From point cloud to parametric 3D model: polygonal modelling approach for optimising Scan to H-BIM and Scan to H-FEM processes

D.R.S.A.T.E.

Doctor in Risk And Environmental, Territorial And Building Development

POLITECNICO DI BARI

02
2024

Coordinator: Prof. Vito Iacobellis

XXXVI CYCLE
Curriculum: ICAR/06 – Topography and Cartography

DICATECh
Department of Civil, Environmental, Building Engineering and Chemistry

Vincenzo Saverio ALFIO

From point cloud to parametric 3D model: polygonal modelling approach for optimising Scan to H-BIM and Scan to H-FEM processes.

Prof. Domenica COSTANTINO
Department of Civil, Environmental, Building Engineering and Chemistry
Polytechnic University of Bari
Prof. Massimiliano PEPE
InGEO - Department of Engineering and Geology
University «G. D'Annunzio» of Chieti - Pescara
Prof. Sorin HERBAN
Department of Overland Communication Ways, Foundation and Cadastral Survey
Universitatea Politehnica Timişoara - Romania

Politecnico di Bari

LIBERATORIA PER L'ARCHIVIAZIONE DELLA TESI DI DOTTORATO

Al Magnifico Rettore
del Politecnico di Bari

Il sottoscritto **Alfio Vincenzo Saverio** nato a **Grottaglie (TA)** il **21/09/1987** residente a **Grottaglie (TA)** in **via Palermo n.64/A** e-mail **vincenzosaverio.alfio@poliba.it** iscritto al 3° anno di Corso di Dottorato di Ricerca in **Rischio e Sviluppo Ambientale, Territoriale ed Edilizio** ciclo **XXXVI** ed essendo stato ammesso a sostenere l'esame finale con la prevista discussione della tesi dal titolo: **From point cloud to parametric 3D model: polygonal modelling approach for optimising Scan to H-BIM and Scan to H-FEM processes,**

DICHIARA

1) di essere consapevole che, ai sensi del D.P.R. n. 445 del 28.12.2000, le dichiarazioni mendaci, la falsità negli atti e l'uso di atti falsi sono puniti ai sensi del codice penale e delle Leggi speciali in materia, e che nel caso ricorressero dette ipotesi, decade fin dall'inizio e senza necessità di nessuna formalità dai benefici conseguenti al provvedimento emanato sulla base di tali dichiarazioni;
2) di essere iscritto al Corso di Dottorato di ricerca **Rischio e Sviluppo Ambientale, Territoriale ed Edilizio** ciclo **XXXVI**, corso attivato ai sensi del "*Regolamento dei Corsi di Dottorato di ricerca del Politecnico di Bari*", emanato con D.R. n.286 del 01.07.2013;
3) di essere pienamente a conoscenza delle disposizioni contenute nel predetto Regolamento in merito alla procedura di deposito, pubblicazione e autoarchiviazione della tesi di dottorato nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica;
4) di essere consapevole che attraverso l'autoarchiviazione delle tesi nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica del Politecnico di Bari (IRIS-POLIBA), l'Ateneo archivierà e renderà consultabile in rete (nel rispetto della Policy di Ateneo di cui al D.R. 642 del 13.11.2015) il testo completo della tesi di dottorato, fatta salva la possibilità di sottoscrizione di apposite licenze per le relative condizioni di utilizzo (di cui al sito http://www.creativecommons.it/Licenze), e fatte salve, altresì, le eventuali esigenze di "embargo", legate a strette considerazioni sulla tutelabilità e sfruttamento industriale/commerciale dei contenuti della tesi, da rappresentarsi mediante compilazione e sottoscrizione del modulo in calce (Richiesta di embargo);
5) che la tesi da depositare in IRIS-POLIBA, in formato digitale (PDF/A) sarà del tutto identica a quelle **consegnate**/inviate/da inviarsi ai componenti della commissione per l'esame finale e a qualsiasi altra copia depositata presso gli Uffici del Politecnico di Bari in forma cartacea o digitale, ovvero a quella da discutere in sede di esame finale, a quella da depositare, a cura dell'Ateneo, presso le Biblioteche Nazionali Centrali di Roma e Firenze e presso tutti gli Uffici competenti per legge al momento del deposito stesso, e che di conseguenza va esclusa qualsiasi responsabilità del Politecnico di Bari per quanto riguarda eventuali errori, imprecisioni o omissioni nei contenuti della tesi;
6) che il contenuto e l'organizzazione della tesi è opera originale realizzata dal sottoscritto e non compromette in alcun modo i diritti di terzi, ivi compresi quelli relativi alla sicurezza dei dati personali; che pertanto il Politecnico di Bari ed i suoi funzionari sono in ogni caso esenti da responsabilità di qualsivoglia natura: civile, amministrativa e penale e saranno dal sottoscritto tenuti indenni da qualsiasi richiesta o rivendicazione da parte di terzi;
7) che il contenuto della tesi non infrange in alcun modo il diritto d'Autore né gli obblighi connessi alla salvaguardia di diritti morali od economici di altri autori o di altri aventi diritto, sia per testi, immagini, foto, tabelle, o altre parti di cui la tesi è composta.

Luogo e data: **Bari, 29/12/2023** Firma _____

Il/La sottoscritto, con l'autoarchiviazione della propria tesi di dottorato nell'Archivio Istituzionale ad accesso aperto del Politecnico di Bari (POLIBA-IRIS), pur mantenendo su di essa tutti i diritti d'autore, morali ed economici, ai sensi della normativa vigente (Legge 633/1941 e ss.mm.ii.),

CONCEDE

- al Politecnico di Bari il permesso di trasferire l'opera su qualsiasi supporto e di convertirla in qualsiasi formato al fine di una corretta conservazione nel tempo. Il Politecnico di Bari garantisce che non verrà effettuata alcuna modifica al contenuto e alla struttura dell'opera.
- al Politecnico di Bari la possibilità di riprodurre l'opera in più di una copia per fini di sicurezza, back-up e conservazione.

