis applied to Parkinson's disease research.

5.2.1.1 Materials

Eleven participants (all males, age: 48 ± 25 years old) took part to the experimental tests. In detail, the control group was composed of 7 healthy subjects (age: 31 ± 11 years old), whereas the PD group was composed of 4 subjects (age: 77 ± 3). All subjects signed informed consent forms.

System set-up and experimental description. All the subject were asked to perform three writing patterns corresponding to as many writing tasks. They were properly differentiated according to a writing size constrained/unconstrained point of view; in detail:

1. a fixed sentence in Italian composed of 8 words all containing equal sized letters (e.g., no use of *d,f,g,h,l,p,q,t* letters) to be written in italic;
2. a sequence of 8 *"l"* with a size of 2 *cm* (2 *cm* visual marker reference on the left of the paper sheet);
3. a sequence of 8 *"l"* with a size of 5 *cm* (5 *cm* visual marker reference on the left of the paper sheet).

Writing task no.1 can be considered not constrained in size (the letter size is up to the patient) while tasks no. 2 and 3 are size constrained, thus forcing the subject to write using a predefined size. To familiarize with the exercise, each subject was asked to perform all three writing tasks once. Then, each subject was asked to perform 4 repetitions of all 3 writing tasks, thus resulting in a total of 12 writing samples per subject. According to the proposed acquisition protocol, the subject was asked to rest for at least 3 s between each writing task.

The set-up and the paper sheet template used for the experimental tests are illustrated in Fig. 5.14. In detail, in Fig. 5.14 (left), it is possible to see the Myo^{TM} Gesture Control Armband[3] for acquiring EMG signals from 8 different points of the forearm. It needs a simple and a brief calibration to start the acquisition.

[3] www.myo.com

In Fig. 5.14 (right), instead, on the printed template of the paper sheet, it is possible to identify two vertical visual marker of 2 *cm* and 5 *cm* used as size references for writing tasks no. 2 and 3, respectively, and three horizontal 1 *cm* markers used for spatial mapping needed for processing.

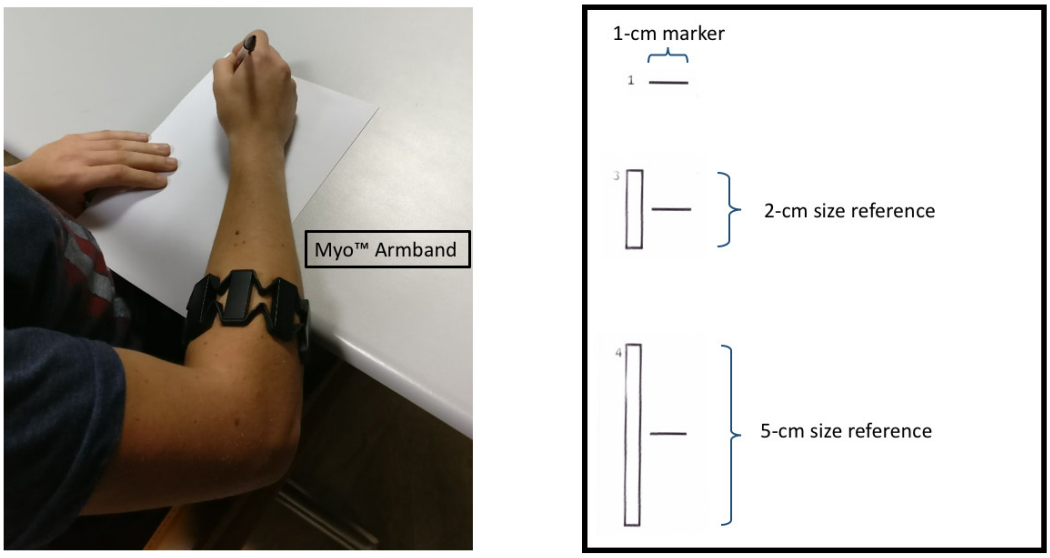


Fig. 5.14 System set-up used for the experimental tests to validate the proposed approach (left). The paper sheet template with two vertical reference marker for writing tasks no. 2 and 3 and three horizontal markers used for spatial mapping.

5.2.1.2 Handwriting Feature

To further investigate the entire handwriting analysis process with respect to the preliminary research work [493], a larger number of features were extracted from biometric signals. In detail, eight dynamic and two static feature types were selected for a total of ten types of features: static - (i) *density ratio*; (ii) *height ratio*; dynamic: (iii) *execution time*; (iv) *execution average linear speed*; (v) *acceleration norm* - mean, standard deviation and maximum value; (vi) *gyroscope component* - mean, standard deviation and maximum value; (vii) *Root Mean Square (RMS)* of the sEMG signal; (viii) *Mean Absolute Value (MAV)* of the sEMG signal; (ix) *Zero Crossing (ZC)* of the sEMG signal.

In next paragraphs, the selected feature will be analysed.

Density ratio feature. Density ratio is a static feature. The variation of the pixel density as defined by Zhi *et al.* [494] (dimensionless value) can represent a potential assessment index of micrography. To extract this feature, firstly the entire written sentence is subdivided in an arbitrary number of cells characterized by having the same width and the height determined by the text upper and lower bounds. In this case, the number of cells is set to: (i) $3 \times number\ of\ words$ in the case of sentence writing pattern; (ii) $number\ of\ letters$ in the case of single word writing pattern. The cell width is, then, calculated as the ratio between the entire width of the sentence and the obtained number of cells. Secondly, black pixels contained in each cell are counted and the result is divided by the area of the belonging cell, thus obtaining the cell density value. In case of micrography, since the size of the written text is progressively reduced from left to right, the expected result is that the ratio of the density of the first cells and that of the last cells (density ratio) is greater than 1. Specifically, since it is possible to suppose that the first cell contains a capital letter (thus resulting in a larger cell area), for the density ratio feature extraction, the ratio between the second and the last cells was used. In writing tasks no. 2 and 3, since they consisted in writing only one word composed of 8-*"l"*, each written text was subdivided in 8 cells. For writing task no. 1, instead, the number of cells was set to 24 (8×3).

Height ratio feature. Height ratio is a static feature corresponding to a measure of the user's ability to maintain the writing task fixed in size (dimensionless value). The feature extraction process is similar to the density ratio, especially for cell subdivision. However, the feature value corresponds to the ratio between the height of the second cell and that of the last cell. Also in this case, micrography should result in a height ratio value greater than 1. In writing task no. 1 the height ratio feature was calculated as described, while in tasks no. 2 and 3 the height ratio was calculated as the ratio between the average height of the 8 *"l"* and the expected letters height, that are 2 *cm* and 5 *cm*, respectively.

Execution time feature. Execution time is a dynamic feature and is the time interval (in seconds) during which the subject performs the writing pattern. The feature extraction from the EMG signal can be performed off-line (i.e., after the completion of the task) and is based on an adaptive threshold to detect the starting and the end instants of time of the writing task. In detail, since the proposed acquisition protocol requires a rest period among each writing tasks, the adaptive threshold is computed as the highest peak value of the EMG signal during rest time both before and after the writing task.

Execution average linear speed feature. Execution average linear speed is a dynamic feature and is the ratio between the sum of the width of each written word and the execution time (as reported by Raudmann *et al.* [495]). It is measured in cm/s and as reference, for the spatial measurement of the word width, the 1 *cm* visual marker on the paper sheet has been used (Fig. 5.14 (right)); thus resulting in a feature extraction process independent from the scanning device and its resolution.

Acceleration norm features. Acceleration norm features are a set of dynamic features. The norm of the signal is computed from the three acceleration data components. The resulting signal is, then, processed in order to obtain three feature values: mean, standard deviation and maximum values of the acceleration signal.

Gyroscope components features. Gyroscope component features are a set of dynamic features. The three gyroscope components k ($k \in \{yaw,\ pitch,\ roll\}$) have been independently used to compute three feature values for each of them (for a total of 9 feature values of this type): mean (Eq. 5.8), standard deviation (Eq. 5.9) and maximum (Eq. 5.10).

$$\mu_k = \frac{1}{n}\sum_{i}^{n} x_{ki}, \tag{5.8}$$

$$\sigma_k = \sqrt{\frac{1}{n-1}\sum_{i}^{n}(x_{ki}-\bar{x}_k)^2}, \tag{5.9}$$

$$M_k = max(x_{ki})_{i=1,..,n} \tag{5.10}$$

RMS of sEMG signals features. Root Mean Square (RMS) is a dynamic feature. It is a time domain feature and the value is computed for each sEMG channel according to Equation 5.11.

$$RMS = \sqrt{\frac{1}{n}\sum_{i=1}^{N}|x_i|^2} \tag{5.11}$$

MAV of sEMG signals features. Mean Absolute Value (MAV) is a dynamic feature. It is a time domain feature and the value is computed for each sEMG channel according to Equation 5.12.

$$MAV = \frac{1}{n}\sum_{i=1}^{N}|x_i| \tag{5.12}$$

ZC of sEMG signals features. Zero Crossing (ZC) is a dynamic feature. It is the number of signal sign variation. Considering two consecutive samples x_k and x_{k+1}, the ZC value is incremented if $x_k > 0$ and $x_{k+1} < 0$, or $x_k < 0$ and $x_{k+1} > 0$ and $|x_k - x_{k+1}| \geq tol$. A tolerance value (*tol*) is used in the last condition to avoid variation due to noise signal. *tol* corresponds to the standard deviation of the sEMG signal acquired over the rest time.

5.2.1.3 Feature Extraction

Regarding the feature extraction, for the static features (density and height ratio), classical computer vision techniques (e.g., morphology operators, image segmentation, etc.) were used to process the scanned paper sheet compiled by the subjects during the tests. In particular, the mean and the deviation standard values were computed for both density and height ratio features over the 4 repetitions for each subject, thus obtaining a total of 4 static feature values per subject. For the dynamic features, instead, the followings were extracted:

- *execution time.* Mean and standard deviation values of the execution time over the 4 repetitions for each subject, to obtain a total of 2 execution time feature values per subject;
- *execution average linear speed.* Mean and the standard deviation values of the execution average linear speed over the 4 repetitions for each subject to obtain a total of 2 values of execution average linear speed feature per subject;
- *acceleration norm* from the 3 acceleration components acquired with the Myo Armband. In particular, for each subject and for each of the four repetitions, the mean, the standard deviation and the maximum values of the acceleration norm were extracted. Then, the average of the four obtained mean, standard deviation and maximum values were computed, thus resulting in a total of 3 acceleration norm feature values per subject;
- *gyroscope component* from the 3 speed components acquired with the Myo Armband. In particular, for each subject and for each of the four repetitions, the mean, the

standard deviation and the maximum values of each of the three gyroscope components were extracted. Then, for each gyroscope component, the average of the four obtained mean, standard deviation and maximum values were computed, thus resulting in a total of 9 gyroscope component feature values per subject;

- *Root Mean Square (RMS)* of the sEMG signals acquired with the Myo Armband. In particular, this feature was computed for each of the 8 sEMG signals made available by the Myo Armband over the 4 repetitions for each subject (8 RMS feature values per subject);
- *Mean Absolute Value (MAV)* of the sEMG signals acquired with the Myo Armband. In particular, this feature was computed for each of the 8 sEMG signals made available by the Myo Armband over the 4 repetitions for each subject (8 MAV feature values per subject);
- *Zero Crossing (ZC)* of the sEMG signals acquired with the Myo Armband. In particular, this feature was computed for each of the 8 sEMG signals made available by the Myo Armband over the 4 repetitions for each subject (8 ZC feature values per subject);

Thus, the total number of the feature values per subject is equal to 44 of which 4 static and 40 dynamic.

5.2.1.4 Feature Reduction and Classification

To understand which are the most representative features among the extracted ones, the principal component analysis was exploited. Regarding writing pattern classification, instead, two different artificial intelligence-based approaches were proposed and compared. The *first approach* relies on finding the optimal topology for an deep neural network classifier by exploiting the Multi-Objective Genetic Algorithm already used in this thesis, and by maximizing the average test accuracy on a certain number of training, validation and test iterations for each ANN topology using different permutations of the dataset.

For each input dataset, the MOGA was executed to find the optimal topology in terms of: (i) number of hidden layers (ranging from 1 to 3), (ii) number of neurons per layer (ranging from 1 to 256 for the first hidden layer, and from 0 to 255 for other hidden layers), and (iii) activation functions in *log-sigmoid* (logsig), *hyperpolic tangent sigmoid* (tansig), *pure linear* (purelin) and *symmetric saturating linear* (satlins), for all the neurons per-single layer. The GA was set-up using the following parameters:

- best ANN accuracy computed on test set as leading search criterion;

- initial population size equals to 50 individuals;
- crossover with 2 points with a probability of 0.8;
- mutation with a probability of 0.2;
- elitism selection system;
- stop criteria set to maximum generations numbers (100) or 20 consecutive generations with fitness value unchanged.

The *second approach*, instead, relies on Support Vector Machine (SVM). SVM attempts to construct an optimal separating hyperplane in the feature space by maximizing a geometric margin between points from the two classes. SVMs demonstrate good generalization performance [496].

The performance of the classifiers were evaluated in terms of accuracy (Eq. 2.13), specificity (Eq. 2.16) and sensitivity (Eq. 2.15).

Experimental data processing description For addressing the five specific research scientific questions reported in the previous subsection, the following scheme has been used during experiments and results evaluation:

- creation of two different feature value sets: the first (set A) including the feature values extracted from all three writing patterns, the second (set B) including the feature values extracted from writing patterns no.2 and no.3;
- application of the PCA on both sets to reduce the features;
- creation four different feature dataset cases:
  Case 1. Dataset with all 44 feature values (40 dynamic, 4 static) extracted from set A;
  Case 2. Dataset with all 44 feature values (40 dynamic, 4 static) extracted from set B;
  Case 3. Dataset with only PCA-obtained feature values from set A;
  Case 4. Dataset with only PCA-obtained feature values from set B.
- application of both AI-based classification approaches on all four dataset cases;
- evaluation of accuracy, specificity and sensitivity (mean values expressed in percentage and standard deviations) obtained with ANN optimal topology approach for each of the four dataset cases;
- evaluation of average performance (mean values expressed in percentage, standard deviations, maximum an minimum values) obtained with the SVM approach for each of the four dataset cases.

5.2.1.5 Results

Two samples of one repetition of all three writing tasks respectively performed by a healthy and a PD subject are reported in Fig. 5.15. In the following paragraphs, will be reported and discussed the results obtained with the PCA and with the AI-based classification algorithms to address the five research scientific questions presented before.

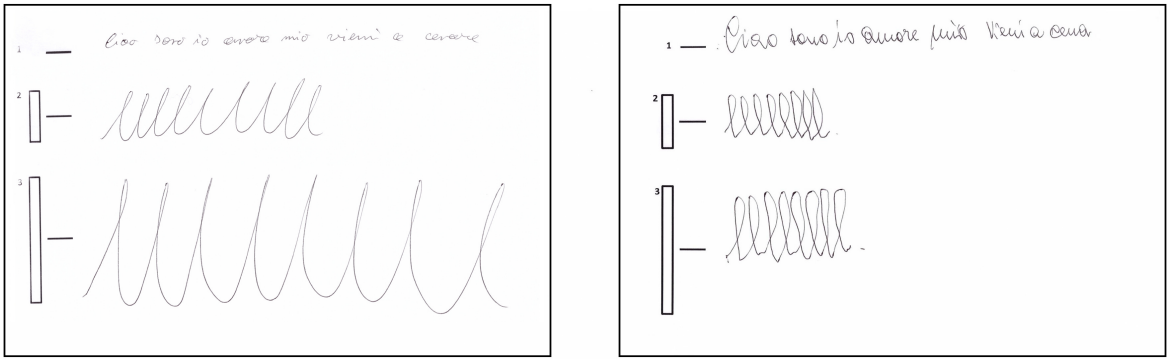


Fig. 5.15 Two samples of a repetition of all three writing tasks respectively performed by a healthy subject (left) and by a PD subject (right).

PCA results. The feature reduction based on PCA has been set up to hold 99.9% of the original information. The new space generated by the PCA is composed by only 8 feature values for the set A and 9 for the set B. To better understand which of the features from the original space are more representative, the new generated space has been analysed by means of *biplot* graphs. In particular, the observed most representative feature values for both sets were the ones related to all sEMG signals and, in particular, the ZC features extracted from the eight sEMG channels. It is worth to mention that ZC feature values computed on signals acquired by adjacent sEMG sensors have similar significance. The output features obtained from PCA have been used for creating the feature dataset of Cases 3 and 4.).

Deep neural network results. Regarding the first approach for feature classification, the genetic algorithm allowed to select the best performing ANN in terms of number of layers, optimal number of neurons per layer and activation function for each layer. The ANN optimal topologies specified by the genetic algorithm in the four cases are:

Case 1. Dataset with all 44 feature values (40 dynamic, 4 static) extracted from all three writing patterns

ANN (Fig. 5.16) with: 3 hidden layers, with 101, 189 and 2 neurons for the hidden

layer and 2 neurons for the output layer. The activation function found by the GA was *logsig* for the hidden layers. For the output layer, the *softmax* function was preliminary selected as activation function. Accuracy: 86.15% ($std = 0.1258$), specificity: 0.9152 ($std = 0.1451$), sensitivity: 0.7720 ($std = 0.2518$).

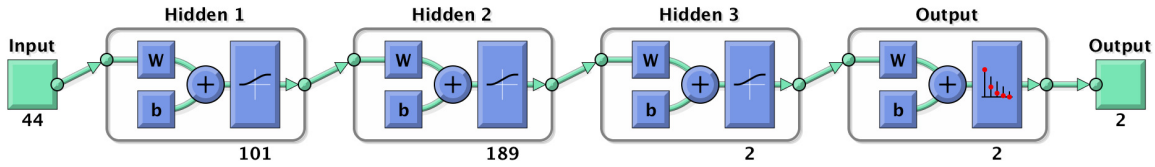


Fig. 5.16 The ANN optimal topology for Case 1.

Case 2. Dataset with all 44 feature values (40 dynamic and 4 static) extracted from two writing patterns (no.2 and no.3)

ANN (Fig. 5.17) with: 3 hidden layers, with 50, 5 and 1 neurons for the hidden layer and 2 neurons for the output layer. The activation function found by the GA was *logsig* for the hidden layers. For the output layer, the *softmax* function was preliminary selected as activation function. Accuracy: 80.24% ($std = 0.1667$), specificity: 0.9227 ($std = 0.1896$), sensitivity: 0.6220 ($std = 0.3587$).

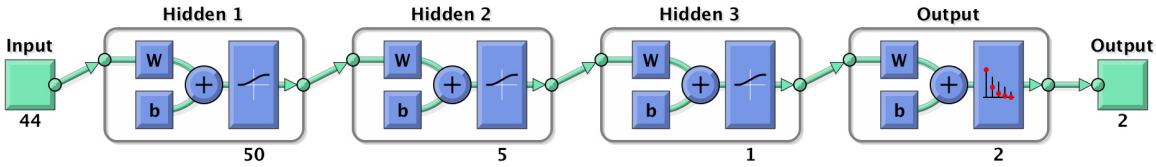


Fig. 5.17 The ANN optimal topology for Case 2.

Case 3. Dataset with only 8 dynamic feature values obtained from PCA extracted from all three writing patterns

ANN (Fig. 5.18) with: 2 hidden layers, with 26 and 8 neurons for the hidden layer and 2 neurons for the output layer. The activation function found by the GA was *logsig* for the hidden layers. For the output layer, the *softmax* function was preliminary selected as activation function. Accuracy: 75.75% ($std = 0.1512$), specificity: 0.8504 ($std = 0.1815$), sensitivity: 0.6027 ($std = 0.2993$).

Case 4. Dataset with only 9 dynamic features obtained from PCA extracted from two writing patterns (no.2 and no.3)

ANN (Fig. 5.19) with: 3 layers, with 117, 145 and 4 neurons for the hidden layer and 2 neurons for the output layer. The activation functions found by the GA were *purelin, tansig and logsig* for the hidden layer. For the output layer, the *softmax* function

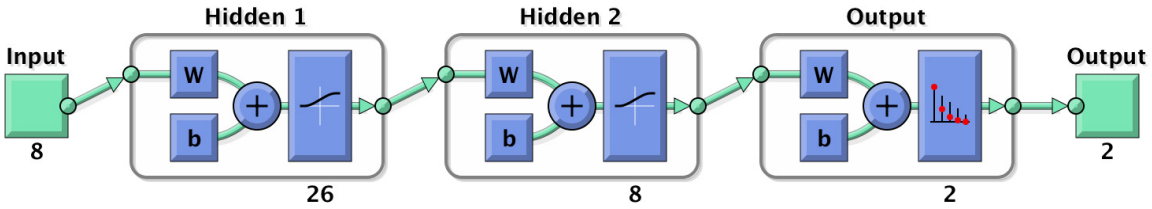


Fig. 5.18 The ANN optimal topology for Case 3.

was preliminary selected as activation function. Accuracy: 82.32% ($std = 0.1689$), specificity: 0.9320 ($std = 0.1561$), sensitivity: 0.6600 ($std = 0.3618$).

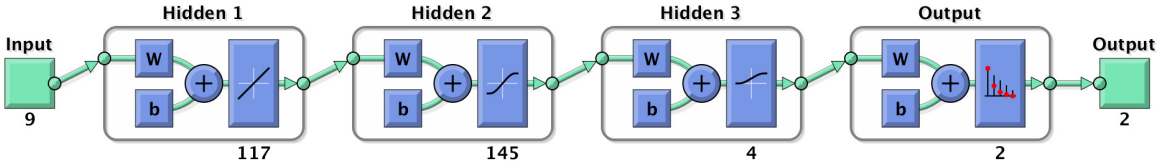


Fig. 5.19 The ANN optimal topology for Case 4.

250 permutations of train, validation and test sets were used as input of the classifiers. For this reason, all the reported results are expressed in terms of mean values and standard deviation.

The ANN training, validation, and test sets were obtained from the input dataset with 60% of samples for the training, 20% for the validation, and 20% for the test sets. Specifically, at each iteration, the above sets were obtained through a random permutation of the input dataset, keeping constant the ratio between the three classes.

The averaged normalized confusion matrices for the four cases of feature datasets are shown in Tables 5.5, 5.6, 5.7 and 5.8.

Table 5.5 Averaged normalized confusion matrix for the case 1 (dataset with all 44 feature values extracted from all three writing patterns) over 250 repetitions.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*28.95% (0.0944)*	*5.30% (0.0907)*
	Negative	*8.55% (0.0944)*	*57.20% (0.0907)*

Table 5.6 Averaged normalized confusion matrix for the case 2 (dataset with all 44 feature values extracted from two writing patterns) over 250 repetitions.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*24.88% (0.1435)*	*4.64% (0.1137)*
	Negative	*15,12% (0.1435)*	*55.36% (0.1137)*

Table 5.7 Averaged normalized confusion matrix for the case 3 (dataset with only 8 dynamic feature values obtained from PCA extracted from all three writing patterns) over 250 repetitions.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*22.60% (0.1122)*	*9.35% (0.1135)*
	Negative	*14.90% (0.1122)*	*53.15% (0.1135)*

Table 5.8 Averaged normalized confusion matrix for the case 4 (dataset with only 8 dynamic feature values obtained from PCA extracted from two writing patterns) over 250 repetitions.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*26.40% (0.1447)*	*4,08% (0.0937)*
	Negative	*13.60% (0.1447)*	*55.92% (0.0937)*

SVM results. Regarding the second approach for feature classification, a SVM classifier was trained. The best found kernel parameter configuration was with the Gaussian kernel and to ensure generalization k-fold cross validation was used with $k = 5$. The SVM approach performance in the four cases of feature dataset are (250 permutations of train, validation and test sets were used as input of the classifiers, then mean and standard deviation are reported):

Case 1. Dataset with all 44 feature values (40 dynamic, 4 static) extracted from all three writing patterns

Average performance: 87.25% ($std = 0.0467$; $max = 0.9697$; $min = 0.6667$).

Case 2. Dataset with all 44 feature values (40 dynamic, 4 static) extracted from two writing patterns (no.2 and no.3)

Average performance: 92.98% ($std = 0.0564$; $max = 1.0000$; $min = 0.7273$).

Case 3. Dataset with only 8 dynamic feature values obtained from PCA extracted from all three writing patterns

Average performance: 78.32% ($std = 0.0360$; $max = 0.8485$; $min = 0.6667$).

Case 4. Dataset with only 9 dynamic features obtained from PCA extracted from two writing patterns (no.2 and no.3)

Average performance: 89.69% ($std = 0.0359$; $max = 1.0000$; $min = 0.7727$).

Classification comparison and discussion. For the sake of clarity, the results obtained for all four dataset cases with both AI-based classification algorithms are tabulated. In detail, dataset Cases 1 and 3 are reported in Table 5.9, whereas dataset Cases 2 and 4 are reported in Table 5.10.

Table 5.9 Results comparison between AI-Algorithms applied on set A (all three writing tasks) with and without feature reduction preprocessing. Each cell reports mean (in percentage) and standard deviation (in brackets) values of accuracy.

	NO PCA	**PCA**
ANN optimal topology	*86.15% (0.1258)*	*75.75% (0.1512)*
SVM	*87.25% (0.0467)*	*78.32% (0.0360)*

Analysing the obtained results, even if the number of the subjects under test was limited, it is possible to make several important considerations regarding the computer-assisted handwriting analysis applied to the Parkinson's disease research field.

The first consideration is about the repeatability of the AI-based classification algorithms: SVM presents a limited standard deviation of accuracy with respect to one presented by the ANN optimal topology for all four dataset cases. Regarding dataset Case 1, the performance of both classification algorithms is quite similar.

Table 5.10 Results comparison between AI-Algorithms applied on set B (writing tasks no.2 and no. 3) with and without feature reduction preprocessing. Each cell reports mean (in percentage) and standard deviation (in brackets) values of accuracy.

	NO PCA	**PCA**
ANN optimal topology	*80.24% (0.1667)*	*82.32% (0.1689)*
SVM	*92.98% (0.0564)*	*89.69% (0.0359)*

In dataset Case 2, the performance obtained by the classification algorithms substantially differs for SVM algorithm which obtained a higher percentage of accuracy. This dataset Case including two writing constrained tasks (task no. 2 and no.3) allowed to reach the best performance on the differentiation between healthy subjects and Parkinson's disease patients, thus, suggesting constrained tasks are more suitable. For both dataset Cases 3 and 4, SVM presents higher accuracy percentage with respect to the ANN optimal topology.

To provide an answer to each of the same five research scientific questions reported in Section 5.2.1, with respect to the application of computer-assisted handwriting analysis to Parkinson's disease, the results obtained from the experimental tests conducted on both healthy subjects and PD patients allowed to infer that:

1. as reported in paragraph *PCA results*, the most representative feature value for differentiating healthy subjects from PD patients is the ZC feature extracted from the sEMG signals acquired on the forearm during the handwriting task. Considering all the proposed static and dynamic features, it is possible to affirm that the analysis of the muscular activity during handwriting tasks plays a fundamental role;
2. the best writing patterns to be used are those constrained in size and consisting of repetitive movements;
3. the best AI-based classification approach in terms of accuracy is SVM;
4. the best AI-based classification approach in terms of repeatability of the result is SVM;
5. the PCA-based feature reduction involves a limited effect on the accuracy obtained through the classification process. The only exception is dataset Case 3, in which the accuracy obtained by both classification algorithms is lower.

In conclusion, the results obtained with the proposed model-free technique for computer-assisted handwriting analysis showed that PD patients can be differentiated from healthy

subjects with a good accuracy. Obtained results allowed to infer that (i) analysis of the muscular activity during handwriting tasks should be taken into account when applied to Parkinson's disease research, (ii) the best writing patterns to be used should be constrained in size and consisting of repetitive movements, (iii) the best AI-based classification approach both in terms of accuracy and of repeatability of the results is SVM, and (iv) feature reduction can negatively affect the performance achievable in case of no application of feature reduction algorithms.

5.2.2 A model-free Technique Based on Biometric Signals for Parkinson's Disease Assessment and Grading

To further investigate the potentiality of handwriting analysis in the differentiation of healthy subjects and PD patients, the CAD prototype presented before was extended to include a different set of feature. A new set-up has been proposed allowing to improve the standardisation of the recorded biometrical signals (i.e., sEMG signals, pen tilt, etc.); it is based on two different devices: an sEMG bracelet sensor and a graphics tablet providing a co-located visual feedback.

Two new research studies were designed: the first investigated the use of the new set of features and compared the performance of the two type of classifier proposed in the previous section (ANN and SVM); the second study, is based on the use of deep neural network only, and investigated the most representative features that better highlight the handwriting differences between mild and moderate PD patients, and PD patients and healthy subjects.

5.2.2.1 Materials

The two study were carried out on two new dataset that extend the one used in the previous section; the subjects were carefully selected to narrow the age variability.

The first new dataset accounts 18 participants (13 males, 5 females, age: $73,40 \pm 9,87$ years old). In detail, the age matched control group was composed of 7 healthy subjects (3 males, 4 females, age: $73,42 \pm 10,62$ years old), whereas the PD group was composed of 11 subjects (10 males, 1 female, age: $73,38 \pm 9,89$).

The second dataset extends the previous and accounts 32 participants (21 males, 11 females, age: 71.4 ± 8.3 years old). In detail, the participants were composed of 21 PD subjects (17 males and 4 females, age: 72.1 ± 8.3) and 11 healthy ones (4 males and 7 females, age: 70.2 ± 10.2 years old); the healthy group was selected to match the age of the PD one. The PD group was subsequently divided into mild and moderate subgroups

according to the degree of the disease. The subgroups were composed of 12 mild patients (9 males and 3 females, age: 70.5 ± 10.0) and 9 moderate ones (8 males and 1 female, age: 73.8 ± 6.0).

All subjects signed an informed consent form.

System set-up. The system set-up is reported in Fig. 5.20. It includes the Myo^{TM} Gesture Control Armband used as sEMG bracelet sensor for acquiring sEMG signals from 8 different points of the forearm, and the WACOM Cintiq 13" HD[4] used as graphic tablet providing co-located visual feedback, pen tip position (planar x-y coordinates) and pressure, and the tilt of the pen with respect to the writing surface. The data recording and synchronisation has been achieved by developing an ad-hoc acquisition software based on the SDKs provided with the two devices; the software is written in C++ and implement Qt[5] interfaces. The development required the creation of two interfaces, one dedicated to the operator and one for the patients performing the test.

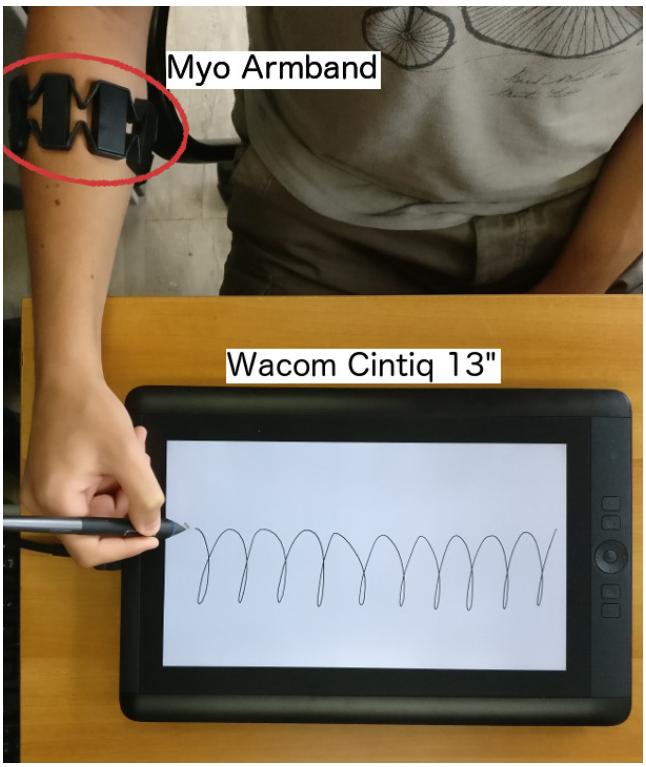


Fig. 5.20 System set-up used for the experimental tests to validate the proposed approaches.

[4] www.wacom.com/en-ch/products/pen-displays/cintiq-13-hd
[5] https://www.qt.io/

Data acquisition. To validate the proposed approach, three writing patterns corresponding to as many writing tasks were selected. They were properly differentiated according to a writing size constrained/unconstrained point of view:

- Pattern 1 – a five-turn spiral drawn in anticlockwise direction;
- Pattern 2 – a sequence of 8 Latin letter "l" with a size of 2.5 *cm*;
- Pattern 3 – a sequence of 8 Latin letter "l" with a size of 5 *cm*.

As before, it is possible to observe that only two writing patterns (Patterns 2 and 3) were size-constrained, and for those, a visual marker has been provided as a size reference.

To familiarize with the exercise, each subject was asked to perform all three writing tasks once. Then, each subject was asked to perform 3 repetitions of all 3 writing tasks, thus resulting in a total of 9 handwritten samples per subject. Each handwriting tasks was interleaved with a rest period of three seconds, and the first pressure point on the tablet has been used as a trigger for the tasks begin.

5.2.2.2 Handwriting Feature

Handwriting features have been derived from biometric signals obtained during handwriting tasks. The two devices allowed to synchronously acquire different signals representing different aspects of handwriting and to extract the following proposed features. In general, the proposed features can be grouped into two groups based on sEMG and pen tip signals. Features derived from sEMG signals are related to the subject's muscle activity and are derived from the sEMG signals obtained from the forearm of the subject; the RMS and the ZC features presented in the previous section were selected and used also in these studies (ZC value has been divided by the length of the signal to normalize the features among the subjects.). The graphics tablet allowed to acquire three biometric signals: pen tip pressure (scalar), pen tilt (2-dimensional), pen tip position (2-dimensional). The features extracted from these signals are the following (mean and standard deviation): (i) Cartesian velocity; (ii) X and (iii) Y velocity components; (iv) Cartesian acceleration; (v) X and (vi) Y acceleration components; (vii) Cartesian jerk; (viii) X and (ix) Y jerk components; (x) pen tip pressure; (xi) pen azimuth; (xii) pen altitude. In addition, specific features related to the specific type of writing pattern were also proposed: *writing size related angle* for letters and *precision related index* for spiral drawing. Next paragraphs focus on the new proposed feature.

Cartesian and XY-velocity component feature. The velocity features are referred to the kinematics of the pen tip writing on the graphic tablet. The X-Y pixel position value has been

used to compute velocity by means of first derivative. This lead to three signals: Cartesian, X- and Y-component velocity values.

Cartesian and XY-acceleration component feature. The acceleration features are referred to the kinematics of the pen tip writing on the graphic tablet. The X-Y pixel position value has been used to compute acceleration by means of second derivative. This lead to three signals: Cartesian, X- and Y-component acceleration values.

Cartesian and XY-jerk component featur.e The jerk features are referred to the kinematics of the pen tip writing on the graphic tablet. The X-Y pixel position value has been used to compute jerk by means of third derivative. This lead to three signals: Cartesian, X- and Y-component jerk values.

Pen tip pressure feature. The pressure exerted by the pen tip on the surface of the graphic tablet is acquired. The pen tip pressure feature is a discrete scalar number.

Azimuth and altitude feature. The azimuth feature is the value of the angle between a reference direction (e.g., the Y axes of the tablet) and the pen direction projected on the horizontal plane. The altitude feature is the value of the angle between the pen direction and the horizontal plane.

Writing size related features for letter-based writing patterns. Writing size related features are extracted from letter-based writing patterns. Starting from the X-Y pen tip position, the upper and the lower picks of the Y coordinate of written letters are computed. These values are, then, separately used as input data of a linear regressor leading to the extraction of two different lines: the upper regression line R_{up} and the lower regression line R_{low}. The angle α between the two lines is subsequently computed. This angle is the first proposed writing size related feature. A graphical representation of the lines extraction and of the α angle is shown in Fig. 5.21.

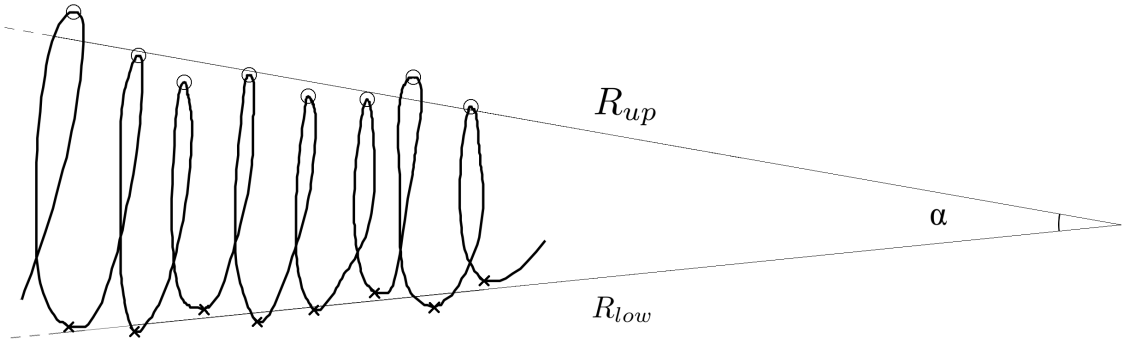


Fig. 5.21 Representation of the regression lines R_{up} and R_{low} and the angle α. Circle and cross marks identifies respectively upper and lower peaks of the Y-coordinate of the pen tip position.

The other two proposed writing size related features are represented by the coefficient of determination (R^2) of the two regression lines computed according to Equation 5.13. These two features allowed to consider the variability of the letter pattern size.

$$\begin{aligned} \bar{y} &= \frac{1}{n}\sum_{i}^{n} y_i, \\ SST &= \sum_{i}(y_i - \bar{y})^2, \\ SSE &= \sum_{i}(y_i - \widehat{y}_i)^2, \\ R^2 &= 1 - \frac{SSE}{SST} \end{aligned} \tag{5.13}$$

Spiral precision index feature. For the spiral drawing pattern, the proposed feature is related to the variability of strokes. For each point P of the X-Y pen tip position, the vector $\vec{r}$ with respect to the spiral centroid point C having origin in P is computed. The angle β between $\vec{r}$ and the direction vector $\vec{d}$ tangent to the spiral in P is then calculated (a graphical representation of $\vec{r}$, $\vec{d}$, C and β is reported in Fig. 5.22). The spiral precision index feature is the standard deviation of the β angles computed for each point P.

To summarise, each subject generated 41 features for writing task 1 and to 43 features for writing task 2 and 3:

- Root Mean Square (RMS) of each sEMG signal (8 RMS features for each subject and for each task);
- Zero Crossing (ZC) of each sEMG signal (8 RMS features for each subject and for each task);

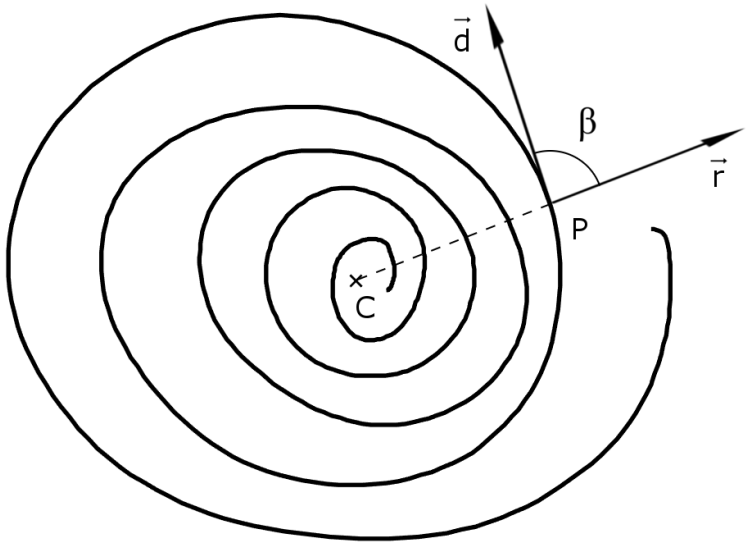


Fig. 5.22 Example of computation of the spiral precision index β.

- mean and standard deviation for each subject and for each task of the following signals:
  - pen tip Cartesian velocity (two features);
  - X and Y velocity components of the pen tip (four features);
  - pen tip Cartesian acceleration (two features);
  - X and Y acceleration components of the pen tip (four features);
  - pen tip Cartesian jerk (two features);
  - X and Y jerk components of the pen tip (four features);
  - pen tip pressure (two features);
  - pen azimuth (two features);
  - pen altitude (two features);
- pattern specific features:
  - index of precision of the spiral drawing (one feature);
  - size related features for the constrained patterns (three features).

5.2.2.3 Comparison Between ANN and SVM Classifiers

The study designed on the first dataset extends the analysis conducted in Section 5.2.1; ANN and SVM classifiers were implemented and tested.

Feature reduction and classification. The same procedure presented in the preliminary application has been replicated. Principal component analysis was investigated to understand which are the most representative features among the extracted ones; deep neural networks optimised by means a Multi-Objective Genetic Algorithm and SVM were used as classifier.

The genetic algorithm was set-up and configured as before; the performance of the classifiers were evaluated in terms of accuracy (Eq. 2.13), specificity (Eq. 2.16) and sensitivity (Eq. 2.15).

Experimental data processing description. For addressing the five specific research scientific questions reported in the previous section, the experiments have been conducted according to the following scheme:

- two different feature value sets were created: the first (set A) including the feature values extracted from the writing patterns no.1 (41 feature values), the second (set B) including the feature values extracted from writing patterns no.2 and no.3 (43 feature values $\times$ 2 writing patterns);
- PCA was applied on both set A and B to reduce the number of feature values obtaining a new feature space;
- four different feature dataset cases were, then, created:
  Case 1. Dataset with all feature values included in set A;
  Case 2. Dataset with all feature values included in set B;
  Case 3. Dataset with only PCA-obtained feature values from set A;
  Case 4. Dataset with only PCA-obtained feature values from set B.
- both AI-based classification approaches were applied on all four dataset cases;
- accuracy, specificity and sensitivity (percentage and standard deviation) obtained with ANN optimal topology approach were evaluated for each of the four dataset cases;
- average performance (percentage, standard deviation, max value, min value) obtained with the SVM approach were evaluated for each of the four dataset cases.

For sake of clarity, the scheme of the conducted experiments is shown in Fig. 5.23.

Results Two samples of one repetition of the writing tasks respectively performed by a healthy subject and a PD subject are reported (not on real scale) in Fig. 5.24 (task no.1), Fig. 5.25 (task no.2) and Fig. 5.26 (task no.3). The results obtained with the PCA and with the AI-based classification algorithms are following reported.

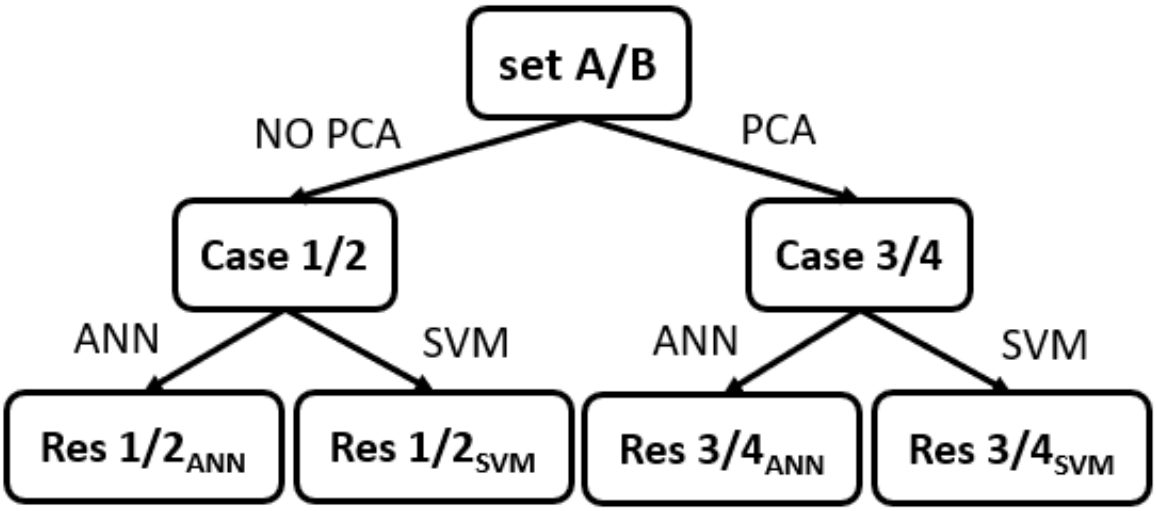


Fig. 5.23 Scheme of the experiment conducted to validate the proposed technique. Features values are grouped in set A and B. By applying or not applying the PCA, four cases are obtained. For each case, both ANN optimal topology and SVM classifier techniques are applied.

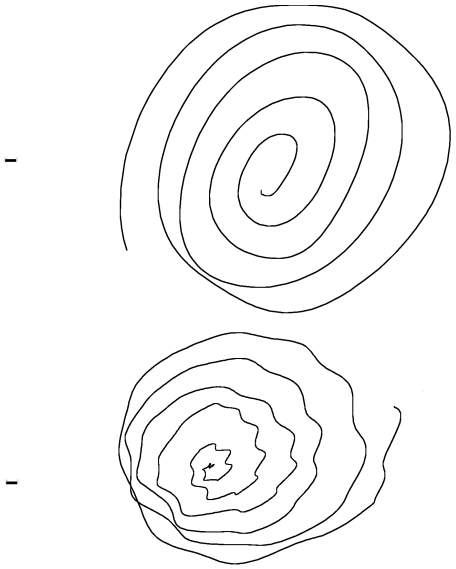


Fig. 5.24 Two samples of one repetition of the writing task no. 1 (spiral drawing) respectively performed by a healthy subject (top) and a PD subject (bottom).

PCA results. The feature reduction based on PCA has been set up to hold 99.9% of the original information. The new space generated by the PCA is composed by 23 features for set A and 25 for set B for each writing pattern. To better understand which of the features from the original space are more representative, the new generated space has been analysed by means of *biplot* graphs.

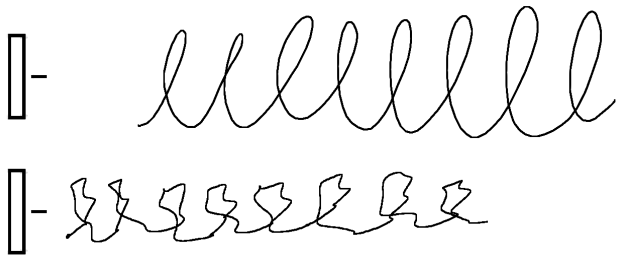


Fig. 5.25 Two samples of one repetition of the writing task no. 2 (2.5 *cm* sized 8 *"l"* sequence) respectively performed by a healthy subject (top) and a PD subject (bottom).

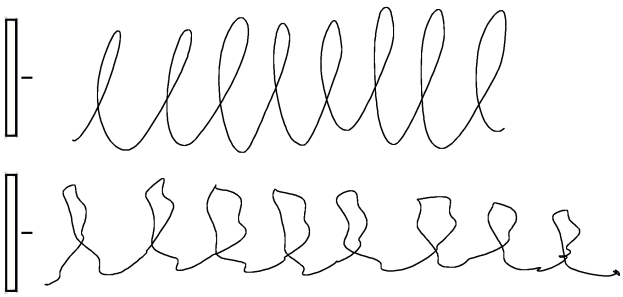


Fig. 5.26 Two samples of one repetition of the writing task no. 3 (5 *cm* sized 8 *"l"* sequence) respectively performed by a healthy subject (top) and a PD subject (bottom).

In detail, the space generated starting from set A present as most representative features the RMS values extracted from each of the eight sEMG channels. Other representative features are the standard deviation of all the velocity signals (both Cartesian and X-Y components), the ZC values of each of the eight sEMG channels and the spiral precision index feature.

The space generated starting from set B, instead, present as most representative features all three writing size related features for letter-based writing patterns (α angle and both coefficients of determination) and the RMS values extracted from each of the eight sEMG channels.

The output feature values have been used for creating the feature dataset of Case 3 and 4.

Results of the ANN optimal topology approach. Regarding the first approach for feature classification, the ANN optimal topologies specified by the genetic algorithm in the four cases of feature dataset are:

Case 1. Dataset with all 41 feature values extracted on writing patterns no.1

ANN with: 2 hidden layers, with 116 and 13 neurons for the hidden layer, and 2 neurons for the output layer. The activation function found by the GA was *logsig*

for the hidden layers. For the output layer, the *softmax* function was preliminary selected as activation function. Accuracy: 94.03% ($std = 0.0649$), specificity: 0.9752 ($std = 0.0870$), sensitivity: 0.9185 ($std = 0.0979$).

Case 2. Dataset with all 43 feature values extracted on two writing patterns (no.2 and no.3)

ANN with: 1 hidden layers, with 20 neurons for the hidden layer, and 2 neurons for the output layer. The activation function found by the GA was *logsig* for the hidden layers. For the output layer, the *softmax* function was preliminary selected as activation function. Accuracy: 97.84% ($std = 0.0305$), specificity: 0.9689 ($std = 0.0529$), sensitivity: 0.9846 ($std = 0.0357$).

Case 3. Dataset with only 23 dynamic feature values obtained from PCA extracted on writing patterns no. 1

ANN with: 3 hidden layers, with 55, 100 and 2 neurons for the hidden layer, and 2 neurons for the output layer. The activation function found by the GA was *logsig* for the hidden layers. For the output layer, the *softmax* function was preliminary selected as activation function. Accuracy: 93.32% ($std = 0.0648$), specificity: 0.9248 ($std = 0.1180$), sensitivity: 0.9385 ($std = 0.0817$).

Case 4. Dataset with only 25 dynamic features obtained from PCA extracted on two writing patterns (no.2 and no.3)

ANN with: 4 layers, with 32, 5 and 32 neurons for the hidden layer, and 2 neurons for the output layer. The activation functions found by the GA were *logsig, logsig and logsig* for the hidden layer. For the output layer, the *softmax* function was preliminary selected as activation function. Accuracy: 97.46% ($std = 0.0347$), specificity: 0.9631 ($std = 0.0643$), sensitivity: 0.9820 ($std = 0.0360$).

All the reported results are expressed in terms of mean values, considering 250 iterations in terms of permutations of train, validation and test sets.

The ANN training, validation, and test sets were obtained from the input dataset with 60% of samples for the training, 20% for the validation, and 20% for the test. Specifically, at each iteration, the above sets were obtained through a random permutation of the input dataset, keeping constant the ratio among classes.

The averaged normalized confusion matrix for the four cases of feature datasets are shown in Table 5.11, Table 5.12, Table 5.13 and 5.14.

Table 5.11 Averaged normalized confusion matrix for case 1 (over 250 iterations). Standard deviation in brackets.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*37.51% (0.0335)*	*0.95% (0.0335)*
	Negative	*5.02% (0.0602)*	*56.52% (0.0602)*

Table 5.12 Averaged normalized confusion matrix for case 2 (over 250 iterations). Standard deviation in brackets.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*37.91% (0.0207)*	*1.22% (0.0207)*
	Negative	*0.94% (0.0218)*	*59.93% (0.0218)*

Table 5.13 Averaged normalized confusion matrix for case 3 (over 250 iterations). Standard deviation in brackets.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*35.57% (0.0454)*	*2.89% (0.0454)*
	Negative	*3.78% (0.0503)*	*57.75% (0.0503)*

SVM results. Regarding the second approach for feature classification, a SVM classifier has been trained. The best found kernel parameter configuration was the one with the Gaussian kernel. To ensure generalization k-fold cross validation was configured with $k = 5$. The SVM approach performance in the four cases of feature dataset are:

Case 1. Dataset with all 41 feature values extracted on writing patterns no. 1

Average performance: 87.91% ($std = 0.0332$; $max = 0.9444$; $min = 0.7500$).

Table 5.14 Averaged normalized confusion matrix for case 4 (over 250 iterations). Standard deviation in brackets.

		True Condition	
		Positive	**Negative**
Predicted Condition	**Positive**	*37.69% (0.0252)*	*1.44% (0.0252)*
	Negative	*1.10% (0.0219)*	*59.77% (0.0219)*

Case 2. Dataset with all 43 feature values extracted on two writing patterns (no. 2 and .3)

Average performance: 94.53% ($std = 0.0197$; $max = 0.9907$; $min = 0.8796$).

Case 3. Dataset with only 23 feature values obtained from PCA extracted on writing patterns no. 1

Average performance: 87.76% ($std = 0.0303$; $max = 0.9444$; $min = 0.7592$).

Case 4. Dataset with only 25 features obtained from PCA extracted on two writing patterns (no. 2 and 3)

Average performance: 94.19% ($std = 0.0150$; $max = 0.9815$; $min = 0.8889$).

All the reported results are expressed in terms of mean values, considering 250 iterations of training and testing process.

Classification comparison and discussion. For the sake of clarity, the results obtained for all four dataset cases with both AI-based classification algorithms were tabulated: dataset cases 1 and 3 are reported in Table 5.15, whereas dataset cases 2 and 4 are reported in Table 5.16.

Analysing the obtained results, even if the number of the subjects under test was limited, it is possible to make several important considerations regarding the computer-assisted handwriting analysis applied to the Parkinson's disease research field.

The first consideration is about the repeatability of the AI-based classification algorithms: both ANN and SVM presents a limited standard deviation ($std < 0.1$) of accuracy for all four cases.

For all dataset cases, ANN optimal topology approach presents higher accuracy percentage with respect to SVM ($3.31\% < \Delta < 6.12\%$). Both classifiers obtained better performances

on feature set B (features extracted from constrained letter-based writing patterns). For both sets (A and B), PCA does not significantly affect the classification performance ($\Delta < 1\%$).

To provide an answer to the five research scientific questions reported and defined in Section 5.2.1, with respect to the application of computer-assisted handwriting analysis to Parkinson's disease, the results obtained from the experimental tests conducted on both healthy subjects and PD patients allowed to infer that:

1. as reported in paragraph *PCA results*, the most representative feature values for differentiating healthy subjects from PD patients are:
    - for writing pattern no. 1 (spiral drawing): RMS and ZC values extracted from each of the eight sEMG channels, pen tip velocities and the spiral precision index feature;
    - for writing pattern no. 2 and 3 (size constrained 8 *"l"* sequence): RMS values extracted from each of the eight sEMG channels and the three writing size related features (α angle and both coefficients of determination);
2. the best writing patterns to be used are those constrained in size and letter-based;
3. the best AI-based classification approach in terms of accuracy is ANN optimal topology;
4. both AI-based classification approach are comparable in terms of repeatability of the result;
5. the PCA-based feature reduction does not significantly affect the accuracy obtained through the classification process.

Table 5.15 Results comparison between AI-based classifier applied on set A (writing pattern no.1) both in case of original and PCA feature space. Each cell reports mean (in percentage) and standard deviation (in brackets) values of accuracy.

	NO PCA	**PCA**
ANN optimal topology	*94.03% (0.0649)*	*93.32% (0.0648)*
SVM	*87.91% (0.0332)*	*87.76% (0.0303)*

In conclusion, the results obtained with the proposed model-free technique for computer-assisted handwriting analysis showed that PD patients can be differentiated from healthy

Table 5.16 Results comparison between AI-based classifier applied on set B (writing patterns no.2 and 3) both in case of original and PCA feature space. Each cell reports mean (in percentage) and standard deviation (in brackets) values of accuracy.

	NO PCA	**PCA**
ANN optimal topology	*97.84% (0.0305)*	*97.46% (0.0347)*
SVM	*94.53% (0.0197)*	*94.11% (0.0156)*

subjects with a high accuracy (up to 97.84%). Obtained results allowed to infer that (i) analysis of the muscular activity during handwriting tasks, as well as pen tip velocities, spiral precision index feature and writing size related features should be taken into account when applied to Parkinson's disease research; (ii) the best writing patterns to be used should be constrained in size and letter-based, (iii) the best AI-based classification approach in terms of accuracy is ANN optimal topology, (iv) both ANN and SVM classifier shows a good performance in terms of repeatability, and (v) PCA feature reduction does not significantly affect the performance in terms of accuracy.