Luogo e data: **Bari, 29/12/2023** Firma _____

D.R.S.A.T.E.

POLITECNICO DI BARI

02

2024

Doctor in Risk And Environmental, Territorial And Building Development

Coordinator: Prof. Vito Iacobellis

XXXVI CYCLE
Curriculum: ICAR/06 – Topography and Cartography

DICATECh
Department of Civil, Environmental, Building Engineering and Chemistry

Vincenzo Saverio ALFIO

From point cloud to parametric 3D model: polygonal modelling approach for optimising Scan to H-BIM and Scan to H-FEM processes.

Prof. Domenica COSTANTINO
Department of Civil, Environmental, Building Engineering and Chemistry
Polytechnic University of Bari

Prof. Massimiliano PEPE
InGEO - Department of Engineering and Geology
University «G. D'Annunzio» of Chieti - Pescara

Prof. Sorin HERBAN
Department of Overland Communication Ways, Foundation and Cadastral Survey
Universitatea Politehnica Timişoara - Romania

D.R.S.A.T.E.

POLITECNICO DI BARI

02

2024

Dottorato in Rischio e Sviluppo Ambientale, Territoriale ed Edilizio

Coordinatore: Prof. Vito Iacobellis

XXXVI CYCLE
Curriculum: ICAR/06 – Topografia e Cartografia

DICATECh
Dipartimento di Ingegneria Civile, Ambientale, del Territorio, Edile e di Chimica

Vincenzo Saverio ALFIO

Dalla nuvola di punti al modello 3D parametrico: approccio di modellazione poligonale per l'ottimizzazione di processi Scan to H-BIM e Scan to H-FEM.

Prof. Domenica COSTANTINO
Department of Civil, Environmental, Building Engineering and Chemistry
Polytechnic University of Bari
Prof. Massimiliano PEPE
InGEO – Dipartimento di Ingegneria e Geologia
Università «G. D'Annunzio» di Chieti - Pescara
Prof. Sorin HERBAN
Dipartimento delle vie di comunicazione terrestri, delle fondazioni e dei rilievi catastali
Universitatea Politehnica Timişoara - Romania

EXTENDED ABSTRACT *(eng)*

3D parametric modelling is a current and important research topic for the digitisation of cultural heritage in order to preserve the evolution and historical identity of an area over time; for this reason, cultural heritage assets must be subjected to continuous maintenance and monitoring, as well as valorisation and promotion. In addition, a considerable amount of these assets are often exposed to a high risk of damage caused by natural disasters and a poor state of care and neglect that leads to inevitable loss. In this context, in order to adopt all safeguarding and preservation measures in the field of CH, new sensors, digital acquisition technologies and suitable methodological approaches make it possible to create high-performance three-dimensional models, capable of increasing the detail and representation of particulars and enriching these models with heterogeneous information.

However, such 3D models are often difficult to manage as they require very high processing and storage capacities. Therefore, it is necessary to define a suitable methodology that is capable of simplifying and optimising the three-dimensional models obtained from a photogrammetric or TLS approach, preserving their accuracy and metric rigour and, at the same time, performing efficiently in BIM (Building Information Modeling) and/or FEM (Finite Element Analysis) applications.

The aim of this work is to identify a suitable process that is able, starting from a point cloud obtained by means of geomatic techniques, to generate three-dimensional surface models that are subsequently decimated and optimised according to the complexity of the case and the desired output. To achieve these objectives, a semi-automatic process is described that is capable of transforming a TIN (Triangulated Irregular Network) surface model into a quad-mesh polygonal model and then converting it into

NURBS (Non-Uniform Rational Basis-Splines) for subsequent import into a BIM software or FEM solver for finite element calculations. As well as the optimisation requirements, this approach also pursues the objective of improving the quality of the final model, providing metrics (validated by means of suitable algorithms) and geometric resolution, resolving all the topological errors of the generated meshes (open surfaces, holes, edges, etc.), as well as optimising the computational aspects, sharing and interoperability between the different software and platforms used.

The experimentation was carried out on two different case studies, and in particular a 14th-century church and a bronze statue, demonstrating the efficiency of the optimisation and management of complex geometries and validating, through a study on the repeatability of the approach considered, an efficient and accurate Scan to BIM and Scan to FEM process applied to existing cultural heritage.

Keywords

Point Cloud, 3D survey, Mesh, Scan to H-BIM, Scan to H-FEM.

EXTENDED ABSTRACT *(ita)*

La modellazione parametrica 3D rappresenta un tema di ricerca attuale e importante per la digitalizzazione del patrimonio culturale al fine di preservare l'evoluzione e l'identità storica di un territorio nel tempo; per questo motivo, i beni appartenenti al patrimonio culturale, devono essere sottoposti a continue attività di manutenzione e monitoraggio, nonché di valorizzazione e promozione. Inoltre, una quantità considerevole di questi beni, spesso sono esposti ad un elevato rischio di danni causato da calamità naturali e da un cattivo stato di custodia e incuria che porta ad un inevitabile perdita. In quest'ottica, al fine di adottare tutte le misure di salvaguardia e tutela nel campo del CH, nuovi sensori, tecnologie di acquisizione digitale ed idonei approcci metodologici, consentono di realizzare modelli tridimensionali altamente performanti, in grado di aumentare il dettaglio e la rappresentazione dei particolari ed arricchire di informazioni eterogenee tali modelli.