5.2.2.4 Inter and Intra Subjects Evaluation

This last section will focus on the capability of the proposed evaluation tool to perform well in the intra and inter subjects classification, that is the differentiation between mild and moderate PD patients, and PD patients and healthy subjects.

Feature selection and classification. Differently from the previous cases, a feature selection procedure has been used to understand the main representative for the subject's status [497]; the used approach is similar to the entropy criterion (i.e., information gain) and it is based on a classification decision tree technique with Gini's diversity index [498].

In the previous study case, the better classification procedure was based on the deep neural networks optimised by means of genetic algorithm. The same procedure and genetic algorithm configuration were used also in this case.

The used performance indexes were accuracy (Eq. 2.13), specificity (Eq. 2.16) and sensitivity (Eq. 2.15). Due to the dependence of the ANN performances from both net initialisation and permutation of training-validation datasets, the net training procedure iterated over 250 data permutations.

Experimental data processing description. The objectives of the conducted experiments were mainly two:

1. separate PD patients from healthy subjects;
2. correctly classify mild and moderate Parkinson patients.

For each objective, the features extracted during the experiments were grouped according to the following scheme:

- creation of three different feature datasets:
  - dataset A including only the features extracted from writing pattern 1 (41 features);
  - dataset B including only the features extracted from writing pattern 2 (43 features);
  - dataset C including only the features extracted from writing pattern 3 (43 features);
- application of the feature selection algorithm to reduce the number of the features;
- creation of six different new feature datasets:
  - Case 1. Dataset including all the feature of set A;
  - Case 2. Dataset including all the feature of set B;
  - Case 3. Dataset including all the feature of set C
  - Case 4. Dataset including only the features obtained by the application of the feature selection algorithm on dataset A;
  - Case 5. Dataset including only the features obtained by the application of the feature selection algorithm on dataset B;
  - Case 6. Dataset including only the features obtained by the application of the feature selection algorithm on dataset C.

Fig. 5.27 depicts the scheme of the experiments.

Results. The presentation and the discussion of the results obtained from the experiments have been subdivided according to the two objectives.

For each case the training procedure was iterated 250 times to assess the stability of the learning process; hence, the confusion matrices and the related results are presented in percentage with the standard deviation reported in brackets.

Objective 1 - Separating PD patients and healthy subjects:

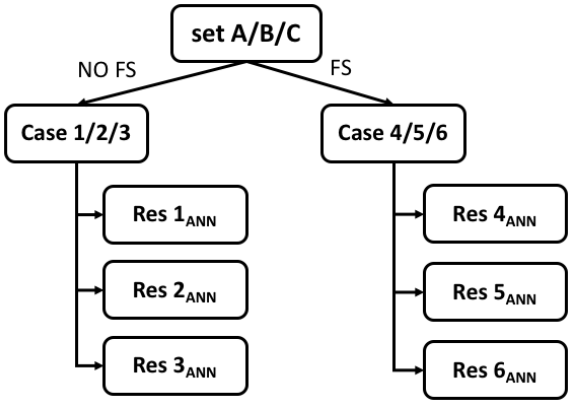


Fig. 5.27 Scheme of the experiment. Features are grouped in three sets: A, B and C. The application of the feature selection algorithm leads to six cases.

- *Feature Selection results*: The application of the feature selection algorithm previously reported, led to a significant reduction of the number of considered features for all three datasets of features extracted from the writing patterns. In particular:
  - for dataset A including the 41 features extracted from writing pattern 1, the sEMG RMS value, three sEMG ZC values, the mean Cartesian velocity and the mean acceleration on X axes were the six selected features to be classified in Case 4;
  - for dataset B including the 43 features extracted from writing pattern 2, the mean jerk on Y axes, three sEMG ZC values, the mean Cartesian acceleration and the mean velocity on X axes were the six selected features to be classified in Case 5;
  - for dataset C including the 43 features extracted from writing pattern 3, two sEMG RMS values, a sEMG ZC value, the mean cartesian velocity, the altitude STD, the azimuth RMS and the mean velocity on X axes were the seven selected features to be classified in Case 6.
- *Classification results:* for each of the six different feature datasets, the MOGA algorithm is applied to provide the optimal ANN topology. The optimal topology results and the confusion matrices are reported in Table 5.17 and in Tables 5.18 to 5.23, respectively; the performances expressed in terms of accuracy, specificity and sensitivity have been summarised in Table 5.24.

Table 5.17 Objective 1: results of the application of the MOGA algorithm on each of the six different feature datasets. The output layer configuration was preliminarily fixed with two neurons and softmax as activation function.

Case	**Number of Features**	**Writing Pattern**	**ANN Topology**		**Accuracy**
			Number of Neurons	**Activation Function**	
1	*41*	1	*186/15/2*	*logsig/logsig/softmax*	*90.76%*
2	*43*	2	*44/10/2*	*logsig/logsig/softmax*	*92.98%*
3	*43*	3	*232/82/7/2*	*logsig/logsig/logsig/softmax*	*95.95%*
4	*6*	1	*222/25/2*	*logsig/logsig/softmax*	*93.78%*
5	*6*	2	*246/12/2*	*logsig/logsig/softmax*	*91.58%*
6	*7*	3	*45/114/21/2*	*satlins/tansig/logsig/softmax*	*96.85%*

Table 5.18 Confusion matrix of Case 1 (Objective 1)

		True Condition	
		PD	***Control***
Predicted Condition	***PD***	*59.75%*	*5.35%*
	Control	*3.89%*	*31.02%*

Table 5.19 Confusion matrix of Case 2 (Objective 1)

		True Condition	
		PD	***Control***
Predicted Condition	***PD***	*60.36%*	*3.75%*
	Control	*3.89%*	*32.62%*

Table 5.20 Confusion matrix of Case 3 (Objective 1)

		True Condition	
		PD	***Control***
Predicted Condition	***PD***	*61.67%*	*2.09%*
	Control	*1.96%*	*34.27%*

Table 5.21 Confusion matrix of Case 4 (Objective 1)

		True Condition	
		PD	***Control***
Predicted Condition	***PD***	*61.40%*	*3.98%*
	Control	*2.24%*	*32.38%*

Table 5.22 Confusion matrix of Case 5 (Objective 1)

		True Condition	
		PD	***Control***
Predicted Condition	***PD***	*60.35%*	*5.13%*
	Control	*2.24%*	*31.24%*

Table 5.23 Confusion matrix of Case 6 (Objective 1)

		True Condition	
		PD	***Control***
Predicted Condition	***PD***	*62.11%*	*1.62%*
	Control	*1.53%*	*34.75%*

Table 5.24 Objective 1: performances of the application of the MOGA algorithm on each of the six different feature datasets. Results are reported as mean and standard deviation values over 250 iterations.

Case	**Accuracy**	**Specificity**	**Sensitivity**
1	*0.9076 [0.0764]*	*0.8530 [0.1553]*	*0.9389 [0.0720]*
2	*0.9298 [0.0523]*	*0.8970 [0.1212]*	*0.9486 [0.0587]*
3	*0.9595 [0.0479]*	*0.9425 [0.0831]*	*0.9691 [0.0575]*
4	*0.9378 [0.0566]*	*0.8905 [0.1356]*	*0.9649 [0.0537]*
5	*0.9158 [0.0526]*	*0.8590 [0.1153]*	*0.9483 [0.0607]*
6	*0.9685 [0.0405]*	*0.9555 [0.0805]*	*0.9760 [0.0500]*

Objective 2 - Separating mild and moderate PD patients:

- *Feature Selection results:* the application of the feature selection algorithm previously reported, led to a significant reduction of the number of considered features for all three datasets of features extracted from writing patterns. In particular:
  - for dataset A including the 41 features extracted from writing pattern 1, two sEMG RMS values, two sEMG ZC values, the mean pressure and the mean altitude were the six selected features to be classified in Case 4;
  - for dataset B including the 43 features extracted from writing pattern 2, two sEMG RMS values, two sEMG ZC values and the mean Cartesian velocity were the five selected features to be classified in Case 5;
  - for dataset C including the 43 features extracted from writing pattern 3, two sEMG RMS values, a sEMG ZC value, the mean Cartesian velocity on X axes and the mean pressure were the five selected features to be classified in Case 6.
- *Classification results:* for each of the six different feature datasets, the MOGA algorithm is applied to provide the optimal ANN topology. The optimal topology results and the confusion matrices are reported in Table 5.25 and in Tables 5.26 to 5.31, respectively; the performances expressed in terms of accuracy, specificity and sensitivity have been summarised in Table 5.32.

Table 5.25 Objective 2: results of the application of the MOGA algorithm on each of the six different feature datasets. The output layer configuration was preliminarily fixed with two neurons and softmax as activation function.

Case	**Number of Features**	**Writing Pattern**	**ANN Topology**		**Accuracy**
			Number of Neurons	**Activation Function**	
1	*41*	1	*59/65/2/2*	*logsig/logsig/logsig/softmax*	*94.34%*
2	*43*	2	*138/18/1/2*	*logsig/logsig/logsig/softmax*	*87.26%*
3	*43*	3	*65/36/7/2*	*logsig/logsig/logsig/softmax*	*91.86%*
4	*6*	1	*123/2*	*logsig/softmax*	*96.00%*
5	*5*	2	*67/24/2*	*logsig/logsig/softmax*	*86.71%*
6	*5*	3	*17/2*	*tansig/softmax*	*91.66%*

Table 5.26 Confusion matrix of Case 1 (Objective 2)

		True Condition	
		Moderate	***Mild***
Predicted Condition	***Moderate***	*39.51%*	*2.31%*
	Mild	*3.34%*	*54.83%*

Table 5.27 Confusion matrix of Case 2 (Objective 2)

		True Condition	
		Moderate	***Mild***
Predicted Condition	***Moderate***	*37.43%*	*7.31%*
	Mild	*5.43%*	*49.83%*

Table 5.28 Confusion matrix of Case 3 (Objective 2)

		True Condition	
		Moderate	***Mild***
Predicted Condition	***Moderate***	*39.51%*	*4.80%*
	Mild	*3.34%*	*52.34%*

Table 5.29 Confusion matrix of Case 4 (Objective 2)

		True Condition	
		Moderate	***Mild***
Predicted Condition	***Moderate***	*41.31%*	*2.46%*
	Mild	*1.54%*	*54.69%*

Table 5.30 Confusion matrix of Case 5 (Objective 2)

		True Condition	
		Moderate	***Mild***
Predicted Condition	***Moderate***	*36.51%*	*6.94%*
	Mild	*6.34%*	*50.20%*

Table 5.31 Confusion matrix of Case 6 (Objective 2)

		True Condition	
		Moderate	***Mild***
Predicted Condition	***Moderate***	*39.29%*	*4.77%*
	Mild	*3.57%*	*52.37%*

Table 5.32 Objective 2: performances of the application of the MOGA algorithm on each of the six different feature datasets. Results are reported as mean and standard deviation values over 250 iterations.

Case	**Accuracy**	**Specificity**	**Sensitivity**
1	*0.9434 [0.0626]*	*0.9595 [0.0763]*	*0.9220 [0.1158]*
2	*0.8726 [0.0850]*	*0.8720 [0.1206]*	*0.8733 [0.1544]*
3	*0.9186 [0.0830]*	*0.9196 [0.1167]*	*0.9220 [0.1286]*
4	*0.9600 [0.0658]*	*0.9570 [0.0939]*	*0.9640 [0.0947]*
5	*0.8671 [0.0837]*	*0.8785 [0.1128]*	*0.8520 [0.1598]*
6	*0.9166 [0.0858]*	*0.9165 [0.1163]*	*0.9167 [0.1313]*

For the sake of clarity, the accuracy of all cases for both objectives are summarised in Table 5.33. As reported in the table, the proposed procedure leads to high accuracy performances; the results for both objectives present accuracy in the range $86 < x < 97$, with a standard deviation lower than 0.09. The low value of the standard deviation allows to assess the stability of the learning process of the optimal ANN. Similar observations can be stated for both objectives for the classification of the selected features. In detail:

- the classification between PD patients and healthy subjects (objective 1) achieves the best accuracy (96.85%) in *Case 6* (seven features selected from the dataset of 43 features extracted from writing pattern 3). The feature selection stated that three out of seven features were related to sEMG signals (RMS and ZC), whereas the others to pen tilt and velocity;
- the classification between mild and moderate PD patients (objective 2) achieves the best accuracy (96.00%) in *Case 4* (six features selected from the dataset of 41 features extracted from writing pattern 1). The feature selection stated that four out of six features were related to sEMG signals (RMS and ZC), whereas the others to pen tilt and velocity;

Table 5.33 Summary of the accuracy values obtained for each of the two objectives for each considered case. Standard deviation over 250 repetitions is reported in brackets.

	Case	Objective	
		1	**2**
All Feature	1	*90.76% (0.0764)*	*94.34% (0.0626)*
	2	*92.98% (0.0523)*	*87.26% (0.0850)*
	3	*95.95% (0.0479)*	*91.86% (0.0830)*
Selected Feature	4	*93.78% (0.0566)*	*96.00% (0.0658)*
	5	*91.58% (0.0526)*	*86.71% (0.0837)*
	6	*96.85% (0.0405)*	*91.66% (0.0858)*

In conclusion it is possible to state that the proposed DSSs are able to classify healthy subjects vs PD patients and mild vs moderate PD patients with a high classification accuracy (more than 90.0%). Furthermore, a limited set of representative feature selected by means of a classification decision tree technique, that uses the Gini's diversity index, improved the overall accuracy (more than 96.0%). Further works are needed to investigate the DSS performance with a larger cohort of subjects that includes severe PD patients too. This will allow to classify PD patients by using more than two PD status classes and to monitor the progress of the disease over time. Furthermore, due to the time-consuming acquisition steps, it is desirable to reduce the required number of pattern tasks; this can be achieved through a proper writing pattern selection among the proposed ones.

Chapter 6

Conclusion

The studies conducted in this thesis aimed to design, develop and evaluate innovative systems to support clinician in their daily practice, focusing on the importance of a quantitative and objective evaluation.

In all the proposed solutions, three CAD requirements were tried to be pursuit: improve clinicians performance, reduce or at least not increase clinicians time and integrate the CAD solution in standard procedures. Most of the proposed CAD solutions satisfy these requirements and the Chronic Kidney Disease study cases fulfils all three. A more speculative research has been conducted for signal processing. Undercomplete autoencoders for surface electromyography analysis have been designed and validated for the evaluation of complex muscle activation patterns. Finally, machine learning has been investigated as signal processing technique for diseases assessment and grading in subjects affected by movement disorders. The developed solutions for signals and images processing, were compared with literature standards and, if possible, a personalised classical pipelines has been proposed and customised to face each clinical challenge.

The introduction on traditional CAD systems, their evolution throw the deep learning paradigm and the description of the study cases have been presented in Chapter 2. A deep literature research about the deep learning research field and its application to the clinical domains of interest have been conducted too. Chapters 3, 4 and 5 focus on the proposed and developed solutions, reporting all the design decision taken to face the respective challenges; the obtained results were compared with standard approaches and literature highlighting all the advantages and drawbacks.

The evaluation of the suitability of kidney from expanded criteria donors, relies on the histological examination of kidney biopsies performed at the time of organ retrieval by pathologists. Several CAD pipelines were proposed facing one step of the biopsy evaluation

from the classification, detection and segmentation perspectives. All the results were validated by renal pathologists which confirmed the reliability of the proposed work-flow; this allows to consider the applied methodology as a milestone in the creation of a CAD system for the renal transplant evaluation, easing pathologists in accomplishing the laborious task of transplantation evaluation and providing rapid and accurate results. Furthermore, the CAD pipeline is fully integrable in the used clinical tools. In the future it could be extended to other kidney biopsies analysis tasks, allowing to define a complete histological evaluation.

A customised and fully-automatic CNNs-based approach has been proposed for the segmentation of liver parenchyma and liver vessels in CT scans. Appropriate loss functions have been investigated to better tune the learning procedure, and different metrics were used for the evaluation; some metrics and post-processing steps were carefully selected to respectively evaluate and improve the model from a surgical planning point of view. The obtained results show that the proposed work-flow is a very promising approach for the vessels segmentation in CT scans, allowing to obtain accurate volumetric reconstructions of the segmented regions. Results allow to state that the system can help radiologists in accomplishing the laborious task of segmenting liver and vessels from CT scan, laying the foundation for further image analysis algorithm on the segmented regions. Future works can include further validation on datasets coming from different cohorts of subjects, and investigation on novel analysis of the segmented regions, targeted to obtain the Couinaud hepatic segments classification.

The proposed strategy for synergy evaluation and the relative comparison with the standard techniques shown promising results and might open new perspectives for muscle synergy extraction techniques and, perhaps, encourages new studies related to the fundamentals of muscle synergies and human motor learning and control. In fact, the findings of such research might moreover be useful for implementing more intuitive simultaneous and proportional myo-electric controls of prostheses and robotic devices, and for the development of innovative diagnostic tools and rehabilitation approaches. In the future, the study of task-oriented synergies and the relative comparison, could reveal interesting information about whether and how those patterns might be used to improve the myo-controllers and rehabilitative therapies.

Finally, the development of CAD solution based on deep neural networks for movement disorders assessment and grading shown promising results. A new software tool for blepharospasm evaluation has been developed; the solution, starting from video registered on patients, is capable to recognise the main symptoms of this focal dystonia by detecting blinks, brief and prolonged spasms. The tool results were analysed, compared with the

human counterpart and correlated with a standard scale. Future works will deal with the time required for the tool usage and with the inclusion of all the symptoms analysed by the standard scale. Handwriting analysis has been investigated as a methodology for Parkinson' disease assessment and grading. Two main study, and as many CAD tool, have been presented investigating how the joining of dynamic features extracted from sEMG and handwritten text/drawing features, can be used for patients evaluation. A first study make use of features extracted from scanned paper sheets analysed by exploiting vision-based features, while a second one investigated advantages and drawbacks of features generable by using tablets as input device. The proposed DSSs demonstrated to be able to distinguish healthy subjects from PD patients and mild from moderate PD patients with high performance. Future works will analyse a wider cohort including also severe PD subjects; this will allow to classify PD patients and to monitor their progresses of over time.

My Publications

[1] Giacomo Donato Cascarano, Francesco Saverio Debitonto, Ruggero Lemma, Antonio Brunetti, Domenico Buongiorno, Irio De Feudis, Andrea Guerriero, Umberto Venere, Silvia Matino, Maria Teresa Rocchetti, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio. Bevilacqua. A neural network for glomerulus classification based on histological images of kidney biopsy. BMC Supplements (Under Review).

[2] Nicola Altini, Giacomo Donato Cascarano, Antonio Brunetti, Irio De Feudis, Domenico Buongiorno, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio Bevilacqua. A deep learning instance segmentation approach for global glomerulosclerosis assessment in donor kidney biopsies. *Electronics*, 9(11):1768, Oct 2020. ISSN 2079-9292. doi: 10.3390/electronics9111768. URL http://dx.doi.org/10.3390/electronics9111768.

[3] Nicola Altini, Giacomo Donato Cascarano, Antonio Brunetti, Francescomaria Marino, Maria Teresa Rocchetti, Silvia Matino, Umberto Venere, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio Bevilacqua. Semantic Segmentation Framework for Glomeruli Detection and Classification in Kidney Histological Sections. *Electronics*, 9(3):503, mar 2020. ISSN 2079-9292. doi: 10.3390/electronics9030503. URL https://www.mdpi.com/2079-9292/9/3/503.

[4] Vitoantonio Bevilacqua, Antonio Brunetti, Giacomo Donato Cascarano, Andrea Guerriero, Francesco Pesce, Marco Moschetta, and Loreto Gesualdo. A comparison between two semantic deep learning frameworks for the autosomal dominant polycystic kidney disease segmentation based on magnetic resonance images. *BMC Medical Informatics and Decision Making*, 19(S9):244, dec 2019. ISSN 1472-6947. doi: 10.1186/s12911-019-0988-4. URL https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0988-4.

[5] Domenico Buongiorno, Giacomo Donato Cascarano, Irio De Feudis, Antonio Brunetti, Leonarda Carnimeo, Giovanni Dimauro, and Vitoantonio Bevilacqua. Deep learning for processing electromyographic signals: A taxonomy-based survey. *Neurocomputing*, 2020. ISSN 0925-2312. doi: 10.1016/j.neucom.2020.06.139. URL http://www.sciencedirect.com/science/article/pii/S0925231220319020.

[6] Domenico Buongiorno, Giacomo Donato Cascarano, Cristian Camardella, Irio De Feudis, Antonio Frisoli, and Vitoantonio Bevilacqua. Task-Oriented Muscle Synergy Extraction

Using An Autoencoder-Based Neural Model. *Information*, 11(4):219, apr 2020. ISSN 2078-2489. doi: 10.3390/info11040219. URL https://www.mdpi.com/2078-2489/11/4/219.

[7] Giacomo Donato Cascarano, Claudio Loconsole, Antonio Brunetti, Antonio Lattarulo, Domenico Buongiorno, Giacomo Losavio, Eugenio Di Sciascio, and Vitoantonio Bevilacqua. Biometric handwriting analysis to support Parkinson's Disease assessment and grading. *BMC Medical Informatics and Decision Making*, 19(S9): 252, dec 2019. ISSN 1472-6947. doi: 10.1186/s12911-019-0989-3. URL https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0989-3.

[8] Antonio Brunetti, Giacomo Donato Cascarano, and Vitoantonio Bevilacqua. Deep Learning for Medical Imaging in the Era of Precision Medicine. In Riccardo Bellazzi, Cecilia Laschi, Silvana Quaglini, and Lucia Sacchi, editors, *AI-enabled health care: from decision support to autonomous robots*, pages 75 – 99. PÀTRON EDITORE, BOLOGNA, 2020.

[9] Giacomo Donato Cascarano, Francesco Saverio Debitonto, Ruggero Lemma, Antonio Brunetti, Domenico Buongiorno, Irio De Feudis, Andrea Guerriero, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio Bevilacqua. An Innovative Neural Network Framework for Glomerulus Classification Based on Morphological and Texture Features Evaluated in Histological Images of Kidney Biopsy. In De-Shuang Huang Zhi-Kai Huang Abir Hussain, editor, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11645 LNAI, pages 727–738. Springer, Cham, CH -, aug 2019. ISBN 9783030267650. doi: 10.1007/978-3-030-26766-7_66. URL http://link.springer.com/10.1007/978-3-030-26766-7_66.

[10] Paola Suavo-Bulzis, Federica Albanese, Davide Mallardi, Francesco Saverio Debitonto, Ruggero Lemma, Annalisa Granatiero, Marisa Spadavecchia, Giacomo Donato Cascarano, Vitoantonio Bevilacqua, Loreto Gesualdo, and Francesco Pesce. P0119 ARTIFICIAL INTELLIGENCE IN RENAL PATHOLOGY: IBM WATSON FOR THE IDENTIFICATION OF GLOMERULOSCLEROSIS. *Nephrology Dialysis Transplantation*, 35(Supplement_3):418, jun 2020. ISSN 0931-0509. doi: 10.1093/ndt/gfaa142.P0119. URL https://academic.oup.com/ndt/article/doi/10.1093/ndt/gfaa142.P0119/5852854.

[11] Nicola Altini, Berardino Prencipe, Antonio Brunetti, Gioacchino Brunetti, Vito Triggiani, Leonarda Carnimeo, Francescomaria Marino, Andrea Guerriero, Laura Villani, Arnaldo Scardapane, and Giacomo Donato Cascarano. A tversky loss-based convolutional neural network for liver vessels segmentation. In De-Shuang Huang, Vitoantonio Bevilacqua, and Abir Hussain, editors, *Intelligent Computing Theories and Application*, pages 342–354, Cham, 2020. Springer International Publishing. ISBN 978-3-030-60799-9. doi: 10.1007/978-3-030-60799-9_30.

[12] Berardino Prencipe, Nicola Altini, Giacomo Donato Cascarano, Andrea Guerriero, and Antonio Brunetti. A novel approach based on region growing algorithm for liver

and spleen segmentation from ct scans. In De-Shuang Huang, Vitoantonio Bevilacqua, and Abir Hussain, editors, *Intelligent Computing Theories and Application*, pages 398–410, Cham, 2020. Springer International Publishing. ISBN 978-3-030-60799-9. doi: 10.1007/978-3-030-60799-9_35.

[13] Antonio Brunetti, Giacomo Donato Cascarano, Irio De Feudis, Marco Moschetta, Loreto Gesualdo, and Vitoantonio Bevilacqua. Detection and Segmentation of Kidneys from Magnetic Resonance Images in Patients with Autosomal Dominant Polycystic Kidney Disease. In De-Shuang Huang Kang-Hyun Jo Zhi-Kai Huang, editor, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11644 LNCS, pages 639–650. Springer, Cham, CH, 2019. ISBN 9783030269685. doi: 10.1007/978-3-030-26969-2_60. URL http://link.springer.com/10.1007/978-3-030-26969-2_60.

[14] Vitoantonio Bevilacqua, Antonio Brunetti, Giacomo Donato Cascarano, Flavio Palmieri, Andrea Guerriero, and Marco Moschetta. A Deep Learning Approach for the Automatic Detection and Segmentation in Autosomal Dominant Polycystic Kidney Disease Based on Magnetic Resonance Images. In De-Shuang Huang;Kang-Hyun Jo;Xiao-Long Zhang, editor, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10955 LNCS, pages 643–649. Springer, Cham, CH -, 2018. ISBN 9783319959320. doi: 10.1007/978-3-319-95933-7_73. URL http://link.springer.com/10.1007/978-3-319-95933-7_73.

[15] Domenico Buongiorno, Giacomo Donato Cascarano, Antonio Brunetti, Irio De Feudis, and Vitoantonio Bevilacqua. A Survey on Deep Learning in Electromyographic Signal Analysis. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11645 LNAI, pages 751–761. Springer, Cham, CH, 2019. ISBN 9783030267650. doi: 10.1007/978-3-030-26766-7_68. URL http://link.springer.com/10.1007/978-3-030-26766-7_68.

[16] Domenico Buongiorno, Cristian Camardella, Giacomo Donato Cascarano, Luis Pelaez Murciego, Michele Barsotti, Irio De Feudis, Antonio Frisoli, and Vitoantonio Bevilacqua. An undercomplete autoencoder to extract muscle synergies for motor intention detection. In *2019 International Joint Conference on Neural Networks (IJCNN)*, volume 2019-July, pages 1–8, Piscataway, NJ -, jul 2019. IEEE. ISBN 978-1-7281-1985-4. doi: 10.1109/IJCNN.2019.8851975. URL https://ieeexplore.ieee.org/document/8851975/.

[17] Irio De Feudis, Domenico Buongiorno, Giacomo Donato Cascarano, Antonio Brunetti, Donato Micele, and Vitoantonio Bevilacqua. A Nonlinear Autoencoder for Kinematic Synergy Extraction from Movement Data Acquired with HTC Vive Trackers. In Anna Esposito, Marcos Faundez-Zanuy, Francesco Carlo Morabito, and Eros Pasero, editors, *Smart Innovation, Systems and Technologies*, volume 184, pages 231–241. Springer, Singapore, Singapore, 2020. ISBN 978-981-15-5093-5. doi: 10.1007/978-981-15-5093-5_22. URL http://link.springer.com/10.1007/978-981-15-5093-5_22.

[18] Giacomo Donato Cascarano, Antonio Brunetti, Domenico Buongiorno, Gianpaolo Francesco Trotta, Claudio Loconsole, Ilaria Bortone, and Vitoantonio Bevilacqua. A Multi-modal Tool Suite for Parkinson's Disease Evaluation and Grading. In Anna Esposito;Marcos Faundez-Zanuy;Francesco Carlo Morabito;Eros Pasero, editor, *Smart Innovation, Systems and Technologies*, volume 151, pages 257–268. Springer, Singapore -, 2020. ISBN 978-981-13-8949-8. doi: 10.1007/978-981-13-8950-4_24. URL http://link.springer.com/10.1007/978-981-13-8950-4_24.

[19] Domenico Buongiorno, Ilaria Bortone, Giacomo Donato Cascarano, Gianpaolo Francesco Trotta, Antonio Brunetti, and Vitoantonio Bevilacqua. A low-cost vision system based on the analysis of motor features for recognition and severity rating of Parkinson's Disease. *BMC Medical Informatics and Decision Making*, 19 (S9):243, dec 2019. ISSN 1472-6947. doi: 10.1186/s12911-019-0987-5. URL https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0987-5.

[20] Gianpaolo F. Trotta, Roberta Pellicciari, Antonio Boccaccio, Antonio Brunetti, Giacomo D. Cascarano, Vito M. Manghisi, Michele Fiorentino, Antonio E. Uva, Giovanni Defazio, and Vitoantonio Bevilacqua. A neural network-based software to recognise blepharospasm symptoms and to measure eye closure time. *Computers in Biology and Medicine*, 112:103376, sep 2019. ISSN 00104825. doi: 10.1016/j.compbiomed.2019.103376. URL https://linkinghub.elsevier.com/retrieve/pii/S0010482519302537.

[21] Claudio Loconsole, Giacomo Donato Cascarano, Antonio Brunetti, Gianpaolo Francesco Trotta, Giacomo Losavio, Vitoantonio Bevilacqua, and Eugenio Di Sciascio. A model-free technique based on computer vision and sEMG for classification in Parkinson's disease by using computer-assisted handwriting analysis. *Pattern Recognition Letters*, 121:28–36, apr 2019. ISSN 01678655. doi: 10.1016/j.patrec.2018.04.006. URL https://www.sciencedirect.com/science/article/pii/S0167865518301260.

[22] Ilaria Bortone, Gianpaolo Francesco Trotta, Giacomo Donato Cascarano, Alberto Argentiero, Nadia Agnello, Giuseppe Nicolardi, and Vitoantonio Bevilacqua. Optimal Classifier of Parkinson's Disease based on features selected by Information Gain in 3D Gait Analysis for Differential Diagnosis. *Gait & Posture*, 57:205–206, sep 2017. ISSN 09666362. doi: 10.1016/j.gaitpost.2017.06.372. URL https://linkinghub.elsevier.com/retrieve/pii/S0966636217305957.

[23] Leonarda Carnimeo, Gianpaolo Francesco Trotta, Antonio Brunetti, Giacomo Donato Cascarano, Domenico Buongiorno, Claudio Loconsole, Eugenio Di Sciascio, and Vitoantonio Bevilacqua. Proposal of a health care network based on big data analytics for PDs. *The Journal of Engineering*, 2019(6):4603–4611, jun 2019. ISSN 2051-3305. doi: 10.1049/joe.2018.5142. URL https://digital-library.theiet.org/content/journals/10.1049/joe.2018.5142.

[24] Ilaria Bortone, Domenico Buongiorno, Giuseppina Lelli, Andrea Di Candia, Giacomo Donato Cascarano, Gianpaolo Francesco Trotta, Pietro Fiore, and Vitoantonio

Bevilacqua. Gait Analysis and Parkinson's Disease: Recent Trends on Main Applications in Healthcare. In Lorenzo Masia;Silvestro Micera;Metin Akay;José L. Pons, editor, *Biosystems and Biorobotics*, volume 21, pages 1121–1125. Springer, Cham, CH -, 2019. ISBN 978-3-030-01844-3. doi: 10.1007/978-3-030-01845-0_224. URL http://link.springer.com/10.1007/978-3-030-01845-0_224.

[25] Giacomo Losavio, Bernadette Tamma, Angelo Abbattista, Ilaria Sabina Tatò, Domenico Buongiorno, Giacomo Donato Cascarano, Antonio Brunetti, Irio De Feudis, and Vitoantonio Bevilacqua. On the Analysis of the Relationship Between Alkaline Water Usage and Muscle Fatigue Recovery. *Advances in Intelligent Systems and Computing*, 1215 AISC:26–31, 2020. ISSN 21945365. doi: 10.1007/978-3-030-51549-2_4. URL http://link.springer.com/10.1007/978-3-030-51549-2_4.

[26] Claudio Loconsole, Giacomo Donato Cascarano, Antonio Lattarulo, Antonio Brunetti, Gianpaolo Francesco Trotta, Domenico Buongiorno, Ilaria Bortone, Irio De Feudis, Giacomo Losavio, Vitoantonio Bevilacqua, and Eugenio Di Sciascio. A comparison between ANN and SVM classifiers for Parkinson's disease by using a model-free computer-assisted handwriting analysis based on biometric signals. In *2018 International Joint Conference on Neural Networks (IJCNN)*, volume 2018-July, pages 1–8, Piscataway, NJ -, jul 2018. IEEE. ISBN 978-1-5090-6014-6. doi: 10.1109/IJCNN.2018.8489293. URL https://ieeexplore.ieee.org/document/8489293/.

[27] Ilaria Bortone, Marco Giuseppe Quercia, Nicola Ieva, Giacomo Donato Cascarano, Gianpaolo Francesco Trotta, Sabina Ilaria Tatò, and Vitoantonio Bevilacqua. Recognition and Severity Rating of Parkinson's Disease from Postural and Kinematic Features During Gait Analysis with Microsoft Kinect. In De-Shuang Huang;Kang-Hyun Jo;Xiao-Long Zhang, editor, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10955 LNCS, pages 613–618. Springer, Cham, CH -, 2018. ISBN 9783319959320. doi: 10.1007/978-3-319-95933-7_70. URL http://link.springer.com/10.1007/978-3-319-95933-7_70.

[28] Ilaria Bortone, Gianpaolo Francesco Trotta, Giacomo Donato Cascarano, Paola Regina, Antonio Brunetti, Irio De Feudis, Domenico Buongiorno, Claudio Loconsole, and Vitoantonio Bevilacqua. A Supervised Approach to Classify the Status of Bone Mineral Density in Post-Menopausal Women through Static and Dynamic Baropodometry. In *2018 International Joint Conference on Neural Networks (IJCNN)*, volume 2018-July, pages 1–7, Piscataway, NJ - USA, jul 2018. IEEE. ISBN 978-1-5090-6014-6. doi: 10.1109/IJCNN.2018.8489205. URL https://ieeexplore.ieee.org/document/8489205/.

[29] Vitoantonio Bevilacqua, Claudio Loconsole, Antonio Brunetti, Giacomo Donato Cascarano, Antonio Lattarulo, Giacomo Losavio, and Eugenio Di Sciascio. A Model-Free Computer-Assisted Handwriting Analysis Exploiting Optimal Topology ANNs on Biometric Signals in Parkinson's Disease Research. In De-Shuang Huang Kang-Hyun Jo Xiao-Long Zhang, editor, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*,

volume 10955 LNCS, pages 650–655. Springer, Cham, CH - CHE, aug 2018. ISBN 9783319959320. doi: 10.1007/978-3-319-95933-7_74. URL http://link.springer.com/10.1007/978-3-319-95933-7_74.

[30] Vitoantonio Bevilacqua, Gianpaolo Francesco Trotta, Antonio Brunetti, Nicholas Caporusso, Claudio Loconsole, Giacomo Donato Cascarano, Francesco Catino, Pantaleo Cozzoli, Giancarlo Delfine, Adriano Mastronardi, Andrea Di Candia, Giuseppina Lelli, and Pietro Fiore. A comprehensive approach for physical rehabilitation assessment in multiple sclerosis patients based on gait analysis. In Vincent Duffy and Nancy Lightner, editors, *Advances in Human Factors and Ergonomics in Healthcare and Medical Devices*, pages 119–128. Springer International Publishing, Cham, 2018. ISBN 978-3-319-60483-1. doi: 10.1007/978-3-319-60483-1_13.

[31] Ilaria Bortone, Gianpaolo Francesco Trotta, Antonio Brunetti, Giacomo Donato Cascarano, Claudio Loconsole, Nadia Agnello, Alberto Argentiero, Giuseppe Nicolardi, Antonio Frisoli, and Vitoantonio Bevilacqua. A novel approach in combination of 3d gait analysis data for aiding clinical decision-making in patients with parkinson's disease. In De-Shuang Huang, Kang-Hyun Jo, and Juan Carlos Figueroa-García, editors, *Intelligent Computing Theories and Application*, pages 504–514. Springer International Publishing, Cham, 2017. ISBN 978-3-319-63312-1. doi: 10.1007/978-3-319-63312-1_44.

[32] Vitoantonio Bevilacqua, Antonio Emmanuele Uva, Michele Fiorentino, Gianpaolo Francesco Trotta, Maurizio Dimatteo, Enrico Nasca, Attilio Nicola Nocera, Giacomo Donato Cascarano, Antonio Brunetti, Nicholas Caporusso, Roberta Pellicciari, and Giovanni Defazio. A Comprehensive Method for Assessing the Blepharospasm Cases Severity. In K.C. Santosh Mallikarjun Hangarge Vitoantonio Bevilacqua Atul Negi, editor, *Communications in Computer and Information Science*, volume 709, pages 369–381. Springer, Singapore -, 2017. ISBN 9789811048586. doi: 10.1007/978-981-10-4859-3_33. URL https://link.springer.com/chapter/10.1007/978-981-10-4859-3_33.

References

[1] Heang-Ping Chan, Lubomir M. Hadjiiski, and Ravi K. Samala. Computer-aided diagnosis in the era of deep learning. *Medical Physics*, 47(5):e218–e227, 2020. doi: https://doi.org/10.1002/mp.13764. URL https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.13764.

[2] Edward Hance Shortliffe and Bruce G Buchanan. *Rule-based expert systems: the MYCIN experiments of the Stanford Heuristic Programming Project*. Addison-Wesley Publishing Company, 1985.

[3] Donald Waterman. A guide to expert systems. 1986.

[4] Shu-Hsien Liao. Expert system methodologies and applications—a decade review from 1995 to 2004. *Expert systems with applications*, 28(1):93–103, 2005.

[5] Arati Gurung, Carolyn G Scrafford, James M Tielsch, Orin S Levine, and William Checkley. Computerized lung sound analysis as diagnostic aid for the detection of abnormal lung sounds: a systematic review and meta-analysis. *Respiratory medicine*, 105(9):1396–1403, 2011.

[6] Fabian J Theis and Anke Meyer-Bäse. *Biomedical signal analysis: Contemporary methods and applications*. MIT Press, 2010.

[7] Ziad Obermeyer and Ezekiel J Emanuel. Predicting the future—big data, machine learning, and clinical medicine. *The New England journal of medicine*, 375(13):1216, 2016.

[8] Wenqing Sun, Bin Zheng, and Wei Qian. Computer aided lung cancer diagnosis with deep learning algorithms. In *Medical imaging 2016: computer-aided diagnosis*, volume 9785, page 97850Z. International Society for Optics and Photonics, 2016.

[9] Maryellen L Giger. Machine learning in medical imaging. *Journal of the American College of Radiology*, 15(3):512–520, 2018.

[10] Antonio Brunetti, Leonarda Carnimeo, Gianpaolo Francesco Trotta, and Vitoantonio Bevilacqua. Computer-assisted frameworks for classification of liver, breast and blood neoplasias via neural networks: A survey based on medical images. *Neurocomputing*, 335:274 – 298, 2019. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2018.06.080.

[11] Harry Weinrauch and Albert W Hetherington. Computers in medicine and biology. *Journal of the American Medical Association*, 169(3):240–245, 1959.

[12] Robert S Ledley. Digital electronic computers in biomedical science. *Science*, 130 (3384):1225–1234, 1959.

[13] Steven G Vandenberg. Medical diagnosis by computer: Recent attempts and outlook for the future. *Behavioral Science*, 5(2):170, 1960.

[14] Robert S Ledley and Lee B Lusted. Reasoning foundations of medical diagnosis. *Science*, 130(3366):9–21, 1959.

[15] Sholom M Weiss, Casimir A Kulikowski, Saul Amarel, and Aran Safir. A model-based method for computer-aided medical decision-making. *Artificial intelligence*, 11(1-2): 145–172, 1978.

[16] Gwilym S Lodwick, Cosmo L Haun, Walton E Smith, Roy F Keller, and Eddie D Robertson. Computer diagnosis of primary bone tumors: A preliminary report. *Radiology*, 80 (2):273–275, 1963.

[17] Phillip H Meyers, Charles M Nice Jr, Hal C Becker, Wilson J Nettleton Jr, James W Sweeney, and George R Meckstroth. Automated computer analysis of radiographic images. *Radiology*, 83(6):1029–1034, 1964.

[18] Fred Winsberg, Milton Elkin, Josiah Macy Jr, Victoria Bordaz, and William Weymouth. Detection of radiographic abnormalities in mammograms by means of optical scanning and computer analysis. *Radiology*, 89(2):211–215, 1967.

[19] Richard M Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.

[20] Mark A Musen, Blackford Middleton, and Robert A Greenes. Clinical decision-support systems. In *Biomedical informatics*, pages 643–674. Springer, 2014.

[21] Kunio Doi. Computer-aided diagnosis in medical imaging: historical review, current status and future potential. *Computerized medical imaging and graphics*, 31(4-5): 198–211, 2007.

[22] SM Astley and Fiona Jane Gilbert. Computer-aided detection in mammography. *Clinical radiology*, 59(5):390–399, 2004.

[23] Kunio Doi. Current status and future potential of computer-aided diagnosis in medical imaging. *The British journal of radiology*, 78(suppl_1):s3–s19, 2005.

[24] Maciej A. Mazurowski, Piotr A. Habas, Jacek M. Zurada, Joseph Y. Lo, Jay A. Baker, and Georgia D. Tourassi. Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks*, 21(2):427 – 436, 2008. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2007.

12.031. URL http://www.sciencedirect.com/science/article/pii/S0893608007002407. Advances in Neural Networks Research: IJCNN '07.

[25] Xingquan Zhu and Xindong Wu. Class noise vs. attribute noise: A quantitative study. *Artificial intelligence review*, 22(3):177–210, 2004.

[26] Robert Nisbet, John Elder, and Gary Miner. Chapter 4 - data understanding and preparation. In Robert Nisbet, John Elder, and Gary Miner, editors, *Handbook of Statistical Analysis and Data Mining Applications*, pages 49 – 75. Academic Press, Boston, 2009. ISBN 978-0-12-374765-5. doi: 10.1016/B978-0-12-374765-5.00004-8.

[27] Davide Chicco. Ten quick tips for machine learning in computational biology, 2017. ISSN 17560381.

[28] Vitoantonio Bevilacqua, A. Aulenta, E. Carioggia, Giuseppe Mastronardi, Filippo Menolascina, G. Simeone, Angelo Paradiso, Antonio Scarpa, and Diego Taurino. Metallic artifacts removal in breast CT images for treatment planning in radiotherapy by means of supervised and unsupervised neural network algorithms. In De-Shuang Huang, Laurent Heutte, and Marco Loog, editors, *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues, Third International Conference on Intelligent Computing, ICIC 2007, Qingdao, China, August 21-24, 2007, Proceedings*, volume 4681 of *Lecture Notes in Computer Science*, pages 1355–1363. Springer, 2007. ISBN 978-3-540-74170-1. doi: 10.1007/978-3-540-74171-8_138.

[29] Muna O Al-Hatmi and Jabar H Yousif. A review of image enhancement systems and a case study of salt &pepper noise removing. *International Journal of Computation and Applied Sciences IJOCAAS*, 3(2):217–223, 2017.

[30] B Suneetha and A JhansiRani. A survey on image processing techniques for brain tumor detection using magnetic resonance imaging. In *Innovations in Green Energy and Healthcare Technologies (IGEHT), 2017 International Conference on*, pages 1–6. IEEE, 2017.

[31] G Niranjana and M Ponnavaikko. A review on image processing methods in detecting lung cancer using ct images. In *Technical Advancements in Computers and Communications (ICTACC), 2017 International Conference on*, pages 18–25. IEEE, 2017.

[32] King-Sun Fu and JK Mui. A survey on image segmentation. *Pattern recognition*, 13(1): 3–16, 1981.

[33] Hui Zhang, Jason E Fritts, and Sally A Goldman. Image segmentation evaluation: A survey of unsupervised methods. *computer vision and image understanding*, 110(2): 260–280, 2008.

[34] Parnian Afshar, Arash Mohammadi, Konstantinos N Plataniotis, Anastasia Oikonomou, and Habib Benali. From handcrafted to deep-learning-based cancer radiomics: challenges and opportunities. *IEEE Signal Processing Magazine*, 36(4):132–160, 2019.

[35] Ji Eun Park, Seo Young Park, Hwa Jung Kim, and Ho Sung Kim. Reproducibility and generalizability in radiomics modeling: possible strategies in radiologic and statistical perspectives. *Korean journal of radiology*, 20(7):1124–1137, 2019.

[36] Robert M Haralick, Karthikeyan Shanmugam, et al. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[37] Chengjun Sun and William G Wee. Neighboring gray level dependence matrix for texture classification. *computer vision, graphics, and image processing*, 23(3):341–352, 1983.

[38] Alex Zwanenburg, Martin Vallières, Mahmoud A Abdalah, Hugo JWL Aerts, Vincent Andrearczyk, Aditya Apte, Saeed Ashrafinia, Spyridon Bakas, Roelof J Beukinga, Ronald Boellaard, et al. The image biomarker standardization initiative: standardized quantitative radiomics for high-throughput image-based phenotyping. *Radiology*, 295 (2):328–338, 2020.

[39] Hugo J. W. L. Aerts, Emmanuel Rios Velazquez, Ralph T. H. Leijenaar, Chintan Parmar, Patrick Grossmann, Sara Carvalho, Johan Bussink, René Monshouwer, Benjamin Haibe-Kains, Derek Rietveld, Frank Hoebers, Michelle M. Rietbergen, C. René Leemans, Andre Dekker, John Quackenbush, Robert J. Gillies, and Philippe Lambin. Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach. *Nature Communications*, 5(1):4006, 2014. doi: 10.1038/ncomms5006. URL https://doi.org/10.1038/ncomms5006.

[40] Mathieu Hatt, Florent Tixier, Larry Pierce, Paul E Kinahan, Catherine Cheze Le Rest, and Dimitris Visvikis. Characterization of pet/ct images using texture analysis: the past, the present... any future? *European journal of nuclear medicine and molecular imaging*, 44(1):151–165, 2017.

[41] Antonio Brunetti, Giacomo Donato Cascarano, and Vitoantonio Bevilacqua. Deep Learning for Medical Imaging in the Era of Precision Medicine. In Riccardo Bellazzi, Cecilia Laschi, Silvana Quaglini, and Lucia Sacchi, editors, *AI-enabled health care: from decision support to autonomous robots*, pages 75 – 99. PÀTRON EDITORE, BOLOGNA, 2020.

[42] Mary M. Galloway. Texture analysis using gray level run lengths. *Computer Graphics and Image Processing*, 4(2):172 – 179, 1975. ISSN 0146-664X. doi: https://doi.org/10.1016/S0146-664X(75)80008-6. URL http://www.sciencedirect.com/science/article/pii/S0146664X75800086.

[43] G. Thibault, J. Angulo, and F. Meyer. Advanced statistical matrices for texture characterization: Application to cell classification. *IEEE Transactions on Biomedical Engineering*, 61(3):630–637, 2014. doi: 10.1109/TBME.2013.2284600.

[44] M. Amadasun and R. King. Textural features corresponding to textural properties. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5):1264–1274, 1989. doi: 10.1109/21.44046.

[45] Stefania Rizzo, Francesca Botta, Sara Raimondi, Daniela Origgi, Cristiana Fanciullo, Alessio Giuseppe Morganti, and Massimo Bellomi. Radiomics: the facts and the challenges of image analysis. *European radiology experimental*, 2(1):1–8, 2018.

[46] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987.

[47] Pyradiomics - radiomic features. URL https://pyradiomics.readthedocs.io/en/latest/features.html. Last visited: November 2020.

[48] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[49] Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10:66–71, 2009.

[50] Bo Li, Chao Wang, and De-Shuang Huang. Supervised feature extraction based on orthogonal discriminant projection. *Neurocomputing*, 73(1-3):191–196, 2009.

[51] De-Shuang Huang and Jian-Xun Mi. A new constrained independent component analysis method. *IEEE transactions on neural networks*, 18(5):1532–1535, 2007.

[52] Zhan-Li Sun, De-Shuang Huang, Chun-Hou Zheng, and Li Shang. Optimal selection of time lags for tdsep based on genetic algorithm. *Neurocomputing*, 69(7-9):884–887, 2006.

[53] Chun-Hou Zheng, De-Shuang Huang, Zhan-Li Sun, Michael R Lyu, and Tat-Ming Lok. Nonnegative independent component analysis based on minimizing mutual information technique. *Neurocomputing*, 69(7-9):878–883, 2006.

[54] Chun-Hou Zheng, De-Shuang Huang, Kang Li, George Irwin, and Zhan-Li Sun. Misep method for postnonlinear blind source separation. *Neural computation*, 19(9):2557–2578, 2007.

[55] Chun-Hou Zheng, De-Shuang Huang, and Li Shang. Feature selection in independent component subspace for microarray data classification. *Neurocomputing*, 69(16-18): 2407–2410, 2006.

[56] Imola K Fodor. A survey of dimension reduction techniques. Technical report, Lawrence Livermore National Lab., CA (US), 2002.

[57] De-Shuang Huang and Wen Jiang. A general cpl-ads methodology for fixing dynamic parameters in dual environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(5):1489–1500, 2012.

[58] IA Basheer and M Hajmeer. Artificial neural networks: fundamentals, computing, design, and application. *Journal of microbiological methods*, 43(1):3–31, 2000.

[59] Eric B Baum and David Haussler. What size net gives valid generalization? In *Advances in neural information processing systems*, pages 81–90, 1989.

[60] Farid U Dowla and Leah L Rogers. *Solving problems in environmental engineering and geosciences with artificial neural networks*. Mit Press, 1995.

[61] Simon Haykin and Neural Network. A comprehensive foundation. *Neural Networks*, 2 (2004):41, 2004.

[62] Timothy Masters. *Practical neural network recipes in C++*. Morgan Kaufmann, 1993.

[63] Carl G Looney. Advances in feedforward neural networks: demystifying knowledge acquiring black boxes. *IEEE Transactions on Knowledge and Data Engineering*, 8(2): 211–226, 1996.

[64] Kevin Swingler. *Applying neural networks: a practical guide*. Morgan Kaufmann, 1996.

[65] Sylvain Arlot, Alain Celisse, et al. A survey of cross-validation procedures for model selection. *Statistics surveys*, 4:40–79, 2010.

[66] Vitoantonio Bevilacqua, Maurizio Triggiani, Maurizio Dimatteo, Giuseppe Bellantuono, Antonio, Leonarda Carnimeo, Francescomaria Marino, Michele Telegrafo, and Marco Moschetta. Computer assisted detection of breast lesions in magnetic resonance images. In De-Shuang Huang, Vitoantonio Bevilacqua, and Prashan Premaratne, editors, *Intelligent Computing Theories and Application - 12th International Conference, ICIC 2016, Lanzhou, China, August 2-5, 2016, Proceedings, Part I*, volume 9771 of *Lecture Notes in Computer Science*, pages 306–316. Springer, 2016. ISBN 978-3-319-42290-9. doi: 10.1007/978-3-319-42291-6_30.

[67] Vitoantonio Bevilacqua, Antonio Brunetti, Maurizio Triggiani, Domenico Magaletti, Michele Telegrafo, and Marco Moschetta. An Optimized Feed-forward Artificial Neural Network Topology to Support Radiologists in Breast Lesions Classification. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion - GECCO '16 Companion*, pages 1385–1392, New York, New York, USA, 2016. ACM, ACM Press. ISBN 9781450343237. doi: 10.1145/2908961.2931733. URL http://dl.acm.org/citation.cfm?doid=2908961.2931733.

[68] Shubhi Sharma and Pritee Khanna. Computer-aided diagnosis of malignant mammograms using zernike moments and svm. *Journal of Digital Imaging*, 28(1):77–90, Feb 2015. ISSN 1618-727X. doi: 10.1007/s10278-014-9719-7.

[69] J Dheeba, N Albert Singh, and S Tamil Selvi. Computer-aided detection of breast cancer on mammograms: A swarm intelligence optimized wavelet neural network approach. *Journal of biomedical informatics*, 49:45–52, 2014.

[70] Ismail Saritas. Prediction of breast cancer using artificial neural networks. *Journal of Medical Systems*, 36(5):2901–2907, 2012.

[71] Vitoantonio Bevilacqua, Paolo Pannarale, Mirko Abbrescia, Claudia Cava, Angelo Paradiso, and Stefania Tommasi. Comparison of data-merging methods with svm attribute selection and classification in breast cancer gene expression. In *BMC bioinformatics*, volume 13, pages 1–15. BioMed Central, 2012.

[72] Hui-Ling Chen, Bo Yang, Jie Liu, and Da-You Liu. A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis. *Expert Systems with Applications*, 38(7):9014–9022, 2011.

[73] Jianmin Jiang, P Trundle, and Jinchang Ren. Medical image analysis with artificial neural networks. *Computerized Medical Imaging and Graphics*, 34(8):617–631, 2010.

[74] RR Janghel, Anupam Shukla, Ritu Tiwari, and Rahul Kala. Breast cancer diagnosis using artificial neural network models. In *Information Sciences and Interaction Sciences (ICIS), 2010 3rd International Conference on*, pages 89–94. IEEE, 2010.

[75] Dustin Newell, Ke Nie, Jeon-Hor Chen, Chieh-Chih Hsu, Hon J. Yu, Orhan Nalcioglu, and Min-Ying Su. Selection of diagnostic features on breast mri to differentiate between malignant and benign lesions using computer-aided diagnosis: differences in lesions presenting as mass and non-mass-like enhancement. *European Radiology*, 20(4): 771–781, Apr 2010. ISSN 1432-1084. doi: 10.1007/s00330-009-1616-y.

[76] Y Rejani and S Thamarai Selvi. Early detection of breast cancer using svm classifier technique. *arXiv preprint arXiv:0912.2314*, 2009.

[77] Mehmet Fatih Akay. Support vector machines combined with feature selection for breast cancer diagnosis. *Expert systems with applications*, 36(2):3240–3247, 2009.

[78] Vitoantonio Bevilacqua, Giuseppe Mastronardi, Filippo Menolascina, Paolo Pannarale, and Antonio Pedone. A novel multi-objective genetic algorithm approach to artificial neural network topology optimisation: The breast cancer classification problem. In *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2006, part of the IEEE World Congress on Computational Intelligence, WCCI 2006, Vancouver, BC, Canada, 16-21 July 2006*, pages 1958–1965. IEEE, 2006. ISBN 0-7803-9490-9. doi: 10.1109/IJCNN.2006.246940.

[79] Hussein A Abbass. An evolutionary artificial neural networks approach for breast cancer diagnosis. *Artificial intelligence in Medicine*, 25(3):265–281, 2002.

[80] De-Shuang Huang. Systematic theory of neural networks for pattern recognition. *Publishing House of Electronic Industry of China, Beijing*, 201, 1996.

[81] De-shuang Huang. Radial basis probabilistic neural networks: Model and application. *International Journal of Pattern Recognition and Artificial Intelligence*, 13(07):1083–1101, 1999.

[82] De-Shuang Huang and Ji-Xiang Du. A constructive hybrid structure optimization methodology for radial basis probabilistic neural networks. *IEEE Transactions on neural networks*, 19(12):2099–2115, 2008.

[83] Subana Shanmuganathan. Artificial neural network modelling: An introduction. In *Artificial neural network modelling*, pages 1–14. Springer, Cham, 2016.

[84] Johan A K Suykens, Joos P L Vandewalle, and Bart L De Moor. *Artificial neural networks for modelling and control of non-linear systems*. Springer Science and Business Media, 2012.