Tuttavia, tali modelli 3D molte volte risultano difficilmente gestibili in quanto richiedono capacità di elaborazione ed archiviazione molto elevate; in questo contesto è necessario quindi definire una idonea metodologia che sia in grado di semplificare ed ottimizzare i modelli tridimensionali ottenuti da un approccio fotogrammetrico o TLS, conservandone l'accuratezza e il rigore metrico e, allo stesso tempo, che siano performanti nelle applicazioni BIM (Building Information Modeling) e/o FEM (Finite Element Analysis).

Lo scopo del presente lavoro è quello di individuare un idoneo processo che sia in grado, partendo da una nuvola di punti ottenuta mediante tecniche geomatiche, di generare modelli di superficie tridimensionali che successivamente vengono decimati e ottimizzati in funzione della complessità del caso e dell'output desiderato. Per

raggiungere tali obiettivi, viene descritto un processo semi-automatico in grado di trasformare un modello di superficie TIN (Triangulated Irregular Network), in un modello poligonale quad-mesh e in seguito convertirlo in NURBS (Non-Uniform Rational Basis-Splines) per essere importato successivamente in un software di progettazione BIM o in un solutore FEM per il calcolo agli elementi finiti. Tale approccio, oltre a rispondere alle esigenze di ottimizzazione, persegue anche l'obiettivo di migliorare la qualità del modello finale, garantendo la metrica (validata mediante idonei algoritmi) e la risoluzione geometrica, risolvendo tutti gli errori topologici delle mesh generate (superfici aperte, buchi, bordi, ecc..), nonché di ottimizzare gli aspetti computazionali, la condivisione e l'interoperabilità tra i diversi software e piattaforme utilizzate.

La sperimentazione è stata condotta su due casi studio differenti, ed in particolare una chiesa del XIV secolo e una statua in bronzo, dimostrando l'efficienza dell'ottimizzazione e gestione delle geometrie complesse e validando, attraverso uno studio sulla ripetibilità dell'approccio considerato, un efficiente ed accurato processo di Scan to BIM e Scan to FEM applicato al patrimonio culturale esistente.

Keywords

Nuvola di punti, Rilievo 3D, Mesh, Scan to H-BIM, Scan to H-FEM.

INDEX

INTRODUCTION

Nowadays, thanks to increasingly advanced technologies, it is possible to obtain three-dimensional reproductions of models or structures of considerable historical, architectural and cultural value, of simple and complex shapes, of different sizes and with techniques and methods appropriate to the type of structure (statues, bridges, castles, etc.). If, in the past, the digitisation of objects belonging to the CH was carried out through two- and three-dimensional graphic representations, the challenge of the future is to enrich the digitisation process with new content of a descriptive (historical, artistic, etc.), material (chemical-physical characteristics), evolutionary (processes of change) and conservation (state of health) nature.

Worldwide, there are numerous cultural heritage sites to be protected and enhanced; for example, Italy is a country that possesses an immense cultural heritage, accounting for about 70 per cent of the world's cultural heritage as well as the largest number of sites included in the Unesco World Heritage List. However, many of these sites are in a state of semi-abandonment, condemned to degradation or complete destruction due to neglect or negligence. The aspects of enhancement, maintenance, management and conservation should be the focus of initiatives to support and promote these assets. Therefore, these processes related to the protection of cultural heritage (CH) should play an important role in the knowledge, memory and testimony of the identity culture of a territory. They can become virtuous on every structure belonging to the CH, where digitisation procedures capable of describing every morphological, typological, semantic and structural detail are triggered.

Very often the digitisation of cultural assets has been limited to the generation of 3D models from metric data acquired from existing information (planimetry, elevations, etc.) or through survey activities with tools capable of generating point clouds of the structure. The former showed a lack of information and metric details capable of describing even complex surfaces. The latter, on the other hand, presented themselves as very heavy datasets from the point of view of storage and therefore difficult to

manage for all users. Therefore, starting with the construction of 3D models acquired efficiently and with a high degree of metric rigour, it will be possible to create models that can be interrogated, both metrically and for the search of semantic information.
In addition, it is possible to generate a 3D point cloud model by integrated survey techniques based on the use of active and passive sensors and several platforms (terrestrial, UAV and aerial).

The aim is to identify a suitable methodological process through the application of algorithms and appropriate processing and parameterisation steps, such as to generate light, metrically valid and semantically rich models, applicable to any acquired dataset. The metric parameterisation of the objects of a structure involves three fundamental steps which, starting with the planning and realisation of the survey, allows for three-dimensional modelling of the acquired data which, in turn, represent the basis for the subsequent realisation of models that can be easily imported and managed on BIM (Building Information Modeling) and/or FEM (Finite Element Analysis) software.

The transformation of the 3D model into parametric form requires the construction of regular surfaces from the point cloud. However, the surfaces that most software is capable of generating do not have regularity criteria, and it is necessary to set up appropriate processes and algorithms capable of correctly structuring the geometries of the object under investigation.

In this context, the processes to be adopted involve the application of simplification and surface regularisation algorithms and the identification of all those features necessary to make such a model implementable and manageable in a BIM/FEM environment. This means analysing complex elements, classifying them and managing the heterogeneous information related to them.

The research is based on the development of advanced functions, also with multi-temporal character, for the automatic or semi-automatic conversion from Point Cloud models to digital BIM/FEM models in the field of application of certain types of structures belonging to the cultural heritage.

This document is made up of chapters organised sequentially, starting with the description of data acquisition methodologies, through to the experimentation required

to produce 3D models that can be managed in a BIM and/or FEM environment. For each chapter, the state of the art is also described and analysed in relation to the topic addressed.