[85] Giacomo Donato Cascarano, Claudio Loconsole, Antonio Brunetti, Antonio Lattarulo, Domenico Buongiorno, Giacomo Losavio, Eugenio Di Sciascio, and Vitoantonio Bevilacqua. Biometric handwriting analysis to support Parkinson's Disease assessment and grading. *BMC Medical Informatics and Decision Making*, 19(S9): 252, dec 2019. ISSN 1472-6947. doi: 10.1186/s12911-019-0989-3. URL https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0989-3.

[86] Claudio Loconsole, Giacomo Donato Cascarano, Antonio Lattarulo, Antonio Brunetti, Gianpaolo Francesco Trotta, Domenico Buongiorno, Ilaria Bortone, Irio De Feudis, Giacomo Losavio, Vitoantonio Bevilacqua, and Eugenio Di Sciascio. A comparison between ANN and SVM classifiers for Parkinson's disease by using a model-free computer-assisted handwriting analysis based on biometric signals. In *2018 International Joint Conference on Neural Networks (IJCNN)*, volume 2018-July, pages 1–8, Piscataway, NJ -, jul 2018. IEEE. ISBN 978-1-5090-6014-6. doi: 10.1109/IJCNN.2018.8489293. URL https://ieeexplore.ieee.org/document/8489293/.

[87] Claudio Loconsole, Giacomo Donato Cascarano, Antonio Brunetti, Gianpaolo Francesco Trotta, Giacomo Losavio, Vitoantonio Bevilacqua, and Eugenio Di Sciascio. A model-free technique based on computer vision and sEMG for classification in Parkinson's disease by using computer-assisted handwriting analysis. *Pattern Recognition Letters*, 121:28–36, apr 2019. ISSN 01678655. doi: 10.1016/j.patrec.2018.04.006. URL https://www.sciencedirect.com/science/article/pii/S0167865518301260.

[88] Vitoantonio Bevilacqua, Gianpaolo Francesco Trotta, Antonio Brunetti, Nicholas Caporusso, Claudio Loconsole, Giacomo Donato Cascarano, Francesco Catino, Pantaleo Cozzoli, Giancarlo Delfine, Adriano Mastronardi, Andrea Di Candia, Giuseppina Lelli, and Pietro Fiore. A Comprehensive Approach for Physical Rehabilitation Assessment in Multiple Sclerosis Patients Based on Gait Analysis. In *Advances in Intelligent Systems and Computing*, volume 590, pages 119–128. 2018. ISBN 9783319604824. doi: 10.1007/978-3-319-60483-1_13. URL http://link.springer.com/10.1007/978-3-319-60483-1_13.

[89] Vitoantonio Bevilacqua, Claudio Loconsole, Antonio Brunetti, Giacomo Donato Cascarano, Antonio Lattarulo, Giacomo Losavio, and Eugenio Di Sciascio. A Model-Free Computer-Assisted Handwriting Analysis Exploiting Optimal Topology ANNs on

Biometric Signals in Parkinson's Disease Research. In De-Shuang Huang Kang-Hyun Jo Xiao-Long Zhang, editor, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10955 LNCS, pages 650–655. Springer, Cham, CH - CHE, aug 2018. ISBN 9783319959320. doi: 10.1007/978-3-319-95933-7_74. URL http://link.springer.com/10.1007/978-3-319-95933-7_74.

[90] W.S. McCulloch and W.H. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5:115–133, 1943.

[91] Andrej Krenker, Andrej Kos, and Janez Bešter. *Introduction to the artificial neural networks*. INTECH Open Access Publisher, 2011.

[92] Marvin Minsky and Seymour A. Papert. *Perceptrons: An Introduction to Computational Geometry*. MIT Press, 1969.

[93] J Hopfield. Neural networks and physical systems with emergent collective computational abilities. In *Proc. of National Academy of Sciences of the United States of America*, volume 79, pages 2554–2558, 2006.

[94] Stuart J Russel and Peter Norvig. *Artificial Intelligence: A Modern Approach*. 3rd edition, 2010. ISBN 0136042597.

[95] C. E. Rasmussen and Z. Ghahramani. Occam's Razor. In *Proc. of Advances in Neural Information Processing Systems*, 2001.

[96] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[97] Shaohua Kevin Zhou, Hayit Greenspan, and Dinggang Shen. *Deep Learning for Medical Image Analysis*. Academic Press, 2017.

[98] Yann A LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012.

[99] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

[100] T Ching, D S Himmelstein, B K Beaulieu-Jones, A A Kalinin, G P Way, E Ferrero, P M Agapow, M Zietz, M M Hoffman, W Xie, G L Rosen, B J Lengerich, et al. Opportunities and obstacles for deep learning in biology and medicine. *Journal of the Royal Society Interface*, 15(141):1–47, 2018.

[101] K. Fukushima. Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4): 193–202, 1980.

[102] Yann Lecun, Leon Bottou, Yoshua Bengio, and Patrick Ha. LeNet. *Proceedings of the IEEE*, 1998. ISSN 00189219. doi: 10.1109/5.726791.

[103] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012 AlexNet. *Advances In Neural Information Processing Systems*, 2012. ISSN 10495258. doi: http://dx.doi.org/10.1016/j.protcy.2014.09.007.

[104] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, dec 2017. ISSN 13618423. doi: 10.1016/j.media.2017.07.005.

[105] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E. Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 2017. ISSN 18728286. doi: 10.1016/j.neucom.2016.12.038.

[106] Zhan-Li Sun, De-Shuang Huang, and Yiu-Ming Cheun. Extracting nonlinear features for multispectral images by fcmc and kpca. *Digital Signal Processing*, 15(4):331–346, 2005.

[107] D. D Huang and Songde Ma. Linear and nonlinear feedforward neural network classifiers: A comprehensive understanding. *Journal of Intelligent Systems*, 9:1 – 38, 1999.

[108] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 242–252, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/allen-zhu19a.html.

[109] Ian J. Goodfellow, Oriol Vinyals, and Andrew M. Saxe. Qualitatively characterizing neural network optimization problems, 2014.

[110] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.

[111] Daniel Justus, John Brennan, Stephen Bonner, and Andrew Stephen McGough. Predicting the computational cost of deep learning models. *2018 IEEE International Conference on Big Data (Big Data)*, Dec 2018. doi: 10.1109/bigdata.2018.8622396. URL http://dx.doi.org/10.1109/BigData.2018.8622396.

[112] Liu Yuille. Limitations of deep learning for vision, and how we might fix them. *The Gradient*, 2019.

[113] Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S. Lew. Deep learning for visual understanding: A review. *Neurocomputing*, 2016. ISSN 18728286. doi: 10.1016/j.neucom.2015.09.116.

[114] Domenico Buongiorno, Giacomo Donato Cascarano, Antonio Brunetti, Irio De Feudis, and Vitoantonio Bevilacqua. A Survey on Deep Learning in Electromyographic Signal Analysis. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11645 LNAI, pages 751–761. Springer, Cham, CH, 2019. ISBN 9783030267650. doi: 10.1007/978-3-030-26766-7_68. URL http://link.springer.com/10.1007/978-3-030-26766-7{_}68.

[115] Domenico Buongiorno, Giacomo Donato Cascarano, Irio De Feudis, Antonio Brunetti, Leonarda Carnimeo, Giovanni Dimauro, and Vitoantonio Bevilacqua. Deep learning for processing electromyographic signals: A taxonomy-based survey. *Neurocomputing*, 2020. ISSN 0925-2312. doi: 10.1016/j.neucom.2020.06.139. URL http://www.sciencedirect.com/science/article/pii/S0925231220319020.

[116] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[117] Guohui Chuai, Hanhui Ma, Jifang Yan, Ming Chen, Nanfang Hong, Dongyu Xue, Chi Zhou, Chenyu Zhu, Ke Chen, Bin Duan, et al. Deepcrispr: optimized crispr guide rna design by deep learning. *Genome biology*, 19(1):80, 2018.

[118] Lei Wang, Zhu-Hong You, De-shuang Huang, and Fengfeng Zhou. Combining high speed elm learning with a deep convolutional neural network feature encoding for predicting protein-rna interactions. *IEEE/ACM transactions on computational biology and bioinformatics*, 2018.

[119] Di Wu, Si-Jia Zheng, Chang-An Yuan, and De-Shuang Huang. A deep model with combined losses for person re-identification. *Cognitive Systems Research*, 54:74–82, 2019.

[120] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. Technical report.

[121] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, 2017-Octob: 2980–2988, 2017. ISSN 15505499. doi: 10.1109/ICCV.2017.322.

[122] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017. ISBN 9781538604571. doi: 10.1109/CVPR.2017.549.

[123] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR

'14, pages 580–587, Washington, DC, USA, 2014. IEEE Computer Society. ISBN 978-1-4799-5118-5. doi: 10.1109/CVPR.2014.81.

[124] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[125] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[126] Yukun Zhu, Raquel Urtasun, Ruslan Salakhutdinov, and Sanja Fidler. segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4703–4711, 2015.

[127] Spyros Gidaris and Nikos Komodakis. Object detection via a multi-region and semantic segmentation-aware cnn model. In *Proceedings of the IEEE international conference on computer vision*, pages 1134–1142, 2015.

[128] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. Technical report.

[129] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.

[130] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 77, pages 770–778. IEEE, jun 2016. ISBN 978-1-4673-8851-1. doi: 10.1109/CVPR.2016.90.

[131] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. 2016. ISSN 01678655. doi: 10.1016/j.patrec.2014.01.008.

[132] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[133] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10590-1. doi: 10.1007/978-3-319-10590-1_53.

[134] Vishwa S Parekh and Michael A Jacobs. Deep learning and radiomics in precision medicine. *Expert review of precision medicine and drug development*, 4(2):59–72, 2019. doi: 10.1080/23808993.2019.1585805.

[135] A. Bizzego, N. Bussola, D. Salvalai, M. Chierici, V. Maggio, G. Jurman, and C. Furlanello. Integrating deep and radiomics features in cancer bioimaging. In *2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–8, 2019. doi: 10.1109/CIBCB.2019.8791473.

[136] P. Afshar, A. Mohammadi, K. N. Plataniotis, A. Oikonomou, and H. Benali. From handcrafted to deep-learning-based cancer radiomics: Challenges and opportunities. *IEEE Signal Processing Magazine*, 36(4):132–160, 2019. doi: 10.1109/MSP.2019.2900993.

[137] Ahmed Hosny, Chintan Parmar, Thibaud P Coroller, Patrick Grossmann, Roman Zeleznik, Avnish Kumar, Johan Bussink, Robert J Gillies, Raymond H Mak, and Hugo JWL Aerts. Deep learning for lung cancer prognostication: A retrospective multi-cohort radiomics study. *PLoS medicine*, 15(11):e1002711, 2018. doi: 10.1371/journal.pmed.1002711.

[138] W. Han, L. Qin, C. Bay, X. Chen, K.-H. Yu, N. Miskin, A. Li, X. Xu, and G. Young. Deep transfer learning and radiomics feature prediction of survival of patients with high-grade gliomas. *American Journal of Neuroradiology*, 41(1):40–48, 2020. ISSN 0195-6108. doi: 10.3174/ajnr.A6365. URL http://www.ajnr.org/content/41/1/40.

[139] June-Goo Lee, Sanghoon Jun, Young-Won Cho, Hyunna Lee, Guk Bae Kim, Joon Beom Seo, and Namkug Kim. Deep learning in medical imaging: general overview. *Korean journal of radiology*, 18(4):570–584, 2017. doi: 10.3348/kjr.2017.18.4.570.

[140] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

[141] Nianyin Zeng, Hong Zhang, Baoye Song, Weibo Liu, Yurong Li, and Abdullah M Dobaie. Facial expression recognition via learning deep sparse autoencoders. *Neurocomputing*, 273:643–649, 2018.

[142] Fanghua Ye, Chuan Chen, and Zibin Zheng. Deep autoencoder-like nonnegative matrix factorization for community detection. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1393–1402. ACM, 2018. doi: 10.1145/3269206.3271697.

[143] Hai-Cheng Yi, Zhu-Hong You, De-Shuang Huang, Xiao Li, Tong-Hai Jiang, and Li-Ping Li. A deep learning framework for robust and accurate prediction of ncrna-protein interactions using evolutionary information. *Molecular Therapy-Nucleic Acids*, 11:337–344, 2018.

[144] Cheng-Yuan Liou, Wei-Chen Cheng, Jiun-Wei Liou, and Daw-Ran Liou. Autoencoder for words. *Neurocomputing*, 139:84–96, 2014.

[145] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[146] Jie Zhang, Shiguang Shan, Meina Kan, and Xilin Chen. Coarse-to-fine auto-encoder networks (cfan) for real-time face alignment. In *European conference on computer vision*, pages 1–16. Springer, 2014.

[147] Xiaojuan Jiang, Yinghua Zhang, Wensheng Zhang, and Xian Xiao. A novel sparse auto-encoder for deep unsupervised learning. In *2013 Sixth International Conference on Advanced Computational Intelligence (ICACI)*, pages 256–261. IEEE, 2013.

[148] Yingbo Zhou, Devansh Arpit, Ifeoma Nwogu, and Venu Govindaraju. Is joint training better for deep auto-encoders? *arXiv preprint arXiv:1405.1380*, 2014.

[149] Marc'Aurelio Ranzato, Christopher Poultney, Sumit Chopra, and Yann L Cun. Efficient learning of sparse representations with an energy-based model. In *Advances in neural information processing systems*, pages 1137–1144, 2007.

[150] Ian Goodfellow, Honglak Lee, Quoc V Le, Andrew Saxe, and Andrew Y Ng. Measuring invariances in deep networks. In *Advances in neural information processing systems*, pages 646–654, 2009.

[151] Quoc V Le, Jiquan Ngiam, Adam Coates, Abhik Lahiri, Bobby Prochnow, and Andrew Y Ng. On optimization methods for deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 265–272. Omnipress, 2011.

[152] Will Y Zou, Andrew Y Ng, and Kai Yu. Unsupervised learning of visual invariance with temporal coherence. In *NIPS 2011 workshop on deep learning and unsupervised feature learning*, volume 3, 2011.

[153] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.

[154] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE transactions on pattern analysis and machine intelligence*, 34 (11):2189–2202, 2012.

[155] Ian Endres and Derek Hoiem. Category-independent object proposals with diverse ranking. *IEEE transactions on pattern analysis and machine intelligence*, 36(2):222–234, 2013.

[156] Jasper R. R. Uijlings, Koen E. A. van de Sande, Theo Gevers, and Arnold W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013. doi: 10.1007/s11263-013-0620-5.

[157] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2009.

[158] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, 2015. doi: 10.1109/TPAMI.2015.2389824.

[159] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms–improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer vision*, pages 5561–5569, 2017.

[160] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. ISSN 01628828. doi: 10.1109/TPAMI.2016.2644615.

[161] Nicola Altini, Giacomo Donato Cascarano, Antonio Brunetti, Francescomaria Marino, Maria Teresa Rocchetti, Silvia Matino, Umberto Venere, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio Bevilacqua. Semantic Segmentation Framework for Glomeruli Detection and Classification in Kidney Histological Sections. *Electronics*, 9(3):503, mar 2020. ISSN 2079-9292. doi: 10.3390/electronics9030503. URL https://www.mdpi.com/2079-9292/9/3/503.

[162] Zhaoxuan Ma, Jiayun Li, Hootan Salemi, Corey Arnold, Beatrice S. Knudsen, Arkadiusz Gertych, and Nathan Ing. Semantic segmentation for prostate cancer grading by convolutional neural networks. (March):46, 2018. ISSN 16057422. doi: 10.1117/12.2293000.

[163] Jing Tang, Jun Li, and Xiangping Xu. Segnet-based gland segmentation from colon cancer histology images. In *2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 1078–1082. IEEE, 2018.

[164] Salma Alqazzaz, Xianfang Sun, Xin Yang, and Len Nokes. Automated brain tumor segmentation on multi-modal mr image using segnet. *Computational Visual Media*, 5 (2):209–219, 2019.

[165] Liang Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11211 LNCS:833–851, 2018. ISSN 16113349. doi: 10.1007/978-3-030-01234-2_49.

[166] Liang Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018. ISSN 01628828. doi: 10.1109/TPAMI.2017.2699184.

[167] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[168] Wei-Ting Xiao, Li-Jen Chang, and Wei-Min Liu. Semantic segmentation of colorectal polyps with deeplab and lstm networks. In *2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, pages 1–2. IEEE, 2018.

[169] Wei Tang, Dongsheng Zou, Su Yang, and Jing Shi. Dsl: Automatic liver segmentation with faster r-cnn and deeplab. In *International Conference on Artificial Neural Networks*, pages 137–147. Springer, 2018.

[170] Wei Tang, Dongsheng Zou, Su Yang, Jing Shi, Jingpei Dan, and Guowu Song. A two-stage approach for automatic liver segmentation with faster r-cnn and deeplab. *Neural Computing and Applications*, pages 1–10, 2020.

[171] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9351:234–241, 2015. ISSN 16113349. doi: 10.1007/978-3-319-24574-4_28.

[172] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.

[173] F. Milletari, N. Navab, and S. Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571, 2016. doi: 10.1109/3DV.2016.79.

[174] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[175] Nicola Altini, Berardino Prencipe, Antonio Brunetti, Gioacchino Brunetti, Vito Triggiani, Leonarda Carnimeo, Francescomaria Marino, Andrea Guerriero, Laura Villani, Arnaldo Scardapane, and Giacomo Donato Cascarano. A tversky loss-based convolutional neural network for liver vessels segmentation. In De-Shuang Huang, Vitoantonio Bevilacqua, and Abir Hussain, editors, *Intelligent Computing Theories and Application*, pages 342–354, Cham, 2020. Springer International Publishing. ISBN 978-3-030-60799-9.

[176] Antonio Pepe, Jianning Li, Malte Rolf-Pissarczyk, et al. Detection, segmentation, simulation and visualization of aortic dissections: A review. *Medical Image Analysis*, 64(101773):1–16, 2020.

[177] Cem M. Deniz, Siyuan Xiang, R. Spencer Hallyburton, Arakua Welbeck, James S. Babb, Stephen Honig, Kyunghyun Cho, and Gregory Chang. Segmentation of the Proximal Femur from MR Images using Deep Convolutional Neural Networks. *Scientific Reports*, 8(16485):1–14, 2018.

[178] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[179] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[180] Aadarsh Jha, Haichun Yang, Ruining Deng, Meghan E. Kapp, Agnes B. Fogo, and Yuankai Huo. Instance Segmentation for Whole Slide Imaging: End-to-End or Detect-Then-Segment. (July):1–12, 2020. URL http://arxiv.org/abs/2007.03593.

[181] Nicola Altini, Giacomo Donato Cascarano, Antonio Brunetti, Irio De Feudis, Domenico Buongiorno, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio Bevilacqua. A deep learning instance segmentation approach for global glomerulosclerosis assessment in donor kidney biopsies. *Electronics*, 9(11):1768, Oct 2020. ISSN 2079-9292. doi: 10.3390/electronics9111768. URL http://dx.doi.org/10.3390/electronics9111768.

[182] Brian W Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2): 442–451, 1975.

[183] Amitav Banerjee, UB Chitnis, SL Jadhav, JS Bhawalkar, and S Chaudhury. Hypothesis testing, type i and type ii errors. *Industrial psychiatry journal*, 18(2):127, 2009.

[184] Geraint Lewis, Jessica Sheringham, Jamie Lopez Bernal, and Tim Crayford. *Mastering public health: a postgraduate guide to examinations and revalidation*. CRC Press, 2014.

[185] Seong Ho Park, Jin Mo Goo, and Chan-Hee Jo. Receiver operating characteristic (roc) curve: practical review for radiologists. *Korean journal of radiology*, 5(1):11–18, 2004.

[186] Lee B Lusted. Decision-making studies in patient management. *New England Journal of Medicine*, 284(8):416–424, 1971.

[187] David J Goodenough, Kurt Rossmann, and Lee B Lusted. Radiographic applications of receiver operating characteristic (roc) curves. *Radiology*, 110(1):89–95, 1974.

[188] Charles E Metz. Basic principles of roc analysis. In *Seminars in nuclear medicine*, volume 8, pages 283–298. Elsevier, 1978.

[189] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

[190] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In *European conference on computer vision*, pages 340–353. Springer, 2012.

[191] Patrick Bilic, Patrick Ferdinand Christ, Eugene Vorontsov, Grzegorz Chlebus, Hao Chen, Qi Dou, Chi-Wing Fu, Xiao Han, Pheng-Ann Heng, Jürgen Hesser, et al. The liver tumor segmentation benchmark (lits). *arXiv preprint arXiv:1901.04056*, 2019.

[192] T. Heimann, B. van Ginneken, M. A. Styner, Y. Arzhaeva, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes, F. Bello, G. Binnig, H. Bischof, A. Bornik, P. M. M. Cashman, Y. Chi, A. Cordova, B. M. Dawant, M. Fidrich, J. D. Furst, D. Furukawa, L. Grenacher, J. Hornegger, D. KainmÜller, R. I. Kitney, H. Kobatake, H. Lamecker, T. Lange, J. Lee, B. Lennon, R. Li, S. Li, H. Meinzer, G. Nemeth, D. S. Raicu, A. Rau, E. M. van Rikxoort, M. Rousson, L. Rusko, K. A. Saddi, G. Schmidt, D. Seghers, A. Shimizu, P. Slagmolen, E. Sorantin, G. Soza, R. Susomboon, J. M. Waite, A. Wimmer, and I. Wolf. Comparison and evaluation of methods for liver segmentation from ct datasets. *IEEE Transactions on Medical Imaging*, 28(8):1251–1265, 2009. doi: 10.1109/TMI.2009.2013851.

[193] What is pathology? URL https://www.mcgill.ca/pathology/about/definition. Last visited: November 2020.

[194] Anant Madabhushi and George Lee. Image analysis and machine learning in digital pathology: Challenges and opportunities. *Medical Image Analysis*, 33:170 – 175, 2016. doi: 10.1016/j.media.2016.06.037.

[195] Xin Qi, Daihou Wang, Ivan Rodero, Javier Diaz-Montes, Rebekah H Gensure, Fuyong Xing, Hua Zhong, Lauri Goodell, Manish Parashar, David J Foran, et al. Content-based histopathology image retrieval using cometcloud. *BMC bioinformatics*, 15(1):287, 2014.

[196] L. A. D. Cooper, A. B. Carter, A. B. Farris, F. Wang, J. Kong, D. A. Gutman, P. Widener, T. C. Pan, S. R. Cholleti, A. Sharma, T. M. Kurc, D. J. Brat, and J. H. Saltz. Digital pathology: Data-intensive frontier in medical imaging. *Proceedings of the IEEE*, 100(4):991–1003, April 2012. doi: 10.1109/JPROC.2011.2182074.

[197] Yukako Yagi and John R Gilbertson. Digital imaging in pathology: the case for standardization, 2005.

[198] Yoshimasa Kawazoe, Kiminori Shimamoto, Ryohei Yamaguchi, Yukako Shintani-Domoto, Hiroshi Uozaki, Masashi Fukayama, and Kazuhiko Ohe. Faster R-CNN-based glomerular detection in multistained human whole slide images. *Journal of Imaging*, 4 (7), 2018. ISSN 2313433X. doi: 10.3390/jimaging4070091.

[199] Vitoantonio Bevilacqua, Nicola Pietroleonardo, Vito Triggiani, Loreto Gesualdo, Anna Maria Di Palma, Michele Rossini, Giuseppe Dalfino, and Nico Mastrofilippo. Neural network classification of blood vessels and tubules based on haralick features evaluated in histological images of kidney biopsy. In De-Shuang Huang and Kyungsook Han, editors, *Advanced Intelligent Computing Theories and Applications*, pages 759–765, Cham, 2015. Springer International Publishing. ISBN 978-3-319-22053-6.

[200] David Ledbetter, Long Ho, and Kevin V Lemley. Prediction of Kidney Function from Biopsy Images Using Convolutional Neural Networks. pages 1–11, 2017. URL http://arxiv.org/abs/1702.01816.

[201] Vitoantonio Bevilacqua, Nicola Pietroleonardo, Vito Triggiani, Antonio Brunetti, Anna Maria Di Palma, Michele Rossini, and Loreto Gesualdo. An innovative neural network framework to classify blood vessels and tubules based on haralick features evaluated in histological images of kidney biopsy. *Neurocomputing*, 228:143–153, mar 2017. ISSN 18728286. doi: 10.1016/j.neucom.2016.09.091.

[202] Jane Hung and Anne Carpenter. Applying faster r-cnn for object detection on malaria images. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 56–61, 2017.

[203] Z. Meng, X. Fan, X. Chen, M. Chen, and Y. Tong. Detecting small signs from large images. In *2017 IEEE International Conference on Information Reuse and Integration (IRI)*, pages 217–224, Aug 2017. doi: 10.1109/IRI.2017.57.

[204] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. 6:429–449, 2002. doi: 10.3233/IDA-2002-6504.

[205] Charles Elkan. The foundations of cost-sensitive learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'01, pages 973–978, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1558608125.

[206] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249 – 259, 2018. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2018.07.011. URL http://www.sciencedirect.com/science/article/pii/S0893608018302107.

[207] David Tellez, Geert Litjens, Péter Bándi, Wouter Bulten, John-Melle Bokhorst, Francesco Ciompi, and Jeroen van der Laak. Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology. *Medical Image Analysis*, 58:101544, 2019. ISSN 13618415. doi: 10.1016/j.media.2019.101544.

[208] P. Y. Simard, D. Steinkraus, and J. C. Platt. Best practices for convolutional neural networks applied to visual document analysis. In *Seventh International Conference on*

Document Analysis and Recognition, 2003. Proceedings., pages 958–963, 2003. doi: 10.1109/ICDAR.2003.1227801.

[209] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):34–41, 2001. doi: 10.1109/38.946629.

[210] Jaime Gallego, Anibal Pedraza, Samuel Lopez, Georg Steiner, Lucia Gonzalez, Arvydas Laurinavicius, and Gloria Bueno. Glomerulus classification and detection based on convolutional neural networks. *Journal of Imaging*, 4(1), 2018. ISSN 2313433X. doi: 10.3390/jimaging4010020.

[211] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

[212] Stephen M. Pizer, E. Philip Amburn, John D. Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B. Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355 – 368, 1987. ISSN 0734-189X. doi: https://doi.org/10.1016/S0734-189X(87)80186-X. URL http://www.sciencedirect.com/science/article/pii/S0734189X8780186X.

[213] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. 4 edition, 2018. ISBN 978-1-292-22304-9.

[214] R.C. Gonzales and B.A. Fittes. Gray-level transformations for interactive image enhancement. *Mechanism and Machine Theory*, 12(1):111 – 122, 1977. ISSN 0094-114X. doi: https://doi.org/10.1016/0094-114X(77)90062-3. URL http://www.sciencedirect.com/science/article/pii/0094114X77900623. Special Issue: Robots and Manipulator Systems.

[215] Brendon Lange Neuen, Steven James Chadban, Alessandro Rhyl Demaio, David Wayne Johnson, and Vlado Perkovic. Chronic kidney disease and the global NCDs agenda. *BMJ Global Health*, 2017. doi: 10.1136/bmjgh-2017-000380.

[216] H. Wang, M. Naghavi, C. Allen, R. M. Barber, A. Carter, D. C. Casey, and et al. Global, regional, and national life expectancy, all-cause mortality, and cause-specific mortality for 249 causes of death, 1980-2015: a systematic analysis for the Global Burden of Disease Study 2015. *The Lancet*, 2016. ISSN 1474547X. doi: 10.1016/S0140-6736(16)31012-1.

[217] Robert A. Wolfe, Valarie B. Ashby, Edgar L. Milford, Akinlolu O. Ojo, Robert E. Ettenger, Lawrence Y.C. Agodoa, Philip J. Held, and Friedrich K. Port. Comparison of Mortality in All Patients on Dialysis, Patients on Dialysis Awaiting Transplantation, and Recipients of a First Cadaveric Transplant. *New England Journal of Medicine*, 2002. ISSN 0028-4793. doi: 10.1056/nejm199912023412303.

[218] Friedrich K. Port, Robert A. Wolfe, Elizabeth A. Mauger, Donald P. Berling, and Kaihong Jiang. Comparison of Survival Probabilities for Dialysis Patients vs Cadaveric Renal Transplant Recipients. *JAMA: The Journal of the American Medical Association*, 1993. ISSN 15383598. doi: 10.1001/jama.1993.03510110079036.

[219] Thaminda Liyanage, Toshiharu Ninomiya, Vivekanand Jha, Bruce Neal, Halle Marie Patrice, Ikechi Okpechi, Ming Hui Zhao, Jicheng Lv, Amit X. Garg, John Knight, Anthony Rodgers, Martin Gallagher, Sradha Kotwal, Alan Cass, and Vlado Perkovic. Worldwide access to treatment for end-stage kidney disease: A systematic review. *The Lancet*, 2015. ISSN 1474547X. doi: 10.1016/S0140-6736(14)61601-9.

[220] J M Cecka. The UNOS Scientific Renal Transplant Registry–ten years of kidney transplants. *Clin Transpl*, 1997.

[221] United Network for Organ Sharing. 2004 Annual Report.

[222] Norberto Perico, Piero Ruggenenti, Mario Scalamogna, and Giuseppe Remuzzi. Tackling the shortage of donor kidneys: How to use the best that we have, 2003. ISSN 02508095.

[223] Phillip S. Moore, Alan C. Farney, Aimee K. Sundberg, Michael S. Rohr, Erica L. Hartmann, Samy S. Iskandar, Michael D. Gautreaux, Jeffrey Rogers, William Doares, Teresa K. Anderson, Patricia L. Adams, and Robert J. Stratta. Dual kidney transplantation: A case-control comparison with single kidney transplantation from standard and expanded criteria donors, 2007. ISSN 00411337.

[224] Giuseppe Remuzzi, Josep Grinyò, Piero Ruggenenti, Marco Beatini, Edward H. Cole, Edgar L. Milford, and Barry M. Brenner. Early experience with dual kidney transplantation in adults using expanded donor criteria. *Journal of the American Society of Nephrology*, 10(12):2591–2598, 1999. ISSN 10466673.

[225] Jolanta Karpinski, Ginette Lajoie, Daniel Cattran, Stanley Fenton, Jeffrey Zaltzman, Carl Cardella, and Edward Cole. Outcome of kidney transplantation from high-risk donors is determined by both structure and function. *Transplantation*, 67(8):1162–1167, apr 1999. ISSN 0041-1337. doi: 10.1097/00007890-199904270-00013.

[226] Giuseppe Remuzzi and Piero Ruggenenti. Renal Transplantation: Single or Dual for Donors Aging > or = 60 Years? *Transplantation*, 69(10):2000–2001, 2000. ISSN 0041-1337. doi: 10.1097/00007890-200005270-00002.

[227] Giacomo Donato Cascarano, Francesco Saverio Debitonto, Ruggero Lemma, Antonio Brunetti, Domenico Buongiorno, Irio De Feudis, Andrea Guerriero, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio Bevilacqua. An Innovative Neural Network Framework for Glomerulus Classification Based on Morphological and Texture Features Evaluated in Histological Images of Kidney Biopsy. pages 727–738. 2019. doi: 10.1007/978-3-030-26766-7_66. URL http://link.springer.com/10.1007/978-3-030-26766-7{_}66.

[228] Maja Temerinac-Ott, Germain Forestier, Jessica Schmitz, Meyke Hermsen, JH Bräsen, Friedrich Feuerhake, and Cédric Wemmert. Detection of glomeruli in renal pathology by mutual comparison of multiple staining modalities. In *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis*, pages 19–24. IEEE, 2017.

[229] Michael Gadermayr, Ann-Kathrin Dombrowski, Barbara Mara Klinkhammer, Peter Boor, and Dorit Merhof. CNN Cascades for Segmenting Whole Slide Images of the Kidney. pages 1–17, 2017. URL http://arxiv.org/abs/1708.00251.

[230] Brandon Ginley, Brendon Lutnick, Kuang-Yu Jen, Agnes B. Fogo, Sanjay Jain, Avi Rosenberg, Vighnesh Walavalkar, Gregory Wilding, John E. Tomaszewski, Rabi Yacoub, Giovanni Maria Rossi, and Pinaki Sarder. Computational Segmentation and Classification of Diabetic Glomerulosclerosis. *Journal of the American Society of Nephrology*, 30(10):1953–1967, 2019. ISSN 1046-6673. doi: 10.1681/asn.2018121259.

[231] Jon N. Marsh, Matthew K. Matlock, Satoru Kudose, Ta Chiang Liu, Thaddeus S. Stappenbeck, Joseph P. Gaut, and S. Joshua Swamidass. Deep learning global glomerulosclerosis in transplant kidney frozen sections. *IEEE Transactions on Medical Imaging*, 37(12):2718–2728, 2018. ISSN 1558254X. doi: 10.1109/TMI.2018.2851150.

[232] Tsuyoshi Kato, Raissa Relator, Hayliang Ngouv, Yoshihiro Hirohashi, Osamu Takaki, Tetsuhiro Kakimoto, and Kinya Okada. Segmental HOG: New descriptor for glomerulus detection in kidney microscopy image. *BMC Bioinformatics*, 2015. ISSN 14712105. doi: 10.1186/s12859-015-0739-1.

[233] Olivier Simon, Rabi Yacoub, Sanjay Jain, John E. Tomaszewski, and Pinaki Sarder. Multi-radial LBP Features as a Tool for Rapid Glomerular Detection and Assessment in Whole Slide Histopathology Images. *Scientific Reports*, 8(1):1–11, 2018. ISSN 20452322. doi: 10.1038/s41598-018-20453-7. URL http://dx.doi.org/10.1038/s41598-018-20453-7.

[234] Gloria Bueno, M. Milagro Fernandez-Carrobles, Lucia Gonzalez-Lopez, and Oscar Deniz. Glomerulosclerosis identification in whole slide images using semantic segmentation. *Computer Methods and Programs in Biomedicine*, 184:105273, 2020. ISSN 18727565. doi: 10.1016/j.cmpb.2019.105273. URL https://doi.org/10.1016/j.cmpb.2019.105273.

[235] Taras Kotyk, Nilanjan Dey, Amira S. Ashour, Dana Balas-Timar, Sayan Chakraborty, Ahmed S. Ashour, and João Manuel R.S. Tavares. Measurement of glomerulus diameter and Bowman's space width of renal albino rats. *Computer Methods and Programs in Biomedicine*, 2016. ISSN 18727565. doi: 10.1016/j.cmpb.2015.10.023.

[236] Yan Zhao, Edgar F. Black, Luigi Marini, Kenton McHenry, Norma Kenyon, Rachana Patil, Andre Balla, and Amelia Bartholomew. Automatic glomerulus extraction in whole slide images towards computer aided diagnosis. In *Proceedings of the 2016 IEEE 12th*

International Conference on e-Science, e-Science 2016, 2017. ISBN 9781509042722. doi: 10.1109/eScience.2016.7870897.

[237] Siddharth Samsi, Wael N. Jarjour, and Ashok Krishnamurthy. Glomeruli segmentation in H&E stained tissue using perceptual organization. In *2012 IEEE Signal Processing in Medicine and Biology Symposium, SPMB 2012*, 2012. ISBN 9781467356664. doi: 10.1109/SPMB.2012.6469464.

[238] Yushan Zheng, Zhiguo Jiang, Fengying Xie, Haopeng Zhang, Yibing Ma, Huaqiang Shi, and Yu Zhao. Feature extraction from histopathological images based on nucleus-guided convolutional neural network for breast lesion classification. *Pattern Recognition*, 71:14–25, nov 2017. ISSN 00313203. doi: 10.1016/j.patcog.2017.05.010. URL https://linkinghub.elsevier.com/retrieve/pii/S0031320317302005.

[239] John D. Bukowy, Alex Dayton, Dustin Cloutier, Anna D. Manis, Alexander Staruschenko, Julian H. Lombard, Leah C. Solberg Woods, Daniel A. Beard, and Allen W. Cowley. Region-Based Convolutional Neural Nets for Localization of Glomeruli in Trichrome-Stained Whole Kidney Sections. *Journal of the American Society of Nephrology*, 2018. ISSN 1046-6673. doi: 10.1681/asn.2017111210.

[240] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[241] Tony Lindeberg. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. *International Journal of Computer Vision*, 11(3):283–318, 1993.

[242] Meyke Hermsen, Thomas de Bel, Marjolijn Den Boer, Eric J Steenbergen, Jesper Kers, Sandrine Florquin, Joris JTH Roelofs, Mark D Stegall, Mariam P Alexander, Byron H Smith, et al. Deep learning–based histopathologic assessment of kidney tissue. *Journal of the American Society of Nephrology*, 30(10):1968–1979, 2019.

[243] J Ferlay, M Colombet, I Soerjomataram, C Mathers, DM Parkin, M Piñeros, A Znaor, and F Bray. Estimating the global cancer incidence and mortality in 2018: Globocan sources and methods. *International journal of cancer*, 144(8):1941–1953, 2019.

[244] Qiuchang Sun, Xiaona Lin, Yuanshen Zhao, Ling Li, Kai Yan, Dong Liang, Desheng Sun, and Zhi-Cheng Li. Deep learning vs. radiomics for predicting axillary lymph node metastasis of breast cancer using ultrasound images: Don't forget the peritumoral region. *Frontiers in oncology*, 10:53, 2020. doi: 10.3389/fonc.2020.00053.

[245] Xin Li, Genggeng Qin, Qiang He, Lei Sun, Hui Zeng, Zilong He, Weiguo Chen, Xin Zhen, and Linghong Zhou. Digital breast tomosynthesis versus digital mammography: integration of image modalities enhances deep learning-based breast mass classification. *European radiology*, 30(2):778–788, 2020. doi: 10.1007/s00330-019-06457-5.

[246] Fung Fung Ting, Yen Jun Tan, and Kok Swee Sim. Convolutional neural network improvement for breast cancer classification. *Expert Systems with Applications*, 120: 103–115, 2019. doi: 10.1016/j.eswa.2018.11.008.

[247] Vitoantonio Bevilacqua, Antonio Brunetti, Andrea Guerriero, Gianpaolo Francesco Trotta, Michele Telegrafo, and Marco Moschetta. A performance comparison between shallow and deeper neural networks supervised classification of tomosynthesis breast lesions images. *Cognitive Systems Research*, 53:3–19, 2019. doi: 10.1016/j.cogsys.2018.04.011.

[248] Alejandro Forner, Josep M Llovet, and Jordi Bruix. Hepatocellular carcinoma. *The Lancet*, 379(9822):1245 – 1255, 2012. ISSN 0140-6736. doi: 10.1016/S0140-6736(11)61347-0.

[249] K Yasaka, H Akai, O Abe, and S Kiryu. Deep learning with cnn showed high diagnostic perfor-mance in differentiation of liver masses at dynamic ct. *Radiology*, 286:887–896, 2018.

[250] Patrick Ferdinand Christ, Mohamed Ezzeldin A Elshaer, Florian Ettlinger, Sunil Tatavarty, Marc Bickel, Patrick Bilic, Markus Rempfler, Marco Armbruster, Felix Hofmann, Melvin D'Anastasi, et al. Automatic liver and lesion segmentation in ct using cascaded fully convolutional neural networks and 3d conditional random fields. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 415–423. Springer, 2016.

[251] Koichiro Yasaka, Hiroyuki Akai, Akira Kunimatsu, Osamu Abe, and Shigeru Kiryu. Liver fibrosis: deep convolutional neural network for staging by using gadoxetic acid–enhanced hepatobiliary phase mr images. *Radiology*, 287(1):146–155, 2018.

[252] L Xiaomeng, C Hao, Q Xiaojuan, et al. H-denseunet: Hybrid densely connected unet for liver and tumor segmentation from ct volumes [j]. *IEEE Transactions on Medical Imaging*, pages 1–1, 2018.

[253] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[254] Jared J. Grantham. Autosomal dominant polycystic kidney disease. *New England Journal of Medicine*, 359(14):1477–1485, oct 2008. ISSN 0028-4793. doi: 10.1056/NEJMcp0804458.

[255] P. C. Harris, K. T. Bae, S. Rossetti, V. E. Torres, J. J. Grantham, A. B. Chapman, L. M. Guay-Woodford, B. F. King, L. H. Wetzel, D. A. Baumgarten, P. J. Kenney, M. Consugar, S. Klahr, W. M. Bennett, C. M. Meyers, Q. Zhang, P. A. Thompson, F. Zhu, and J. P. Miller. Cyst number but not the rate of cystic growth is associated with the mutated gene in autosomal dominant polycystic kidney disease. *Journal of the*

American Society of Nephrology, 17(11):3013–3019, sep 2006. ISSN 1046-6673. doi: 10.1681/ASN.2006080835.

[256] Vicente E. Torres, Arlene B. Chapman, Olivier Devuyst, Ron T. Gansevoort, Jared J. Grantham, Eiji Higashihara, Ronald D. Perrone, Holly B. Krasa, John Ouyang, and Frank S. Czerwiec. Tolvaptan in patients with autosomal dominant polycystic kidney disease. *New England Journal of Medicine*, 367(25):2407–2418, 2012. doi: 10.1056/NEJMoa1205511.

[257] Maria V. Irazabal, Vicente E. Torres, Marie C. Hogan, James Glockner, Bernard F. King, Troy G. Ofstie, Holly B. Krasa, John Ouyang, and Frank S. Czerwiec. Short-term effects of tolvaptan on renal function and volume in patients with autosomal dominant polycystic kidney disease. *Kidney International*, 80(3):295–301, aug 2011. ISSN 15231755. doi: 10.1038/ki.2011.119.

[258] Jean Nicolas Vauthey, Eddie K. Abdalla, Dorota A. Doherty, Philippe Gertsch, Marc J. Fenstermacher, Evelyne M. Loyer, Jan Lerut, Roland Materne, Xuemei Wang, Arthur Encarnacion, Delise Herron, Christian Mathey, Giovanni Ferrari, Chuslip Charnsangavej, Kim Anh Do, and Alban Denys. Body surface area and body weight predict total liver volume in western adults. *Liver Transplantation*, 8(3):233–240, mar 2002. ISSN 15276465. doi: 10.1053/jlts.2002.31654.

[259] S. A. Emamian, M. B. Nielsen, J. F. Pedersen, and L. Ytte. Kidney dimensions at sonography: Correlation with age, sex, and habitus in 665 adult volunteers. *American Journal of Roentgenology*, 160(1):83–86, jan 1993. ISSN 0361803X. doi: 10.2214/ajr.160.1.8416654.

[260] Kyongtae T Bae, Paul K Commean, and Jeongrim Lee. Volumetric measurement of renal cysts and parenchyma using mri: Phantoms and patients with polycystic kidney disease. *Journal of Computer Assisted Tomography*, 24(4):614–619, 2000. ISSN 03638715. doi: 10.1097/00004728-200007000-00019.

[261] B F King, J E Reed, E J Bergstralh, P F Sheedy, and V E Torres. Quantification and longitudinal trends of kidney, renal cyst, and renal parenchyma volumes in autosomal dominant polycystic kidney disease. *Journal of the American Society of Nephrology : JASN*, 11(8):1505–1511, aug 2000. ISSN 1046-6673.

[262] Eiji Higashihara, Kikuo Nutahara, Takatsugu Okegawa, Mitsuhiro Tanbo, Hidehiko Hara, Isao Miyazaki, Kuninori Kobayasi, and Toshiaki Nitatori. Kidney volume estimations with ellipsoid equations by magnetic resonance imaging in autosomal dominant polycystic kidney disease. *Nephron*, 129(4):253–262, 2015. ISSN 22353186. doi: 10.1159/000381476.

[263] M. V. Irazabal, L. J. Rangel, E. J. Bergstralh, S. L. Osborn, A. J. Harmon, J. L. Sundsbak, K. T. Bae, A. B. Chapman, J. J. Grantham, M. Mrug, M. C. Hogan, Z. M. El-Zoghby, P. C. Harris, B. J. Erickson, B. F. King, and V. E. Torres. Imaging classification

of autosomal dominant polycystic kidney disease: A simple model for selecting patients for clinical trials. *Journal of the American Society of Nephrology*, 26(1):160–172, jan 2015. ISSN 1046-6673. doi: 10.1681/ASN.2013101138.

[264] Kyongtae T. Bae, Cheng Tao, Jinhong Wang, Diana Kaya, Zhiyuan Wu, Junu T. Bae, Arlene B. Chapman, Vicente E. Torres, Jared J. Grantham, Michal Mrug, William M. Bennett, Michael F. Flessner, and Doug P. Landsittel. Novel approach to estimate kidney and cyst volumes using mid-slice magnetic resonance images in polycystic kidney disease. *American Journal of Nephrology*, 38(4):333–341, 2013. ISSN 02508095. doi: 10.1159/000355375.

[265] Jared J. Grantham and Vicente E. Torres. The importance of total kidney volume in evaluating progression of polycystic kidney disease. *Nature Reviews Nephrology*, 12 (11):667–677, nov 2016. ISSN 1759507X. doi: 10.1038/nrneph.2016.135.

[266] Jared J. Grantham, Vicente E. Torres, Arlene B. Chapman, Lisa M. Guay-Woodford, Kyongtae T. Bae, Bernard F. King, Louis H. Wetzel, Deborah A. Baumgarten, Phillip J. Kenney, Peter C. Harris, Saulo Klahr, William M. Bennett, Gladys N. Hirschman, Catherine M. Meyers, Xiaoling Zhang, Fang Zhu, and John P. Miller. Volume progression in polycystic kidney disease. *New England Journal of Medicine*, 354(20): 2122–2130, may 2006. ISSN 0028-4793. doi: 10.1056/NEJMoa054341.

[267] M Biswas, V Kuppili, L Saba, DR Edla, HS Suri, E Cuadrado-Godia, JR Laird, RT Marinhoe, JM Sanches, A Nicolaides, et al. State-of-the-art review on deep learning in medical imaging. *Frontiers in bioscience (Landmark edition)*, 24:392–426, 2019.

[268] Zeynettin Akkus, Alfiia Galimzianova, Assaf Hoogi, Daniel L. Rubin, and Bradley J. Erickson. Deep learning for brain MRI segmentation: State of the art and future directions. *J. Digital Imaging*, 30(4):449–459, 2017. doi: 10.1007/s10278-017-9983-4.

[269] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553): 436–444, may 2015. ISSN 14764687. doi: 10.1038/nature14539.

[270] Riccardo Magistroni, Cristiana Corsi, Teresa Martí, and Roser Torra. A review of the imaging techniques for measuring kidney and cyst volume in establishing autosomal dominant polycystic kidney disease progression. *American journal of nephrology*, 48 (1):67–78, 2018.

[271] Lennox Hoyte, Wen Ye, Linda Brubaker, Julia R Fielding, Mark E Lockhart, Marta E Heilbrun, Morton B Brown, Simon K Warfield, and Pelvic Floor Disorders Network. Segmentations of mri images of the female pelvic floor: A study of inter-and intra-reader reliability. *Journal of Magnetic Resonance Imaging*, 33(3):684–691, 2011.

[272] O. Gambino, S. Vitabile, G. Lo Re, G. La Tona, S. Librizzi, R. Pirrone, E. Ardizzone, and M. Midiri. Automatic volumetric liver segmentation using texture based region growing. In *2010 International Conference on Complex, Intelligent and Software Intensive Systems*, pages 146–152, 2010. doi: 10.1109/CISIS.2010.118.

[273] P. Arjun, M. K. Monisha, A. Mullaiyarasi, and G. Kavitha. Analysis of the liver in ct images using an improved region growing technique. In *2015 International Conference on Industrial Instrumentation and Control (ICIC)*, pages 1561–1566, 2015. doi: 10.1109/IIC.2015.7150998.

[274] Xiaoqi Lu, Jianshuai Wu, Xiaoying Ren, Baohua Zhang, and Yinhui Li. The study and application of the improved region growing algorithm for liver segmentation. *Optik*, 125(9):2142–2147, 2014.

[275] A. Mostafa, M. Abd Elfattah, A. Fouad, A. E. Hassanien, H. Hefny, and Tai-Hoon Kim. Region growing segmentation with iterative k-means for ct liver images. In *2015 4th International Conference on Advanced Information Technology and Sensor Application (AITS)*, pages 88–91, 2015. doi: 10.1109/AITS.2015.31.

[276] S. Arica, T. S. Avşar, and G. Erbay. A plain segmentation algorithm utilizing region growing technique for automatic partitioning of computed tomography liver images. In *2018 Medical Technologies National Congress (TIPTEKNO)*, pages 1–4, 2018. doi: 10.1109/TIPTEKNO.2018.8597108.

[277] SS Kumar, RS Moni, and J Rajeesh. Automatic segmentation of liver and tumor for cad of liver. *Journal of advances in information technology*, 2(1):63–70, 2011.

[278] Z. Yan, W. Wang, H. Yu, and J. Huang. Based on pre-treatment and region growing segmentation method of liver. In *2010 3rd International Congress on Image and Signal Processing*, volume 3, pages 1338–1341, 2010. doi: 10.1109/CISP.2010.5648010.

[279] Junbin Huang, Wenhong Qu, Lingquan Meng, and Chenhui Wang. Based on statistical analysis and 3d region growing segmentation method of liver. In *2011 3rd International Conference on Advanced Computer Control*, pages 478–482, 2011. doi: 10.1109/ICACC.2011.6016458.

[280] B. Lakshmipriya, K. Jayanthi, B. Pottakkat, and G. Ramkumar. Liver segmentation using bidirectional region growing with edge enhancement in nsct domain. In *2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA)*, pages 1–5, 2018. doi: 10.1109/ICSCAN.2018.8541257.

[281] S. Rafiei, N. Karimi, B. Mirmahboub, K. Najarian, B. Felfeliyan, S. Samavi, and S. M. Reza Soroushmehr. Liver segmentation in abdominal ct images using probabilistic atlas and adaptive 3d region growing. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6310–6313, 2019. doi: 10.1109/EMBC.2019.8857835.

[282] Zheng Zhou, Zhang Xue-chang, Zheng Si-ming, Xu Hua-fei, and Shi Yue-ding. Semi-automatic liver segmentation in ct images through intensity separation and region growing. *Procedia computer science*, 131:220–225, 2018.

[283] Yufei Chen, Zhicheng Wang, Weidong Zhao, and Xiaochun Yang. Liver segmentation from ct images based on region growing method. In *2009 3rd International Conference on Bioinformatics and Biomedical Engineering*, pages 1–4. IEEE, 2009.

[284] Tarek Gaber, Aboul Ella Hassanien, Nashwa El-Bendary, and Nilanjan Dey. *The 1st international conference on advanced intelligent system and informatics (AISI2015), November 28-30, 2015, Beni Suef, Egypt*, volume 407. Springer, 2015.

[285] S. A. Elmorsy, M. A. Abdou, Y. F. Hassan, and A. Elsayed. K3. a region growing liver segmentation method with advanced morphological enhancement. In *2015 32nd National Radio Science Conference (NRSC)*, pages 418–425, 2015. doi: 10.1109/NRSC.2015.7117857.

[286] Vladimir Vezhnevets and Vadim Konouchine. Growcut: Interactive multi-label nd image segmentation by cellular automata. In *proc. of Graphicon*, volume 1, pages 150–156, 2005.

[287] V. Czipczer and A. Manno-Kovacs. Automatic liver segmentation on ct images combining region-based techniques and convolutional features. In *2019 International Conference on Content-Based Multimedia Indexing (CBMI)*, pages 1–6, 2019. doi: 10.1109/CBMI.2019.8877400.

[288] Vitoantonio Bevilacqua, Antonio Brunetti, Gianpaolo Francesco Trotta, Giovanni Dimauro, Katarina Elez, Vito Alberotanza, and Arnaldo Scardapane. A novel approach for hepatocellular carcinoma detection and classification based on triphasic CT protocol. In *2017 IEEE Congress on Evolutionary Computation, CEC 2017, Donostia, San Sebastián, Spain, June 5-8, 2017*, pages 1856–1863, 2017. doi: 10.1109/CEC.2017.7969527.

[289] Vitoantonio Bevilacqua, Leonarda Carnimeo, Antonio Brunetti, Andrea De Pace, Pietro Galeandro, Gianpaolo Francesco Trotta, Nicholas Caporusso, Francescomaria Marino, Vito Alberotanza, and Arnaldo Scardapane. Synthesis of a neural network classifier for hepatocellular carcinoma grading based on triphasic ct images. In K.C. Santosh, Mallikarjun Hangarge, Vitoantonio Bevilacqua, and Atul Negi, editors, *Recent Trends in Image Processing and Pattern Recognition*, pages 356–368, Singapore, 2017. Springer Singapore. ISBN 978-981-10-4859-3.

[290] Lei Xu, Yingliang Zhu, Yuhao Zhang, and Haima Yang. Liver segmentation based on region growing and level set active contour model with new signed pressure force function. *Optik*, 202:163705, 2020.

[291] Antonia Mihaylova and Veska Georgieva. Spleen segmentation in mri sequence images using template matching and active contours. *Procedia Computer Science*, 131: 15–22, 2018.

[292] Antonia Mihaylova, Veska Georgieva, and Plamen Petrov. Multistage approach for automatic spleen segmentation in mri sequences. *International Journal of Reasoning-based Intelligent Systems*, 12(2):128–137, 2020.

[293] A. Behrad and H. Masoumi. Automatic spleen segmentation in mri images using a combined neural network and recursive watershed transform. In *10th Symposium on Neural Network Applications in Electrical Engineering*, pages 63–67, 2010. doi: 10.1109/NEUREL.2010.5644110.

[294] H. Jiang, Z. Ma, B. Zhang, and Y. Zhang. A spleen segmentation method based on pca-iso. In *2011 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT)*, pages 928–933, 2011.

[295] R. Gauriau, R. Ardori, D. Lesage, and I. Bloch. Multiple template deformation application to abdominal organ segmentation. In *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, pages 359–362, 2015. doi: 10.1109/ISBI.2015.7163887.

[296] SM Reza Soroushmehr, Pavani Davuluri, Somayeh Molaei, Rosalyn Hobson Hargraves, Yang Tang, Charles H Cockrell, Kevin Ward, and Kayvan Najarian. Spleen segmentation and assessment in ct images for traumatic abdominal injuries. *Journal of medical systems*, 39(9):87, 2015.

[297] Vitoantonio Bevilacqua, Antonio Brunetti, Giacomo Donato Cascarano, Flavio Palmieri, Andrea Guerriero, and Marco Moschetta. A deep learning approach for the automatic detection and segmentation in autosomal dominant polycystic kidney disease based on magnetic resonance images. In De-Shuang Huang, Kang-Hyun Jo, and Xiaolong Zhang, editors, *Intelligent Computing Theories and Application - 14th International Conference, ICIC 2018, Wuhan, China, August 15-18, 2018, Proceedings, Part II*, volume 10955 of *Lecture Notes in Computer Science*, pages 643–649, Cham, 2018. Springer. ISBN 978-3-319-95932-0. doi: 10.1007/978-3-319-95933-7_73.

[298] Vitoantonio Bevilacqua, Antonio Brunetti, Giacomo Donato Cascarano, Andrea Guerriero, Francesco Pesce, Marco Moschetta, and Loreto Gesualdo. A comparison between two semantic deep learning frameworks for the autosomal dominant polycystic kidney disease segmentation based on magnetic resonance images. *BMC Medical Informatics and Decision Making*, 19(9):1–12, 2019.

[299] Bob D de Vos, Jelmer M Wolterink, Pim A de Jong, Tim Leiner, Max A Viergever, and Ivana Išgum. Convnet-based localization of anatomical structures in 3-d medical images. *IEEE transactions on medical imaging*, 36(7):1470–1481, 2017.

[300] Xuesong Lu, Qinlan Xie, Yunfei Zha, and Defeng Wang. Fully automatic liver segmentation combining multi-dimensional graph cut with shape information in 3d ct images. *Scientific reports*, 8(1):1–9, 2018.