Chapter 1: Sensors used for the reconstruction of 3D models and their surveying techniques are described. In addition to traditional sensors, new types of sensors are described and analysed, such as the low-cost sensors implemented in the latest multimedia devices. The entire sensor technology is described by evaluating its applicability and highlighting its advantages and disadvantages.

Chapter 2: describes the Computer Vision techniques and the SfM approach for 3D reconstruction of the point cloud using the photogrammetric survey approach co. Furthermore, starting from the point cloud, the main processing processes that allow the generation of 3D surface models are described.

Chapter 3: Starting from the techniques described in the previous chapter, commercial and Open-Source software, capable of performing computing processes, are briefly analysed, reporting advantages and disadvantages, fields of application, as well as integration between different software and possible limitations.

Chapter 4: The objective of this chapter is the theoretical description of the different methods of reconstructing polygonal 3D surfaces, starting with the point cloud. The topological and representational aspects are illustrated and described, as well as the 3D reconstruction algorithms, up to the definition of polygon meshes and NURBS, which are the focus of this paper.

Chapter 5: This chapter describes an application that, starting from an integrated photogrammetric and TLS survey, analyses the processes of polygon mesh reconstruction and NURBS construction, up to their implementation, management, and interoperability in the BIM environment.

Chapter 6: This chapter describes an application of terrestrial photogrammetric surveying applied to a bronze statue and analyses the processes for reconstructing NURBS geometries obtained from appropriately simplified and reworked quad-mesh models for correct handling in FEM software. In particular, to test the repeatability of

the method, similar structures but with different complexities and morphologies are analysed, both in terms of geometry and construction material.

The paper includes a final chapter describing the objectives achieved and possible future developments of the research with a view to improving and fully automating the proposed approach.

1. SENSORS AND SURVEYNG TECHNIQUES FOR 3D POINT CLOUD RECONSTRUCTION

The continuous evolution of sensor-based survey techniques and the development of increasingly high-performance systems for digital data visualisation highlight the added value of their use in the field of 3D reconstruction of objects. In particular, the active contribution that these technologies can provide in the interpretation phase, in the preservation and archiving of data and in valorisation, leading to a general improvement in the quality of representation and information of each asset (Guidi et al., 2010), is becoming increasingly clear.

In order to reconstruct a 3D model, an important role is played by both the type of sensor used and the different digital surveying techniques and methodologies.

In this context, by adopting one of the different types of existing sensors, the result of the three-dimensional survey of a given object is the 'point cloud', i.e. a set of millions of points in a three-dimensional space, characterised by a series of information such as position (x, y, z coordinates), colour (RGB tern), intensity and another series of scalar lengths, referring to a given coordinate system.

The characteristics of geometric resolution, metric accuracy and definition of a point cloud are a function of the type of sensor used.

In general, three-dimensional sensors are instruments that generate a 3D image of the scene they frame. Depending on the nature of the light that is used, if it is natural light, the measurement methods are said to be 'passive' (cameras, scanning instruments, thermal imaging cameras, etc.); if, on the other hand, the light is encoded in such a way that it plays a role in the measurement process, we speak of 'active sensors' (laser scanners, structured light projection instruments, radar, etc.).

The choice of the sensor to be used and the surveying techniques to be adopted are a function of the characteristics of the surface to be surveyed, as well as a function of the required accuracy and geometric resolution, size and spatial location, in addition to a number of other factors such as experience and project costs.

In addition, for the creation of 3D reality-based models, different types of sensors from those described above can be used, integrating the outputs of the individual survey processes in an efficient manner. The following pipeline (Fig. 1.1) schematises the different types of sensors used in the field of three-dimensional surveying.

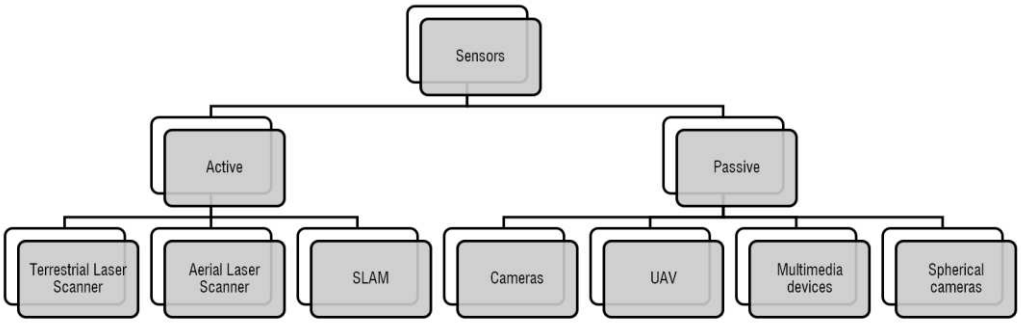


Fig. 1.1 - Schematisation of sensors used in 3D surveying.

1.1. Active Sensors

1.1.1. Terrestrial Laser Scanner (TLS)

An active sensor basically measures distances and is able to directly return a large number of 3D coordinates of the measured surface. The active sensors most commonly used in the field of 3D reconstruction are laser scanners (ground or airborne) and fringe projection systems (or structured light).

In general, these measuring instruments consist of a light emitter and a sensor to record its response. In fact, laser light, due to its physical properties, can generate bright spots even at long distance intervals.

Depending on the type of light emission and projection, it is possible to have a single laser spot, a moving laser light blade or a pattern of light projected at different resolutions. In the first case, a thin beam of light is generated that produces a bright dot on the surface to be measured. The camera sensor (e.g. digital video camera) is moved laterally to the light source in such a way as to frame the light spot; the light source, the projection centre of the sensor and the light spot form a triangular geometry.

Through the knowledge of a series of a priori known parameters such as baseline, angle and focal length, and with the help of trigonometric relations, it is possible to determine the distance between the instrument and the object and consequently the spatial position of the spot on the object.