[301] S. Rafiei, E. Nasr-Esfahani, K. Najarian, N. Karimi, S. Samavi, and S. M. R. Soroushmehr. Liver segmentation in ct images using three dimensional to two dimensional

fully convolutional network. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2067–2071, 2018. doi: 10.1109/ICIP.2018.8451238.

[302] Hojin Kim, Jinhong Jung, Jieun Kim, Byungchul Cho, Jungwon Kwak, Jeong Yun Jang, Sang-wook Lee, June-Goo Lee, and Sang Min Yoon. Abdominal multi-organ auto-segmentation using 3d-patch-based deep convolutional neural network. *Scientific Reports*, 10(1):1–9, 2020.

[303] C Couinaud. Liver lobes and segments: notes on the anatomical architecture and surgery of the liver. *La Presse médicale*, 62(33):709, 1954.

[304] Thomas S Helling and Benoit Blondeau. Anatomic segmental resection compared to major hepatectomy in the treatment of liver neoplasms. *HPB*, 7(3):222–225, 2005.

[305] Dário AB Oliveira, Raul Q Feitosa, and Mauro M Correia. Segmentation of liver, its vessels and lesions from ct images for surgical planning. *Biomedical engineering online*, 10(1):1–23, 2011.

[306] Terry S Yoo, Michael J Ackerman, William E Lorensen, Will Schroeder, Vikram Chalana, Stephen Aylward, Dimitris Metaxas, and Ross Whitaker. Engineering and algorithm design for an image processing api: a technical report on itk-the insight toolkit. *Studies in health technology and informatics*, pages 586–592, 2002.

[307] Xiaopeng Yang, Jae Do Yang, Hong Pil Hwang, Hee Chul Yu, Sungwoo Ahn, Bong-Wan Kim, and Heecheon You. Segmentation of liver and vessels from ct images and classification of liver segments for preoperative liver surgical planning in living donor liver transplantation. *Computer methods and programs in biomedicine*, 158:41–52, 2018.

[308] Evgin Goceri, Zarine K Shah, and Metin N Gurcan. Vessel segmentation from abdominal magnetic resonance images: adaptive and reconstructive approach. *International journal for numerical methods in biomedical engineering*, 33(4):e2811, 2017.

[309] Y. Chi, J. Liu, S. K. Venkatesh, S. Huang, J. Zhou, Q. Tian, and W. L. Nowinski. Segmentation of liver vasculature from contrast enhanced ct images using context-based voting. *IEEE Transactions on Biomedical Engineering*, 58(8):2144–2153, 2011. doi: 10.1109/TBME.2010.2093523.

[310] Ye-zhan Zeng, Yu-qian Zhao, Ping Tang, Miao Liao, Yi-xiong Liang, Sheng-hui Liao, and Bei-ji Zou. Liver vessel segmentation and identification based on oriented flux symmetry and graph cuts. *Computer methods and programs in biomedicine*, 150:31–39, 2017.

[311] Roberto Merletti and Dario Farina. *Surface Electromyography: Physiology, Engineering and Applications*. 2016. ISBN 9781119082934. doi: 10.1002/9781119082934.

[312] Dario Farina and Deborah Falla. Effect of muscle-fiber velocity recovery function on motor unit action potential properties in voluntary contractions. *Muscle and Nerve*, 2008. ISSN 0148639X. doi: 10.1002/mus.20948.

[313] Peppoloni L., Filippeschi A., Ruffaldi E., and Avizzano C.A. (WMSDs issue) A novel wearable system for the online assessment of risk for biomechanical load in repetitive efforts. *International Journal of Industrial Ergonomics*, 2016. ISSN 1872-8219. doi: 10.1016/j.ergon.2015.07.002.

[314] Maura Casadio, Pietro G. Morasso, and Vittorio Sanguineti. Direct measurement of ankle stiffness during quiet standing: Implications for control modelling and clinical application. *Gait and Posture*, 2005. ISSN 09666362. doi: 10.1016/j.gaitpost.2004.05.005.

[315] Vito Monaco, Alessio Ghionzoli, and Silvestro Micera. Age-Related Modifications of Muscle Synergies and Spinal Cord Activity During Locomotion. *Journal of Neurophysiology*, 2010. ISSN 0022-3077. doi: 10.1152/jn.00525.2009.

[316] J. R. Cram. Biofeedback Applications. In *Electromyography*. 2005. doi: 10.1002/0471678384.ch17.

[317] T. F. Besier, D. G. Lloyd, T. R. Ackland, and J. L. Cochrane. Anticipatory effects on knee joint loading during running and cutting maneuvers. *Medicine and Science in Sports and Exercise*, 2001. ISSN 01959131. doi: 10.1097/00005768-200107000-00015.

[318] Domenico Buongiorno, Michele Barsotti, Francesco Barone, Vitoantonio Bevilacqua, and Antonio Frisoli. A linear approach to optimize an EMG-driven neuromusculoskeletal model for movement intention detection in myo-control: A case study on shoulder and elbow joints. *Frontiers in Neurorobotics*, 2018. ISSN 16625218. doi: 10.3389/fnbot.2018.00074.

[319] M Atzori, M Cognolato, and H Müller. Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Frontiers in Neurorobotics*, 10(SEP), 2016. ISSN 16625218. doi: 10.3389/fnbot.2016.00009.

[320] Ivan Vujaklija, Vahid Shalchyan, Ernest N. Kamavuako, Ning Jiang, Hamid R. Marateb, and Dario Farina. Online mapping of emg signals into kinematics by autoencoding. *Journal of NeuroEngineering and Rehabilitation*, 15(1):21, Mar 2018. ISSN 1743-0003. doi: 10.1186/s12984-018-0363-1. URL https://doi.org/10.1186/s12984-018-0363-1.

[321] Roberto Merletti and Dario Farina. *Surface electromyography: physiology, engineering, and applications*. John Wiley & Sons, 2016.

[322] Dario Farina, Corrado Cescon, and Roberto Merletti. Influence of anatomical, physical, and detection-system parameters on surface emg. *Biological cybernetics*, 86(6):445–456, 2002.

[323] Weiqiang Li and Kazuyoshi Sakamoto. The influence of location of electrode on muscle fiber conduction velocity and emg power spectrum during voluntary isometric contraction measured with surface array electrodes. *Applied Human Science*, 15(1): 25–32, 1996.

[324] L Mesin, R Merletti, and Alberto Rainoldi. Surface emg: the issue of electrode location. *Journal of Electromyography and Kinesiology*, 19(5):719–726, 2009.

[325] Alberto Rainoldi, M Nazzaro, R Merletti, D Farina, I Caruso, and S Gaudenti. Geometrical factors in surface emg of the vastus medialis and lateralis muscles. *Journal of Electromyography and Kinesiology*, 10(5):327–336, 2000.

[326] Bart Freriks and Hermie Hermens. *European recommendations for surface electromyography: results of the SENIAM project*. Roessingh Research and Development, 2000.

[327] B Lv, X Sheng, and X Zhu. Improving Myoelectric Pattern Recognition Robustness to Electrode Shift by Autoencoder. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, volume 2018-July, pages 5652–5655. Institute of Electrical and Electronics Engineers Inc., 2018. ISBN 9781538636466. doi: 10.1109/EMBC.2018.8513525.

[328] Claudio Castellini and Patrick van der Smagt. Surface emg in advanced hand prosthetics. *Biological cybernetics*, 100(1):35–47, 2009.

[329] G. Kanitz, C. Cipriani, and B. B. Edin. Classification of transient myoelectric signals for the control of multi-grasp hand prostheses. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(9):1756–1764, Sep. 2018. ISSN 1558-0210. doi: 10.1109/TNSRE.2018.2861465.

[330] D. Farina, N. Jiang, H. Rehbaum, A. Holobar, B. Graimann, H. Dietl, and O. C. Aszmann. The extraction of neural information from the surface emg for the control of upper-limb prostheses: Emerging avenues and challenges. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(4):797–809, July 2014. ISSN 1534-4320. doi: 10.1109/TNSRE.2014.2305111.

[331] N. Jiang *, K. B. Englehart, and P. A. Parker. Extracting simultaneous and proportional neural control information for multiple-dof prostheses from the surface electromyographic signal. *IEEE Transactions on Biomedical Engineering*, 56(4):1070–1080, 2009.

[332] D. Buongiorno, C. Camardella, G. D. Cascarano, L. Pelaez Murciego, M. Barsotti, I. De Feudis, A. Frisoli, and V. Bevilacqua. An undercomplete autoencoder to extract muscle synergies for motor intention detection. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, July 2019. doi: 10.1109/IJCNN.2019.8851975.

[333] D. Buongiorno, M. Barsotti, E. Sotgiu, C. Loconsole, M. Solazzi, V. Bevilacqua, and A. Frisoli. A neuromusculoskeletal model of the human upper limb for a myoelectric exoskeleton control using a reduced number of muscles. In *2015 IEEE World Haptics Conference (WHC)*, pages 273–279, June 2015. doi: 10.1109/WHC.2015.7177725.

[334] Domenico Buongiorno, Francesco Barone, Massimiliano Solazzi, Vitoantonio Bevilacqua, and Antonio Frisoli. A linear optimization procedure for an emg-driven neuromusculoskeletal model parameters adjusting: Validation through a myoelectric exoskeleton control. In Fernando Bello, Hiroyuki Kajimoto, and Yon Visell, editors, *Haptics: Perception, Devices, Control, and Applications*, pages 218–227, Cham, 2016. Springer International Publishing. ISBN 978-3-319-42324-1.

[335] Domenico Buongiorno, Francesco Barone, Denise J. Berger, Benedetta Cesqui, Vitoantonio Bevilacqua, Andrea d'Avella, and Antonio Frisoli. Evaluation of a pose-shared synergy-based isometric model for hand force estimation: Towards myocontrol. In Jaime Ibáñez, José González-Vargas, José María Azorín, Metin Akay, and José Luis Pons, editors, *Converging Clinical and Engineering Research on Neurorehabilitation II*, pages 953–958, Cham, 2017. Springer International Publishing. ISBN 978-3-319-46669-9. doi: 10.1007/978-3-319-46669-9_154.

[336] Cristian Camardella, Michele Barsotti, Luis Pelaez Murciego, Domenico Buongiorno, Vitoantonio Bevilacqua, and Antonio Frisoli. Evaluating generalization capability of bio-inspired models for a myoelectric control: A pilot study. In De-Shuang Huang, Zhi-Kai Huang, and Abir Hussain, editors, *Intelligent Computing Methodologies*, pages 739–750, Cham, 2019. Springer International Publishing. ISBN 978-3-030-26766-7.

[337] Domenico Buongiorno, Giacomo Donato Cascarano, Cristian Camardella, Irio De Feudis, Antonio Frisoli, and Vitoantonio Bevilacqua. Task-oriented muscle synergy extraction using an autoencoder-based neural model. *Information*, 11(4):219, 2020. doi: 10.3390/info11040219.

[338] Purushothaman Geethanjali. Myoelectric control of prosthetic hands: State-of-the-art review, 2016. ISSN 11791470.

[339] Jamileh Yousefi and Andrew Hamilton-Wright. Characterizing emg data using machine-learning tools. *Computers in Biology and Medicine*, 51:1 – 13, 2014. ISSN 0010-4825. doi: https://doi.org/10.1016/j.compbiomed.2014.04.018.

[340] Ercan Gokgoz and Abdulhamit Subasi. Effect of multiscale pca de-noising on emg signal classification for diagnosis of neuromuscular disorders. *Journal of Medical Systems*, 38(4):31, Apr 2014. ISSN 1573-689X. doi: 10.1007/s10916-014-0031-3.

[341] Abdulhamit Subasi. Classification of emg signals using pso optimized svm for diagnosis of neuromuscular disorders. *Computers in Biology and Medicine*, 43(5):576 – 586, 2013. ISSN 0010-4825. doi: https://doi.org/10.1016/j.compbiomed.2013.01.020.

[342] S. S. Nair, R. M. French, D. Laroche, and E. Thomas. The application of machine learning algorithms to the analysis of electromyographic patterns from arthritic patients. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):174–184, April 2010. ISSN 1558-0210. doi: 10.1109/TNSRE.2009.2032638.

[343] Nihal Fatma Güler and Sabri Koçer. Classification of emg signals using pca and fft. *Journal of Medical Systems*, 29(3):241–250, Jun 2005. ISSN 1573-689X. doi: 10.1007/s10916-005-5184-7.

[344] Bekir Karlik. Machine learning algorithms for characterization of emg signals. *International Journal of Information and Electronics Engineering*, 4(3):189, 2014.

[345] Domenico Buongiorno, Gianpaolo Francesco Trotta, Ilaria Bortone, Nicola Di Gioia, Felice Avitto, Giacomo Losavio, and Vitoantonio Bevilacqua. Assessment and rating of movement impairment in parkinson's disease using a low-cost vision-based system. In De-Shuang Huang, M. Michael Gromiha, Kyungsook Han, and Abir Hussain, editors, *Intelligent Computing Methodologies*, pages 777–788, Cham, 2018. Springer International Publishing. ISBN 978-3-319-95957-3.

[346] Leonarda Carnimeo, Gianpaolo Francesco Trotta, Antonio Brunetti, Giacomo Donato Cascarano, Domenico Buongiorno, Claudio Loconsole, Eugenio Di Sciascio, and Vitoantonio Bevilacqua. Proposal of a health care network based on big data analytics for pds. *The Journal of Engineering*, March 2019. doi: 10.1049/joe.2018.5141.

[347] Giacomo Donato Cascarano, Antonio Brunetti, Domenico Buongiorno, Gianpaolo Francesco Trotta, Claudio Loconsole, Ilaria Bortone, and Vitoantonio Bevilacqua. *A Multi-modal Tool Suite for Parkinson's Disease Evaluation and Grading*, pages 257–268. Springer Singapore, Singapore, 2020. ISBN 978-981-13-8950-4. doi: 10.1007/978-981-13-8950-4_24.

[348] De-Shuang Huang, Xing-Ming Zhao, Guang-Bin Huang, and Yiu-Ming Cheung. Classifying protein sequences using hydropathy blocks. *Pattern recognition*, 39(12): 2293–2300, 2006.

[349] Ji-Xiang Du, De-Shuang Huang, Xiao-Feng Wang, and Xiao Gu. Shape recognition based on neural networks trained by differential evolution algorithm. *Neurocomputing*, 70(4-6):896–903, 2007.

[350] Kun-Hong Liu and De-Shuang Huang. Cancer classification using rotation forest. *Computers in biology and medicine*, 38(5):601–610, 2008.

[351] Zhong-Qiu Zhao and De-Shuang Huang. A mended hybrid learning algorithm for radial basis function neural networks to improve generalization capability. *Applied Mathematical Modelling*, 31(7):1271–1281, 2007.

[352] Fei Han, Qing-Hua Ling, and De-Shuang Huang. An improved approximation approach incorporating particle swarm optimization and a priori information into neural networks. *Neural Computing and Applications*, 19(2):255–261, 2010.

[353] Vitoantonio Bevilacqua, Antonio Brunetti, Maurizio Triggiani, Domenico Magaletti, Michele Telegrafo, and Marco Moschetta. An optimized feed-forward artificial neural network topology to support radiologists in breast lesions classification. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*, GECCO '16 Companion, pages 1385–1392, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450343237. doi: 10.1145/2908961.2931733. URL https://doi.org/10.1145/2908961.2931733.

[354] Nicholas Caporusso, Luigi Biasi, Giovanni Cinquepalmi, Gianpaolo Francesco Trotta, Antonio Brunetti, and Vitoantonio Bevilacqua. A wearable device supporting multiple touch-and gesture-based languages for the deaf-blind. In *International Conference on Applied Human Factors and Ergonomics*, pages 32–41. Springer, 2017.

[355] Ji-Xiang Du, De-Shuang Huang, Guo-Jun Zhang, and Zeng-Fu Wang. A novel full structure optimization algorithm for radial basis probabilistic neural networks. *Neurocomputing*, 70(1-3):592–596, 2006.

[356] Nianyin Zeng, Han Li, Zidong Wang, Weibo Liu, Songming Liu, Fuad E Alsaadi, and Xiaohui Liu. Deep-reinforcement-learning-based images segmentation for quantitative analysis of gold immunochromatographic strip. *Neurocomputing*, 2020.

[357] B Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., ... Kingsbury. Deep Neural Networks for Acoustic Modeling in Speech Recognition. *IEEE Signal Processing Magazine*, 2012.

[358] Di Wu, Si-Jia Zheng, Wen-Zheng Bao, Xiao-Ping Zhang, Chang-An Yuan, and De-Shuang Huang. A novel deep model with multi-loss and efficient training for person re-identification. *Neurocomputing*, 324:69–75, 2019.

[359] Xinhua Liu, Yao Zou, Chengjuan Xie, Hailan Kuang, and Xiaolin Ma. Bidirectional face aging synthesis based on improved deep convolutional generative adversarial networks. *Information*, 10(2):69, 2019.

[360] Xinhua Liu, Gaoqiang Hu, Xiaolin Ma, and Hailan Kuang. An enhanced neural network based on deep metric learning for skin lesion segmentation. In *2019 Chinese Control And Decision Conference (CCDC)*, pages 1633–1638. IEEE, 2019.

[361] Nianyin Zeng, Zidong Wang, Bachar Zineddin, Yurong Li, Min Du, Liang Xiao, Xiaohui Liu, and Terry Young. Image-based quantitative analysis of gold immunochromatographic strip via cellular neural network approach. *IEEE transactions on medical imaging*, 33(5):1129–1136, 2014.

[362] Antonio Brunetti, Domenico Buongiorno, Gianpaolo Francesco Trotta, and Vitoantonio Bevilacqua. Computer vision and deep learning techniques for pedestrian detection and tracking: A survey. *Neurocomputing*, 300:17–33, jul 2018. ISSN 18728286. doi: 10.1016/j.neucom.2018.01.092.

[363] Vitoantonio Bevilacqua, Antonio Brunetti, Gianpaolo Francesco Trotta, Domenico De Marco, Marco Giuseppe Quercia, Domenico Buongiorno, Alessia D'Introno, Francesco Girardi, and Attilio Guarini. A novel deep learning approach in haematology for classification of leucocytes. In *Italian Workshop on Neural Nets*, pages 265–274. Springer, 2017.

[364] Antonio Brunetti, Giacomo Donato Cascarano, Irio De Feudis, Marco Moschetta, Loreto Gesualdo, and Vitoantonio Bevilacqua. Detection and segmentation of kidneys from magnetic resonance images in patients with autosomal dominant polycystic kidney disease. In *International Conference on Intelligent Computing*, pages 639–650. Springer, 2019.

[365] Wenxuan Xu, Lin Zhu, and De-Shuang Huang. Dcde: An efficient deep convolutional divergence encoding method for human promoter recognition. *IEEE transactions on nanobioscience*, 18(2):136–145, 2019.

[366] Di Wu, Si-Jia Zheng, Fei Cheng, Yang Zhao, Chang-An Yuan, Xiao Qin, Yong-Li Jiang, and De-Shuang Huang. A hybrid deep model for person re-identification. In *International Conference on Intelligent Computing*, pages 229–234. Springer, 2018.

[367] Si-Jia Zheng, Di Wu, Fei Cheng, Yang Zhao, Chang-An Yuan, Xiao Qin, and De-Shuang Huang. A simple and effective deep model for person re-identification. In *International Conference on Intelligent Computing*, pages 223–228. Springer, 2018.

[368] Xinhua Liu, Chengjuan Xie, Hailan Kuang, and Xiaolin Ma. Face aging simulation with deep convolutional generative adversarial networks. In *2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pages 220–224. IEEE, 2018.

[369] Giovanni Dimauro, Giorgio Ciprandi, Francesca Deperte, Francesco Girardi, Enrico Ladisa, Sergio Latrofa, and Matteo Gelardi. Nasal cytology with deep learning techniques. *International Journal of Medical Informatics*, 122:13 – 19, 2019. ISSN 1386-5056. doi: https://doi.org/10.1016/j.ijmedinf.2018.11.010.

[370] Vitoantonio Bevilacqua, Antonio Brunetti, Gianpaolo Francesco Trotta, Leonarda Carnimeo, Francescomaria Marino, Vito Alberotanza, and Arnaldo Scardapane. A deep learning approach for hepatocellular carcinoma grading. *International Journal of Computer Vision and Image Processing (IJCVIP)*, 7(2):1–18, 2017.

[371] Manfredo Atzori, Arjan Gijsberts, Ilja Kuzborskij, Simone Elsig, Anne Gabrielle Mittaz Hager, Olivier Deriaz, Claudio Castellini, Henning Müller, and Barbara Caputo. Characterization of a benchmark database for myoelectric movement classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2015. ISSN 15344320. doi: 10.1109/TNSRE.2014.2328495.

[372] Max Ortiz-Catalan, Rickard Brånemark, and Bo Håkansson. Biopatrec: A modular research platform for the control of artificial limbs based on pattern recognition algorithms. *Source code for biology and medicine*, 8(1):11, 2013. doi: 10.1186/1751-0473-8-11.

[373] Weidong Geng, Yu Du, Wenguang Jin, Wentao Wei, Yu Hu, and Jiajun Li. Gesture recognition by instantaneous surface EMG images. *Scientific Reports*, 2016. ISSN 20452322. doi: 10.1038/srep36571.

[374] Christos Sapsanis, George Georgoulas, Anthony Tzes, and Dimitrios Lymberopoulos. Improving EMG based classification of basic hand movements using EMD. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2013. ISBN 9781457702167. doi: 10.1109/EMBC.2013.6610858.

[375] Christoph Amma, Thomas Krings, Jonas Böer, and Tanja Schultz. Advancing muscle-computer interfaces with high-density electromyography. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 929–938, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450331456. doi: 10.1145/2702123.2702501. URL https://doi.org/10.1145/2702123.2702501.

[376] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000.

[377] Christian O'Reilly, Nadia Gosselin, Julie Carrier, and Tore Nielsen. Montreal archive of sleep studies: an open-access resource for instrument benchmarking and exploratory research. *Journal of sleep research*, 23(6):628–635, 2014.

[378] N Ganapathy, R Swaminathan, and T M Deserno. Deep Learning on 1-D Biosignals: a Taxonomy-based Survey. *Yearbook of medical informatics*, 27(1):98–109, 2018. ISSN 23640502. doi: 10.1055/s-0038-1667083.

[379] O Faust, Y Hagiwara, T J Hong, O S Lih, and U R Acharya. Deep learning for healthcare applications based on physiological signals: A review. *Computer Methods and Programs in Biomedicine*, 161:1–13, 2018. ISSN 01692607. doi: 10.1016/j.cmpb.2018.04.005.

[380] Carlo J De Luca, L Donald Gilmore, Mikhail Kuznetsov, and Serge H Roy. Filtering the surface emg signal: Movement artifact and baseline noise contamination. *Journal of biomechanics*, 43(8):1573–1579, 2010.

[381] Ouriel Barzilay and Alon Wolf. A fast implementation for emg signal linear envelope computation. *Journal of Electromyography and Kinesiology*, 21(4):678–682, 2011.

[382] Rodrigo Lício Ortolan, Ricardo Naoki Mori, Roberto R Pereira, Cristina MN Cabral, José Carlos Pereira, and Alberto Cliquet. Evaluation of adaptive/nonadaptive filtering and wavelet transform techniques for noise reduction in emg mobile acquisition equipment. *IEEE transactions on neural systems and rehabilitation engineering*, 11(1): 60–69, 2003.

[383] David P Allen. A frequency domain hampel filter for blind rejection of sinusoidal interference from electromyograms. *Journal of neuroscience methods*, 177(2):303–310, 2009.

[384] B. Hudgins, P. Parker, and R. N. Scott. A new strategy for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 40(1):82–94, 1993.

[385] U Côté-Allard, C L Fall, A Campeau-Lecoursy, C Gosseliny, F Laviolettez, and B Gosselin. Transfer learning for sEMG hand gestures recognition using convolutional neural networks. In *2017 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2017*, volume 2017-Janua, pages 1663–1668. Institute of Electrical and Electronics Engineers Inc., 2017. ISBN 9781538616451. doi: 10.1109/SMC.2017.8122854.

[386] Y Du, W Jin, W Wei, Y Hu, and W Geng. Surface EMG-based inter-session gesture recognition enhanced by deep domain adaptation. *Sensors (Switzerland)*, 17(3), 2017. ISSN 14248220. doi: 10.3390/s17030458.

[387] Mark Ison and Panagiotis Artemiadis. The role of muscle synergies in myoelectric control: trends and challenges for simultaneous multifunction control. *Journal of neural engineering*, 11(5):051001, 2014.

[388] Manfredo Atzori, Arjan Gijsberts, Claudio Castellini, Barbara Caputo, Anne-Gabrielle Mittaz Hager, Simone Elsig, Giorgio Giatsidis, Franco Bassetto, and Henning Müller. Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific data*, 1:140053, 2014. doi: 10.1038/sdata.2014.53.

[389] W Wei, Y Wong, Y Du, Y Hu, M Kankanhalli, and W Geng. A multi-stream convolutional neural network for sEMG-based gesture recognition in muscle-computer interface. *Pattern Recognition Letters*, 119:131–138, 2019. ISSN 01678655. doi: 10.1016/j.patrec.2017.12.005.

[390] Joel Stein, Kailas Narendran, John McBean, Kathryn Krebs, and Richard Hughes. Electromyography-controlled exoskeletal upper-limb–powered orthosis for exercise training after stroke. *American journal of physical medicine & rehabilitation*, 86(4): 255–261, 2007.

[391] Ho Shing Lo and Sheng Quan Xie. Exoskeleton robots for upper-limb rehabilitation: State of the art and future prospects. *Medical Engineering & Physics*, 34(3):261 – 268, 2012. ISSN 1350-4533. doi: https://doi.org/10.1016/j.medengphy.2011.10.004. URL http://www.sciencedirect.com/science/article/pii/S1350453311002694.

[392] D. Leonardis, M. Barsotti, C. Loconsole, M. Solazzi, M. Troncossi, C. Mazzotti, V. P. Castelli, C. Procopio, G. Lamola, C. Chisari, M. Bergamasco, and A. Frisoli. An emg-controlled robotic hand exoskeleton for bilateral rehabilitation. *IEEE Transactions on Haptics*, 8(2):140–151, April 2015. ISSN 1939-1412. doi: 10.1109/TOH.2015.2417570.

[393] Aidan D Roche, Hubertus Rehbaum, Dario Farina, and Oskar C Aszmann. Prosthetic myoelectric control strategies: a clinical perspective. *Current Surgery Reports*, 2(3):44, 2014.

[394] Vito Papapicco, Andrea Parri, Elena Martini, Vitoantonio Bevilacqua, Simona Crea, and Nicola Vitiello. Locomotion mode classification based on support vector machines and hip joint angles: A feasibility study for applications in wearable robotics. In Fanny Ficuciello, Fabio Ruggiero, and Alberto Finzi, editors, *Human Friendly Robotics*, pages 197–205, Cham, 2019. Springer International Publishing. ISBN 978-3-319-89327-3.

[395] Panagiotis K Artemiadis and Kostas J Kyriakopoulos. Emg-based teleoperation of a robot arm in planar catching movements using armax model and trajectory monitoring techniques. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 3244–3249. IEEE, 2006.

[396] Panagiotis K Artemiadis and Kostas J Kyriakopoulos. An emg-based robot control scheme robust to time-varying emg signal features. *IEEE Transactions on Information Technology in Biomedicine*, 14(3):582–588, 2010.

[397] Daniel Graupe and William K Cline. Functional separation of emg signals via arma identification methods for prosthesis control purposes. *IEEE Transactions on Systems, Man, and Cybernetics*, (2):252–259, 1975.

[398] Thomas S Buchanan, David G Lloyd, Kurt Manal, and Thor F Besier. Neuromusculoskeletal modeling: estimation of muscle forces and joint moments and movements from measurements of neural command. *Journal of applied biomechanics*, 20(4):367, 2004.