The principal scheme illustrated above can be extended from a single point of light to a set of aligned points forming a segment (or blade). Active systems of this kind use a blade of light generated by a laser reflected from a rotating mirror or cylindrical lens and recorded by a digital camera offset from a base. For each point forming the light profile, the three-dimensional co-ordinates corresponding to the illuminated segment are obtained in one go, similarly to the single spot sensor. In order to obtain a three-dimensional image relative to a portion of an object, it is necessary to move these active sensors in a mechanically controlled manner, which leads to the concept of scanning, resulting in a range map relative to a rectangular area whose size depends on the extent of the movement. In the case of active structured light sensors, a special projector is used instead of the laser light emitter that emits structured light, while the image of the surface invested by this light pattern is acquired by a digital camera, offset by one base.

In summary, triangulation systems allow high measurement performance, with resolutions down to a tenth of a millimetre and a measurement uncertainty of a few tens of microns. On the other hand, they are systems that work at limited distances (within 2 m) and with limited framing ranges. For this reason, for the survey of larger objects such as architectural structures, archaeological excavations, statues or even entire areas, equipment is used that is capable of measuring the pointing direction with great precision, to which is added a distance meter that involves sending a laser pulse and the subsequent measurement of the time taken by the light to go to the surface and return to the measuring instrument (TOF - Time Of Flight). This time interval makes it possible to trace the distance between the instrument and the surface, which, integrated with the knowledge of the angles, makes it possible to define the 3D coordinates of the area targeted by the laser. A time-of-flight scanner automates the pointing operation

using a motor that automatically changes the pointing direction over a suitably programmable zone.

The problem associated with systems of this type concerns the accuracy of acquisition; in fact, the high speed of light produces a randomness in the estimation of the time of flight, which means an accuracy of up to 2 mm, approximately 20 times greater than triangulation systems that guarantee an accuracy of a tenth of a millimetre. This is why time-of-flight laser scanners are used in contexts where the objects to be detected are large and, where a millimetre uncertainty is acceptable.

The functional scheme of a time-of-flight scanner involves a moving light source and a detector embedded within the same unit. Three main strategies can be used to measure 3D coordinates by exploiting the propagation delay the sending of light pulses or Pulsed Wave (PW) to the scene, and the direct measurement of the delay between transmitted and received pulse (time of flight); the sending of a fixed frequency continuous light radiation or Continuous Wave (CW) and the indirect measurement of the time of flight, through the estimation of the phase shift between transmitted and received signal; the sending of a variable-frequency continuous light radiation or Continuous Wave-Frequency Modulated (CW-FM) and the indirect measurement of the time-of-flight, through the set of phase shifts between transmitted and received signal at different frequencies, and the corresponding amplitude alterations (Russo et al., 2011).

If through the PW category of instruments it is possible to measure even very long distances by limiting the acquisition speed and precision (varying from a few decimetre to a few millimetre), through the CW category of instruments, it is possible to improve both the precision (below one millimetre) and increase the acquisition time (one million points per second), limiting the distance related to the wavelength of the light signal (tens of metres), The third category of CW-FM instruments is very sophisticated and requires a level of signal processing that makes it extremely expensive. The main advantage lies in the very high accuracy obtainable, better than a tenth of a millimetre. On the other hand, the system is rather slow while maintaining a working range limited to a few tens of metres.

In general, laser scanners based on the time-of-flight principle can be used both for terrestrial applications and from aircraft or helicopters for land surveying, as opposed to triangulation laser scanners that are used strictly for terrestrial applications due to their limited working distance (Costantino et al., 2013).

1.1.2. Aerial Laser Scanner (ALS)

Systems based on the principle of distance measurement are generally referred to as Laser Radar or LiDAR, an acronym for Light Detection And Ranging, which stands for a pure distance measurement function using light.

In the current understanding, this definition has been improperly attributed to a particular category of laser scanners, those mounted on aircraft, whose more correct definition is Airborne Laser Scanner (ALS). The first LiDAR remote sensing systems were used by NASA in the 1970s for mapping Arctic and Antarctic ice-covered areas, but despite a number of remote sensing research projects during the late 1970s and 1980s, adoption in disciplines other than geomatics did not occur until the 1990s.

It was not until the 1990s, with the improvement of GPS systems and their combination with Inertial Navigation Systems (INS), that we saw the growth of ALS systems and their use to survey the territory quickly and accurately.

In this case, the system consists of a laser emitter/receiver that allows the measurement of the distance between the device and the ground and an integrated system consisting of a GNSS (in differential mode) and an inertial platform (INS), which allows real-time positioning and orientation in an absolute reference system of the 3D acquisition instrument during the measurement.

Knowing the time elapsed between the emission of the pulse and the return, it is possible to trace the distance of the hit point in the terrain (Vosselman et al., 2010). The most recent ALS systems make it possible to record different reflections of the emitted signal, thus making it possible to discriminate the detected surface in terrain, vegetation, urbanisation, etc. Depending on the flight altitude and the scanning system used, an ALS survey can provide point clouds with densities ranging from one point to 15-20 points per m^2.

In addition to providing dense data over large areas, LiDAR differs from other remote sensing technologies in its ability to provide data from the ground and objects even under forest cover. When a LiDAR sensor is flown over an area of forest or scrubland, some signals penetrate the vegetation canopy, reach the ground, and are recorded again by the sensor, effectively enabling the creation of terrain models (Kangas et al., 2018). In fact, such devices are widely used mainly for the production and return of digital terrain models (DTM) and surface models (DSM) of large areas. Most LiDAR systems emit light in the near-infrared spectrum (Near InfraRed or NIR) and are therefore affected by atmospheric conditions such as clouds, snow, rain or humidity. ALS sensors produce a dense point cloud of the object: the greater the density of the points, the greater the level of detail of the object being examined. The parameters that determine the point density of the cloud, and consequently the planning of the ALS flight, are different, among them is the scanning frequency, i.e. the number of pulses or beams emitted by the laser instrument in 1 second. ALS sensors have different scanning mechanisms, i.e. oscillating, sinusoidal, fibre scanner, rotating polygon, etc. Each scanning mechanism produces a different spatial distribution of points.