[399] David G Lloyd and Thor F Besier. An emg-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo. *Journal of biomechanics*, 36(6):765–776, 2003.

[400] Shaowei Yao, Yu Zhuang, Zhijun Li, and Rong Song. Adaptive admittance control for an ankle exoskeleton using an emg-driven musculoskeletal model. *Frontiers in Neurorobotics*, 12:16, 2018. ISSN 1662-5218. doi: 10.3389/fnbot.2018.00016. URL https://www.frontiersin.org/article/10.3389/fnbot.2018.00016.

[401] Domenico Buongiorno, Michele Barsotti, Francesco Barone, Vitoantonio Bevilacqua, and Antonio Frisoli. A linear approach to optimize an emg-driven neuromusculoskeletal model for movement intention detection in myo-control: A case study on shoulder and

elbow joints. *Frontiers in Neurorobotics*, 12:74, 2018. ISSN 1662-5218. doi: 10.3389/fnbot.2018.00074. URL https://www.frontiersin.org/article/10.3389/fnbot.2018.00074.

[402] Denise J. Berger and Andrea d'Avella. Effective force control by muscle synergies. *Frontiers in Computational Neuroscience*, 8:46, 2014. ISSN 1662-5188. doi: 10.3389/fncom.2014.00046. URL https://www.frontiersin.org/article/10.3389/fncom.2014.00046.

[403] Arjan Gijsberts, Rashida Bohra, David Sierra González, Alexander Werner, Markus Nowak, Barbara Caputo, Maximo Alejandro Roa, and Claudio Castellini. Stable myoelectric control of a hand prosthesis using non-linear incremental learning. *Frontiers in neurorobotics*, 8:8, 2014.

[404] Denise J Berger, Reinhard Gentner, Timothy Edmunds, Dinesh K Pai, and Andrea d'Avella. Differences in adaptation rates after virtual surgeries provide direct evidence for modularity. *Journal of Neuroscience*, 33(30):12384–12394, 2013.

[405] Emilio Bizzi and Vincent CK Cheung. The neural origin of muscle synergies. *Frontiers in computational neuroscience*, 7:51, 2013.

[406] Jason J Kutch and Francisco J Valero-Cuevas. Challenges and new approaches to proving the existence of muscle synergies of neural origin. *PLoS computational biology*, 8(5), 2012.

[407] Cristiano Alessandro, Ioannis Delis, Francesco Nori, Stefano Panzeri, and Bastien Berret. Muscle synergies in neuroscience and robotics: from input-space to task-space perspectives. *Frontiers in Computational Neuroscience*, 7:43, 2013. ISSN 1662-5188. doi: 10.3389/fncom.2013.00043. URL https://www.frontiersin.org/article/10.3389/fncom.2013.00043.

[408] Vincent CK Cheung, Andrea Turolla, Michela Agostini, Stefano Silvoni, Caoimhe Bennis, Patrick Kasi, Sabrina Paganoni, Paolo Bonato, and Emilio Bizzi. Muscle synergy patterns as physiological markers of motor cortical damage. *Proceedings of the national academy of sciences*, 109(36):14652–14656, 2012.

[409] Vincent CK Cheung, Lamberto Piron, Michela Agostini, Stefano Silvoni, Andrea Turolla, and Emilio Bizzi. Stability of muscle synergies for voluntary actions after cortical stroke in humans. *Proceedings of the National Academy of Sciences*, 106(46):19563–19568, 2009.

[410] Nikolaus Wenger, Eduardo Martin Moraud, Jerome Gandar, Pavel Musienko, Marco Capogrosso, Laetitia Baud, Camille G Le Goff, Quentin Barraud, Natalia Pavlova, Nadia Dominici, et al. Spatiotemporal neuromodulation therapies engaging muscle synergies improve motor control after spinal cord injury. *Nature medicine*, 22(2):138, 2016.

[411] Laura Dipietro, Hermano I Krebs, Susan E Fasoli, Bruce T Volpe, Joel Stein, C Bever, and Neville Hogan. Changing motor synergies in chronic stroke. *Journal of neurophysiology*, 98(2):757–768, 2007.

[412] Peppino Tropea, Vito Monaco, Martina Coscia, Federico Posteraro, and Silvestro Micera. Effects of early and intensive neuro-rehabilitative treatment on muscle synergies in acute post-stroke patients: a pilot study. *Journal of neuroengineering and rehabilitation*, 10(1):103, 2013.

[413] Seyed Safavynia, Gelsy Torres-Oviedo, and Lena Ting. Muscle synergies: implications for clinical evaluation and rehabilitation of movement. *Topics in spinal cord injury rehabilitation*, 17(1):16–24, 2011.

[414] M. Fiorentino, A. E. Uva, M. M. Foglia, and V. Bevilacqua. Wearable rumble device for active asymmetry measurement and correction in lower limb mobility. In *2011 IEEE International Symposium on Medical Measurements and Applications*, pages 550–554, May 2011. doi: 10.1109/MeMeA.2011.5966767.

[415] Andrea d'Avella, Alessandro Portone, Laure Fernandez, and Francesco Lacquaniti. Control of fast-reaching movements by muscle synergy combinations. *Journal of Neuroscience*, 26(30):7791–7810, 2006. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.0830-06.2006. URL http://www.jneurosci.org/content/26/30/7791.

[416] Francesca Lunardini, Claudia Casellato, Andrea d'Avella, Terence D Sanger, and Alessandra Pedrocchi. Robustness and reliability of synergy-based myocontrol of a multiple degree of freedom robotic arm. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(9):940–950, 2015.

[417] Ning Jiang, Hubertus Rehbaum, Ivan Vujaklija, Bernhard Graimann, and Dario Farina. Intuitive, online, simultaneous, and proportional myoelectric control over two degrees-of-freedom in upper limb amputees. *IEEE transactions on neural systems and rehabilitation engineering*, 22(3):501–510, 2013.

[418] Matthew C Tresch, Vincent CK Cheung, and Andrea d'Avella. Matrix factorization algorithms for the identification of muscle synergies: evaluation on simulated and experimental data sets. *Journal of neurophysiology*, 95(4):2199–2212, 2006.

[419] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 556–562. MIT Press, 2001. URL http://papers.nips.cc/paper/1861-algorithms-for-non-negative-matrix-factorization.pdf.

[420] Matthew C. Tresch, Vincent C. K. Cheung, and Andrea d'Avella. Matrix factorization algorithms for the identification of muscle synergies: Evaluation on simulated and experimental data sets. *Journal of Neurophysiology*, 95(4):2199–2212, 2006. doi: 10.1152/jn.00222.2005. URL https://doi.org/10.1152/jn.00222.2005. PMID: 16394079.

[421] Luis Pelaez Murciego, Michele Barsotti, and Antonio Frisoli. Synergy-based multi-fingers forces reconstruction and discrimination from forearm emg. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*, pages 204–213. Springer, 2018.

[422] Hubertus Rehbaum, Ning Jiang, Liliana Paredes, Sebastian Amsuess, Bernhard Graimann, and Dario Farina. Real time simultaneous and proportional control of multiple degrees of freedom from surface emg: preliminary results on subjects with limb deficiency. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 1346–1349. IEEE, 2012.

[423] Ning Jiang, Strahinja Dosen, Klaus-Robert Muller, and Dario Farina. Myoelectric control of artificial limbs—is there a need to change focus?[in the spotlight]. *IEEE Signal Processing Magazine*, 29(5):152–150, 2012.

[424] Tim A. Valk, Leonora J. Mouton, Egbert Otten, and Raoul M. Bongers. Fixed muscle synergies and their potential to improve the intuitive control of myoelectric assistive technology for upper extremities. *Journal of NeuroEngineering and Rehabilitation*, 16(1):6, Jan 2019. ISSN 1743-0003. doi: 10.1186/s12984-018-0469-5. URL https://doi.org/10.1186/s12984-018-0469-5.

[425] Dennis Tkach, He Huang, and Todd A Kuiken. Study of stability of time-domain features for electromyographic pattern recognition. *Journal of neuroengineering and rehabilitation*, 7(1):21, 2010.

[426] Mayo Clinic. Movement disorders. URL https://www.mayoclinic.org/diseases-conditions/movement-disorders/symptoms-causes/syc-20363893. Last visited: November 2020.

[427] Giovanni Defazio, Mark Hallett, Hyder A. Jinnah, and Alfredo Berardelli. Development and validation of a clinical guideline for diagnosing blepharospasm. *Neurology*, 81(3):236–240, 2013. ISSN 0028-3878. doi: 10.1212/WNL.0b013e31829bfdf6. URL https://n.neurology.org/content/81/3/236.

[428] F Grandas, J Elston, N Quinn, and C D Marsden. Blepharospasm: a review of 264 patients. *Journal of Neurology, Neurosurgery & Psychiatry*, 51(6):767–772, 1988. ISSN 0022-3050. doi: 10.1136/jnnp.51.6.767. URL https://jnnp.bmj.com/content/51/6/767.

[429] Mark Hallett, Craig Evinger, Joseph Jankovic, and Mark Stacy. Update on blepharospasm. *Neurology*, 71(16):1275–1282, 2008. ISSN 0028-3878. doi: 10.1212/01.wnl.0000327601.46315.85. URL https://n.neurology.org/content/71/16/1275.

[430] H. A. Jinnah, Alfredo Berardelli, Cynthia Comella, Giovanni DeFazio, Mahlon R. DeLong, Stewart Factor, Wendy R. Galpern, Mark Hallett, Christy L. Ludlow, Joel S. Perlmutter, Ami R. Rosen, and for the Dystonia Coalition Investigators. The focal dystonias: Current views and challenges for future research. *Movement Disorders*, 28

(7):926–943, 2013. doi: 10.1002/mds.25567. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mds.25567.

[431] Giovanni Defazio, Mark Hallett, Hyder A. Jinnah, Glenn T. Stebbins, Angelo F. Gigante, Gina Ferrazzano, Antonella Conte, Giovanni Fabbrini, and Alfredo Berardelli. Development and validation of a clinical scale for rating the severity of blepharospasm. *Movement Disorders*, 30(4):525–530, 2015. doi: https://doi.org/10.1002/mds.26156. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mds.26156.

[432] Davide Martino, Giovanni Defazio, Giovanni Alessio, Giovanni Abbruzzese, Paolo Girlanda, Michele Tinazzi, Giovanni Fabbrini, Lucio Marinelli, Giovanni Majorana, Maria Buccafusca, Laura Vacca, Paolo Livrea, and Alfredo Berardelli. Relationship between eye symptoms and blepharospasm: A multicenter case–control study. *Movement Disorders*, 20(12):1564–1570, 2005. doi: https://doi.org/10.1002/mds.20635. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mds.20635.

[433] Antonella Conte, Giovanni Defazio, Gina Ferrazzano, Mark Hallett, Antonella Macerollo, Giovanni Fabbrini, and Alfredo Berardelli. Is increased blinking a form of blepharospasm? *Neurology*, 80(24):2236–2241, 2013. ISSN 0028-3878. doi: 10.1212/WNL.0b013e318296e99d. URL https://n.neurology.org/content/80/24/2236.

[434] Marta Ugarte and Masoud Teimory. Apraxia of lid opening. *British Journal of Ophthalmology*, 91(7):854–854, 2007. ISSN 0007-1161. doi: 10.1136/bjo.2007.124040. URL https://bjo.bmj.com/content/91/7/854.

[435] David A. Peterson, Gwen C. Littlewort, Marian S. Bartlett, Antonella Macerollo, Joel S. Perlmutter, H.A. Jinnah, Mark Hallett, and Terrence J. Sejnowski. Objective, computerized video-based rating of blepharospasm severity. *Neurology*, 87(20):2146–2153, 2016. ISSN 0028-3878. doi: 10.1212/WNL.0000000000003336. URL https://n.neurology.org/content/87/20/2146.

[436] Robert E. Burke, Stanley Fahn, C. David Marsden, Susan B. Bressman, Carol Moskowitz, and Joseph Friedman. Validity and reliability of a rating scale for the primary torsion dystonias. *Neurology*, 35(1):73–73, 1985. ISSN 0028-3878. doi: 10.1212/WNL.35.1.73. URL https://n.neurology.org/content/35/1/73.

[437] Cynthia L. Comella, Sue Leurgans, Joanne Wuu, Glenn T. Stebbins, Teresa Chmura, , and The Dystonia Study Group. Rating scales for dystonia: A multicenter assessment. *Movement Disorders*, 18(3):303–312, 2003. doi: https://doi.org/10.1002/mds.10377. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mds.10377.

[438] Joseph Jankovic and Janet Orman. Botulinum a toxin for cranial-cervical dystonia. *Neurology*, 37(4):616–616, 1987. ISSN 0028-3878. doi: 10.1212/WNL.37.4.616. URL https://n.neurology.org/content/37/4/616.

[439] Vitoantonio Bevilacqua, Antonio Emmanuele Uva, Michele Fiorentino, Gianpaolo Francesco Trotta, Maurizio Dimatteo, Enrico Nasca, Attilio Nicola Nocera,

Giacomo Donato Cascarano, Antonio Brunetti, Nicholas Caporusso, Roberta Pellicciari, and Giovanni Defazio. A comprehensive method for assessing the blepharospasm cases severity. In K.C. Santosh, Mallikarjun Hangarge, Vitoantonio Bevilacqua, and Atul Negi, editors, *Recent Trends in Image Processing and Pattern Recognition*, pages 369–381, Singapore, 2017. Springer Singapore. ISBN 978-981-10-4859-3.

[440] Christel Bidet-Ildei, Pierre Pollak, Sonia Kandel, Valérie Fraix, and Jean-Pierre Orliaguet. Handwriting in patients with Parkinson disease: Effect of L-dopa and stimulation of the sub-thalamic nucleus on motor anticipation. *Human movement science*, 30(4):783–791, 2011.

[441] Eli Carmeli, Hagar Patish, and Raymond Coleman. The aging hand. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 58(2):M146–M152, 2003.

[442] J E McLennan, K Nakano, H R Tyler, and R S Schwab. Micrographia in Parkinson's disease. *Journal of the neurological sciences*, 15(2):141–152, 1972.

[443] Tamar Flash, Rivka Inzelberg, Edna Schechtman, and Amos D Korczyn. Kinematic analysis of upper limb trajectories in parkinson's disease. *Experimental neurology*, 118 (2):215–226, 1992.

[444] David I Margolin and Alan M Wing. Agraphia and micrographia: Clinical manifestations of motor programming and performance disorders. *Acta Psychologica*, 54(1): 263–283, 1983.

[445] F Müller and G E Stelmach. Prehension movements in Parkinson's disease. *Advances in psychology*, 87:307–319, 1992.

[446] José L Contreras-Vidal, Hans-Leo Teulings, and George E Stelmach. Micrographia in Parkinson's disease. *Neuroreport*, 6(15):2089–2092, 1995.

[447] A W A Van Gemmert, H-L Teulings, Jose L Contreras-Vidal, and G E Stelmach. Parkinsons disease and the control of size and speed in handwriting. *Neuropsychologia*, 37(6):685–694, 1999.

[448] Arend W A Van Gemmert, Hans-Leo Teulings, and George E Stelmach. Parkinsonian patients reduce their stroke size with increased processing demands. *Brain and cognition*, 47(3):504–512, 2001.

[449] H L Teulings, Jose L Contreras-Vidal, G E Stelmach, and Charles Howard Adler. Adaptation of handwriting size under distorted visual feedback in patients with Parkinson's disease and elderly and young controls. *Journal of Neurology, Neurosurgery & Psychiatry*, 72(3):315–324, 2002.

[450] Peter Drotar, Jiri Mekyska, Zdenek Smekal, Irena Rektorova, Lucia Masarova, and Marcos Faundez-Zanuy. Prediction potential of different handwriting tasks for diagnosis

of parkinson's. In *E-Health and Bioengineering Conference (EHB), 2013*, pages 1–4. IEEE, 2013.

[451] John G Nutt and G Frederick Wooten. Diagnosis and initial management of parkinson's disease. *New England Journal of Medicine*, 353(10):1021–1027, 2005.

[452] John G Nutt, Eric S Lea, Laura Van Houten, Robert A Schuff, and Gary J Sexton. Determinants of tapping speed in normal control subjects and subjects with Parkinson's disease: differing effects of brief and continued practice. *Movement disorders*, 15(5): 843–849, 2000.

[453] Andrew M Gordon. Task-dependent deficits during object release in parkinson's disease. *Experimental neurology*, 153(2):287–298, 1998.

[454] James R Tresilian, George E Stelmach, and Charles H Adler. Stability of reach-to-grasp movement patterns in parkinson's disease. *Brain*, 120(11):2093–2111, 1997.

[455] Miya K Rand, George E Stelmach, and James R Bloedel. Movement accuracy constraints in parkinson's disease patients. *Neuropsychologia*, 38(2):203–212, 2000.

[456] A W A Van Gemmert, Charles Howard Adler, and G E Stelmach. Parkinson's disease patients undershoot target size in handwriting and similar tasks. *Journal of Neurology, Neurosurgery & Psychiatry*, 74(11):1502–1508, 2003.

[457] P. Viviani and G. McCollum. The relation between linear extent and velocity in drawing movements. *Neuroscience*, 10(1):211 – 218, 1983. ISSN 0306-4522. doi: https://doi.org/10.1016/0306-4522(83)90094-5. URL http://www.sciencedirect.com/science/article/pii/0306452283900945.

[458] Eric Helsper, Hans-Leo Teulings, Elisabeth Karamat, and George E Stelmach. Preclinical parkinson features in optically scanned handwriting. *Handwriting and Drawing Research: Basic and Applied Issues. IOS Press, Amsterdam*, pages 241–250, 1996.

[459] Mitchell G Longstaff, Padma R Mahant, Mark A Stacy, Arend W A Van Gemmert, Berta C Leis, and George E Stelmach. Discrete and dynamic scaling of the size of continuous graphic movements of parkinsonian patients and elderly controls. *Journal of Neurology, Neurosurgery & Psychiatry*, 74(3):299–304, 2003.

[460] Atilla Ünlü, Rüdiger Brause, and Karsten Krakow. Handwriting analysis for diagnosis and prognosis of parkinson's disease. In *International Symposium on Biological and Medical Data Analysis*, pages 441–450. Springer, 2006.

[461] Sara Rosenblum, Margalit Samuel, Sharon Zlotnik, Ilana Erikh, and Ilana Schlesinger. Handwriting as an objective tool for Parkinson's disease diagnosis. *Journal of neurology*, 260(9):2357–2361, 2013.

[462] John Wann and Ian Nimmo-Smith. The control of pen pressure in handwriting: A subtle point. *Human Movement Science*, 10(2):223–246, 1991.

[463] Denis Alamargot and Marie-France Morin. Does handwriting on a tablet screen affect students' graphomotor execution? A comparison between grades two and nine. *Human movement science*, 44:32–41, 2015.

[464] Paola Suavo-Bulzis, Federica Albanese, Davide Mallardi, Francesco Saverio Debitonto, Ruggero Lemma, Annalisa Granatiero, Marisa Spadavecchia, Giacomo Donato Cascarano, Vitoantonio Bevilacqua, Loreto Gesualdo, and Francesco Pesce. P0119 ARTIFICIAL INTELLIGENCE IN RENAL PATHOLOGY: IBM WATSON FOR THE IDENTIFICATION OF GLOMERULOSCLEROSIS. *Nephrology Dialysis Transplantation*, 35(Supplement_3):418, jun 2020. ISSN 0931-0509. doi: 10.1093/ndt/gfaa142.P0119. URL https://academic.oup.com/ndt/article/doi/10.1093/ndt/gfaa142.P0119/5852854.

[465] Berardino Prencipe, Nicola Altini, Giacomo Donato Cascarano, Andrea Guerriero, and Antonio Brunetti. A novel approach based on region growing algorithm for liver and spleen segmentation from ct scans. In *International Conference on Intelligent Computing*, pages 398–410. Springer, 2020.

[466] Giacomo Donato Cascarano, Francesco Saverio Debitonto, Ruggero Lemma, Antonio Brunetti, Domenico Buongiorno, Irio De Feudis, Andrea Guerriero, Umberto Venere, Silvia Matino, Maria Teresa Rocchetti, Michele Rossini, Francesco Pesce, Loreto Gesualdo, and Vitoantonio. Bevilacqua. A neural network for glomerulus classification based on histological images of kidney biopsy. BMC Supplements (Under Review).

[467] Tony F. Chan and Luminita A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 2001. ISSN 10577149. doi: 10.1109/83.902291.

[468] Stuart Lloyd. Least squares quantization in PCM. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[469] Sunhua Wan, Hsiang Chieh Lee, Xiaolei Huang, Ting Xu, Tao Xu, Xianxu Zeng, Zhan Zhang, Yuri Sheikine, James L. Connolly, James G. Fujimoto, and Chao Zhou. Integrated local binary pattern texture features for classification of breast tissue imaged by optical coherence microscopy. *Medical Image Analysis*, 2017. ISSN 13618423. doi: 10.1016/j.media.2017.03.002.

[470] Bowen Song, Guopeng Zhang, Wei Zhu, and Zhengrong Liang. ROC operating point selection for classification of imbalanced data with application to computer-aided polyp detection in CT colonography. *International Journal of Computer Assisted Radiology and Surgery*, 2014. ISSN 18616429. doi: 10.1007/s11548-013-0913-8.

[471] Azeddine Elhassouny, Le Nhan Tam, Dina Sayed, Bjoern Steffens, and Lak Sri. *Building Cognitive Applications with IBM Watson Services: Volume 3 Visual Recognition*. IBM Redbooks, 2017.

[472] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, 39(6):1137–1149, 2017. ISSN 01628828. doi: 10.1109/TPAMI.2016.2577031.

[473] Waleed Abdulla. Mask r-cnn for object detection and instance segmentation on keras and tensorflow. https://github.com/matterport/Mask_RCNN, 2017.

[474] Jaya S Kulchandani and Kruti J Dangarwala. Moving object detection: Review of recent research trends. In *2015 International Conference on Pervasive Computing (ICPC)*, pages 1–5. IEEE, 2015.

[475] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 391–405, Cham, 2014. Springer. ISBN 978-3-319-10601-4. doi: 10.1007/978-3-319-10602-1_26.

[476] Ross B. Girshick. Fast R-CNN. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1440–1448, 2015. doi: 10.1109/ICCV.2015.169.

[477] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[478] Amber L Simpson, Michela Antonelli, Spyridon Bakas, Michel Bilello, Keyvan Farahani, Bram Van Ginneken, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, et al. A large annotated medical image dataset for the development and evaluation of segmentation algorithms. *arXiv preprint arXiv:1902.09063*, 2019.

[479] M. Edman. Segmentation using a region growing algorithm. Technical report, 10 2007.

[480] Chen Shen, Fausto Milletari, Holger R. Roth, Hirohisa Oda, Masahiro Oda, Yuichiro Hayashi, Kazunari Misawa, and Kensaku Mori. Improving V-Nets for multi-class abdominal organ segmentation. In Elsa D. Angelini and Bennett A. Landman, editors, *Medical Imaging 2019: Image Processing*, volume 10949, pages 76 – 82. International Society for Optics and Photonics, SPIE, 2019. doi: 10.1117/12.2512790. URL https://doi.org/10.1117/12.2512790.

[481] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. Tversky loss function for image segmentation using 3d fully convolutional deep networks. In *International Workshop on Machine Learning in Medical Imaging*, pages 379–387. Springer, 2017.

[482] Antonio Frisoli, Fabrizio Rocchi, Simone Marcheschi, Andrea Dettori, Fabio Salsedo, and Massimo Bergamasco. A new force-feedback arm exoskeleton for haptic interaction in virtual environments. In *First Joint Eurohaptics Conference and Symposium on*

Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics Conference, pages 195–201. IEEE, 2005.

[483] Stacie A Chvatal, Gelsy Torres-Oviedo, Seyed A Safavynia, and Lena H Ting. Common muscle synergies for control of center of mass and force in nonstepping and stepping postural behaviors. *Journal of neurophysiology*, 106(2):999–1015, 2011.

[484] Lena H Ting and Jane M Macpherson. A limited set of muscle synergies for force control during a postural task. *Journal of neurophysiology*, 93(1):609–613, 2005.

[485] Gelsy Torres-Oviedo, Jane M Macpherson, and Lena H Ting. Muscle synergy organization is robust across a variety of postural perturbations. *Journal of neurophysiology*, 96(3):1530–1546, 2006.

[486] Victor R Barradas, Jason J Kutch, Toshihiro Kawase, Yasuharu Koike, and Nicolas Schweighofer. When 90% of the variance is not enough: residual emg from muscle synergy extraction influences task performance. *BioRxiv*, page 634758, 2019.

[487] Gianpaolo F. Trotta, Roberta Pellicciari, Antonio Boccaccio, Antonio Brunetti, Giacomo D. Cascarano, Vito M. Manghisi, Michele Fiorentino, Antonio E. Uva, Giovanni Defazio, and Vitoantonio Bevilacqua. A neural network-based software to recognise blepharospasm symptoms and to measure eye closure time. *Computers in Biology and Medicine*, 112:103376, sep 2019. ISSN 00104825. doi: 10.1016/j.compbiomed.2019.103376. URL https://linkinghub.elsevier.com/retrieve/pii/S0010482519302537.

[488] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[489] Christos Sagonas, Epameinondas Antonakos, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: database and results. *Image and Vision Computing*, 47:3 – 18, 2016. ISSN 0262-8856. doi: https://doi.org/10.1016/j.imavis.2016.01.002. URL http://www.sciencedirect.com/science/article/pii/S0262885616000147. 300-W, the First Automatic Facial Landmark Detection in-the-Wild Challenge.

[490] A. BERARDELLI, J. C. ROTHWELL, B. L. DAY, and C. D. MARSDEN. PATHOPHYSIOLOGY OF BLEPHAROSPASM AND OROMANDIBULAR DYSTONIA. *Brain*, 108(3):593–608, 09 1985. ISSN 0006-8950. doi: 10.1093/brain/108.3.593. URL https://doi.org/10.1093/brain/108.3.593.

[491] Bo Zhang, Wenjun Wang, and Bo Cheng. Driver eye state classification based on cooccurrence matrix of oriented gradients. *Advances in Mechanical Engineering*, 7(2):707106, 2015. doi: 10.1155/2014/707106. URL https://doi.org/10.1155/2014/707106.

[492] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: the rprop algorithm. In *IEEE International Conference on Neural Networks*, pages 586–591 vol.1, 1993. doi: 10.1109/ICNN.1993.298623.

[493] Claudio Loconsole, Gianpaolo Francesco Trotta, Antonio Brunetti, Joseph Trotta, Angelo Schiavone, Sabina Ilaria Tatò, Giacomo Losavio, and Vitoantonio Bevilacqua. Computer vision and emg-based handwriting analysis for classification in parkinson's disease. In De-Shuang Huang, Kang-Hyun Jo, and Juan Carlos Figueroa-García, editors, *Intelligent Computing Theories and Application*, pages 493–503, Cham, 2017. Springer International Publishing. ISBN 978-3-319-63312-1.

[494] Naiqian Zhi, Beverly Jaeger, Andrew Gouldstone, Rifat Sipahi, and Samuel Frank. Toward monitoring parkinson's through analysis of static handwriting samples: A quantitative analytical framework. *IEEE journal of biomedical and health informatics*, 2016.

[495] Mari Raudmann, Pille Taba, and Kadri Medijainen. Handwriting speed and size in individuals with Parkinson's disease compared to healthy controls: the possible effect of cueing. *Acta Kinesiologiae Universitatis Tartuensis*, 20:40–47, 2014.

[496] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[497] Zhan Li Sun, De Shuang Huang, Yiu Ming Cheung, Jiming Liu, and Guang Bin Huang. Using FCMC, FVS, and PCA techniques for feature extraction of multispectral images. *IEEE Geoscience and Remote Sensing Letters*, 2(2):108–112, apr 2005. ISSN 1545598X. doi: 10.1109/LGRS.2005.844169. URL http://ieeexplore.ieee.org/document/1420284/.

[498] Leo Breiman. *Classification and regression trees*. 2017.