The most common configurations of the spatial distribution of points are shown below and in Fig. 1.2 (Čekada et al., 2010):

- sawtooth model;
- equivalent stabilised sawtooth model (sinusoidal);
- parallel line diagram;
- elliptical scheme;

In the saw model and the stabilised saw model, the pulse is directed through the scan area by an oscillating mirror, returns are generated continuously in both scan directions. Although this configuration is designed to preserve the spacing between the returns, the pulse density is not uniform. In the parallel line model, a rotating polygonal mirror directs the pulses along parallel lines through the swipe, and data are generated only in the scan direction. The elliptical model is achieved by means of a rotating mirror rotating about an axis perpendicular to the plane of rotation.

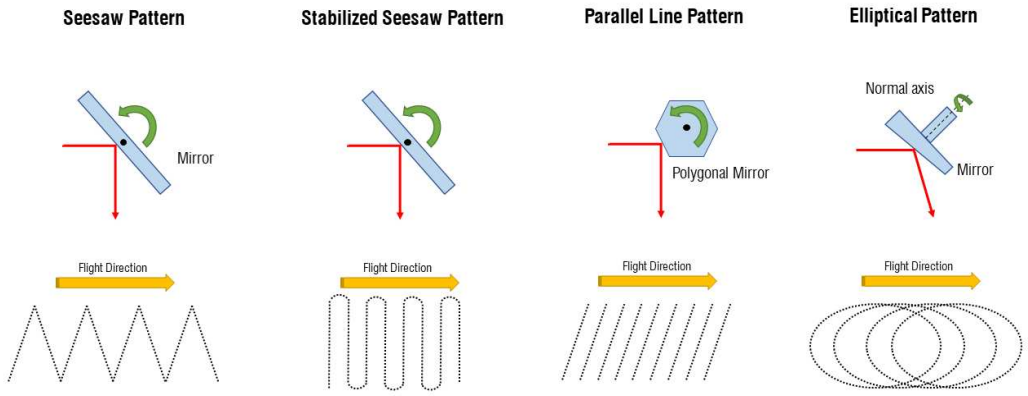


Fig. 1.2 - Spatial distribution of points.

Recent developments in laser scanner technology have resulted in a system that provides three scanning patterns (seesaw, stabilised and parallel), all in a plane nominally orthogonal to the scanner's longitudinal axis and nominally centred on the nadir; each scanning mechanism corresponds to a specific formula for calculating point density.

A further parameter affecting spot density is the beam divergence, i.e. the increase in beam diameter that occurs as the distance between the laser instrument and the acquisition plane increases. Typical beam divergence settings range from 0.1 to 1.0 millirad. This parameter is very important in ALS surveying because it determines the level of penetration of the pulse into forests and other vegetated areas, the level of detail that can be detected, the sharpness with which the contours of buildings and other objects can be recorded and the level of risk (such as to eyes).

The scan angle is the angle at which the axis of the light beam is in relation to the 'focal' plane of the LiDAR instrument. Sometimes, similar to photogrammetric terminology, this angle is called FOV (Field Of View). In order to increase the point density when constructing the flight plan, it is necessary to change one or more of the above-mentioned parameters. There are two other ways to increase point density:

- by designing a suitable 90° cross-strip flight plan. This approach is especially popular in surveys of dense urban areas or areas of special interest where additional detail of spatial objects is required.
- through a special LiDAR technology. called Multiple Pulses in Air Technology (MPiA) (recently developed) that allows a higher pulse frequency to be used, whereby the ASL system fires a second laser pulse before receiving the reflection of the previous pulse, so that the pulse frequency at a given flight height can be doubled. In addition to the MPiA system, density-enhancing technologies have been developed, such as the "Continuous MultiPulse" (CMP), "Multiple Time Around" (MTA), in which the laser rangefinder can fire a new pulse towards the ground without having to wait for the reflection of the previous pulse to arrive at the instrument (Pepe, 2017).

1.1.3. SLAM devices

Simultaneous localisation and mapping (SLAM), also known as Concurrent Mapping and Localisation (CML), is one of the techniques that have arisen due to the advent of robotics, which consists of simultaneously constructing a map of the environment and determining the position of the device within it (Aulinas et al., 2018).

In general, there are two types of technology components used to realise SLAM. The first type is the processing of sensor signals, including front-end processing, which is largely dependent on the sensors used. The second type is pose graph optimisation, including back-end processing, which is independent of the sensor. To delve into the front-end processing component, it is necessary to examine visual SLAM and LiDAR SLAM separately, i.e. the two different SLAM methodologies.

Visual SLAM (or vSLAM) uses images captured by cameras and other image sensors and can use simple cameras (wide-angle, fish-eye and spherical cameras), compound-eye cameras (stereo and multi-point cameras) and RGB-D cameras (depth and ToF cameras). Furthermore, it can be implemented at low cost with relatively inexpensive cameras. As cameras provide a large volume of information, they can be used

to detect reference points (previously measured positions). The detection of such points can also be combined with graph-based optimisation, achieving flexibility in SLAM implementation.

If the vSLAM method uses a single camera as the only sensor, one then speaks of a monocular SLAM, but with this method it is difficult to define depth. To overcome this problem, the use of AR markers, chessboards or other known and useful objects for localisation or data fusion techniques are used to integrate information from other sensors such as the IMU (Inertial Measurement Unit), which are able to measure physical quantities such as speed and orientation. Technology related to vSLAM includes structure from motion (SfM), visual odometry and beam steering.

Visual SLAM algorithms can be classified into two categories: sparse methods that match the characteristic points of images and employ algorithms such as PTAM and ORB-SLAM, and dense methods that exploit the overall brightness of images and employ algorithms such as DTAM, LSD-SLAM, DSO and SVO.

1.1.4. SLAM LiDAR

With regard to SLAM LiDAR technology, laser sensors are used for surveying. Compared to cameras, ToF and other sensors, lasers are considerably more accurate and are also used for applications with high-speed moving vehicles such as self-driving cars and drones. Output values from laser sensors are generally 2D (x, y) or 3D (x, y, z) point cloud data. The laser sensor point cloud provides high-precision distance measurements and is very effective for map construction. Motion is estimated sequentially by combining point clouds. The calculated movement (distance travelled) is used to locate the vehicle. Registration algorithms such as Iterative Closest Point (ICP) and Normal Distributions Transform (NDT) are used for matching LIDAR point clouds. Maps of 2D or 3D point clouds can be represented as a grid map or a voxel map.

One problem with this method of acquisition is that the point clouds may not be very detailed in terms of density, and this can cause uncertainty in the alignment processes with a consequent loss of position of the carrier or the vehicle transporting it. Furthermore, the alignment between different scans requires generally high

computing and processing power, which is why it is necessary to optimise the processing to improve speed. Although SLAM technology is used for some practical applications, there are several technical challenges that prevent its more widespread adoption. Some of the problems with this technology are:

- Errors, deviations, and distortions in the acquired data.
- Localisation errors.
- Computational aspects.

In the first case, the SLAM estimates sequential movement, which includes a certain margin of error. The error accumulates over time, causing significant deviation from actual values, can also cause the map data to collapse or distort, making subsequent searches difficult. Position estimation errors such as these are unavoidable. It is important to detect loop closures and determine how to correct or eliminate the accumulated error. By solving error minimisation as an optimisation problem, more accurate map data can be generated. This type of optimisation is called beam adjustment in the visual SLAM.

In the case of localisation, the mapping of images and point clouds does not take motion characteristics into account. In some cases, this approach can generate discontinuous position estimates. This type of localisation error can be avoided by using a retrieval algorithm or by merging the motion model with multiple sensors and performing calculations based on the sensor data. There are several methods for using a motion model with sensor fusion. the most common method is to use Kalman filtering for localisation, more flexible Bayesian filters and unscented Kalman filters can also be used (Hoxhaj et al., 2008). Some commonly used sensors in this case are inertial measurement devices such as IMU, AHRS (Attitude and Heading Reference System) or INS (Inertial Navigation System), accelerometer sensors, gyroscopic sensors, and magnetic sensors). In the case of MMS (Mobile Mapping System) applications, wheel encoders attached to the vehicle can be used for odometry.

If the localisation is unsuccessful, it can be recovered through feature extraction processes so that a high-speed scan can be performed. Some methods based on image features include BoF (Bag of Features) and BoVW (Bag of Visual Words). More

recently, a methodology used is related to the Deep Learning approach and algorithms for comparing distances from features.

Computational aspects are those related to processing such as the localisation and processing of images and the extraction of high-density point clouds. In addition, optimisation calculations such as ring closure are computationally intensive processes. This can be mitigated by using multicore CPUs for processing, embedded GPUs for speed or SIMD (Single Instruction Multiple Data) computation.

1.2. Passive sensors

Sunlight is the main source of radiation measured by a passive sensor that collects the radiation emitted and reflected by the object or surface under investigation. Passive sensors include optical sensors, more appropriately transducers, which detect light rays and transform them into electronic signals. An optical sensor generally consists of a light source (e.g. LED) and a receiver (e.g. photodiode): the measurement takes place when the object interrupts or reflects the amount of light emitted. Optical sensors are also image sensors, i.e. capable of converting an image into an electrical signal by focusing it on a grid made up of many small point sensors that one by one detect light and convert it: the best known technologies are CCD (Charged-Couple Device) and CMOS (Complementary metal-oxide semiconductor) integrated circuits.

1.2.1. Terrestrial camera sensors

Terrestrial photographic sensors, or more commonly photographic cameras, capture light on an electronic medium; this sensor numerically encodes the acquired information into a matrix of luminance values and stores the processed image on a memory device. Three main categories of cameras are currently available on the market: compact, bridge, mirrorless and SLR.

In the first case, everything is concentrated in a single device that is usually small and lightweight. The settings of the acquisition parameters are fully automatic, and the framing and display is done by means of screens that allow the scene to be viewed as it will be recorded on the image file.

Mirrorless cameras are cameras with interchangeable lenses that are replacing SLRs around the world. The lack of the mirror mechanism and pentaprism allows for a considerable reduction in size and updated functions. Bridges are amateur cameras that bridge the gap between compact and interchangeable lens cameras (SLR or mirrorless). They have the advantages of the former, i.e. ease of use, fixed optics, and portability, and the advantages of the latter, i.e. image quality and above all availability of wide zoom ranges.

The dSLR, finally, is a camera with which framing is done directly through the lens thanks to a system of mirrors: the screen only serves to display the images once captured. When taking photos, the mirror is raised allowing the light to pass through to the sensor. The latter model, which is considered fully professional, allows interchangeability of lenses as well as control over advanced functions such as exposure and focus.

For surveying purposes and for photo-modelling requirements, the latter is the most suitable solution precisely because of the possibility of interchanging lenses according to the type of restitution, the control of advanced parameters, the size of the object to be reconstructed and the photographic shooting conditions.

The lenses used in these types of cameras consist of thin lenses capable of introducing insignificant spherical aberrations, thus making it possible to obtain from a subject, an image whose sharpness is the theoretical maximum and is limited only by the wavelength of the light used. Lenses are characterised by two parameters: focal length and aperture. The focal length is the focal distance at which the image of a subject is focused at an infinite distance. If light is allowed to pass through a lens, from the opposite side the image will be focused at a focal distance equal to the focal length. If the subject is not placed at infinity, the further away from the lens its image will be formed. If S_f is the distance of the subject in focus and I_f denotes the distance at which the image is formed beyond the lens, i.e. the focal distance, and with L the focal length, it is possible to determine:

$$\frac{1}{S_f} + \frac{1}{I_f} = \frac{1}{L} \tag{1}$$

which relates the focal length L with the distances of the subject and its image from the lens. In theory, the minimum distance the subject can reach in relation to the lens and which allows an image to form is the focal length, but this would result in an image projected to infinity. In practice, the minimum focusing distance is dictated by the maximum excursion of the lens relative to the sensor. The further the lens can move away from the sensor (the greater the I_f) the closer the subject can be (the smaller is S_f). The other parameter that characterises a lens with its complete optical scheme and its diaphragm (relative aperture), also expressed by the symbol $f/$ which represents the ratio between the focal length and the aperture diameter.

$$f/ = \frac{L}{DD} \tag{2}$$

where:

L focal length;

DD represents the diameter of the diaphragm.

For example, a lens with a focal length of 105mm, and with a relative aperture set to f/2.8 will have an aperture of $105/2.8 = 37.5$mm in diameter.

With the same focal length and subject distance, the optical magnification ratio does not change. What does change is the angle of field visible through the two sensors (Fig. 1.3).

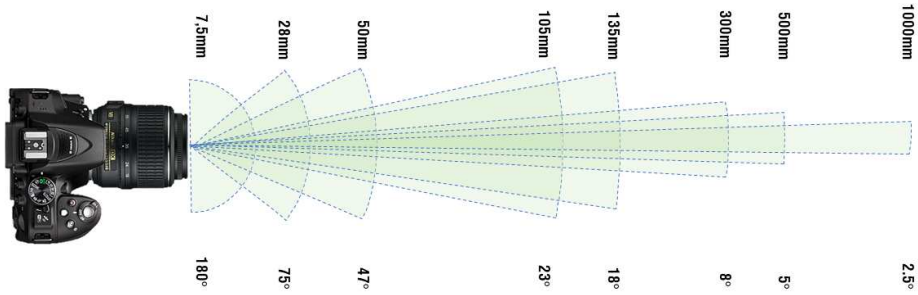


Fig. 1.3 - Angle of field and focal lengths.

The field of view (FOV) can be calculated as a function of focal length L and the size of the diagonal D_s of the sensor. The field of view A_c (expressed in degrees) is:

$$A_c = 2 \cdot \arctan\left({}^{D_s}/_{2L}\right) \cdot {}^{180}/_{\pi} \tag{3}$$

Once established that the optical magnification (i.e. that due to the focal length of the lens and the distance of the subject, and not the size and resolution of the sensor) depends only on the focal length of the lens, the 1:1 magnification ratio will only be achieved when the distance of the subject S_f and the focal length I_f are equal. Recalling that S_f e I_f are related to each other, defining U as the value that S_f e I_f reach when they are equal, one can find this value as a function of focal length:

$$\frac{1}{U} + \frac{1}{U} = \frac{1}{F} \rightarrow \frac{2}{U} = \frac{1}{F} \rightarrow U = 2 \cdot F \tag{4}$$

i.e. a 1:1 magnification ratio is obtained when the subject, and consequently its image, is at a distance from the lens equal to twice the focal length.

Under the assumption of an ideal lens, i.e. one capable of focusing an image without introducing any distortion or aberration and producing an extremely sharp image, the maximum resolution obtainable will be that of the pixel size. If two bright dots or only one of the same intensities fall on a pixel, the signal produced by the pixel will be exactly the same.

By increasing the size of the details or, rather, by moving two bright points on the subject away from each other, producing two image points on the sensor, initially at a distance of less than the pixel size and then gradually increasing in distance, two distinct pixels will be seen in the digital image produced by the sensor. These two pixels cannot be adjacent, otherwise it cannot be determined whether the subject points were themselves adjacent or distinct but too close to be distinct. Ultimately, at least three pixels are needed to distinguish two points. In order to distinguish a series of points, there must be at least two pixels for each element. In signal theory, this theoretical limit is called Nyquist's Sampling Theorem: to represent a regular structure consisting of N repetitions, at least 2 times N pixels are therefore required.

It is difficult to achieve this definition and the first cause is the presence of the aliasing filter in front of the sensor. This filter (sometimes called low pass, because it cuts off spatial frequencies above the Nyquist limit, thus preventing aliasing) means that the minimum detail requires at least three pixels to be captured and distinguished. Anything below three pixels begins to be fuzzy, indistinct.

The need for this filter also derives from the way in which the three colour components of a pixel are generated from the Colour Filter Array, or Bayer matrix. The Bayer matrix determines, in the RAW/NEF format, the recording of only one colour component for each pixel, delegating to adjacent pixels the task of providing the other two missing colour components to arrive at the RGB mode. In order to have adequate colour consistency in inheriting colour components from adjacent pixels, it is necessary that upstream each point of the image is also diffused (blurred) on the two adjacent pixels (horizontally and vertically).

After all these considerations we can reasonably define an areola of three pixels in diameter below which details are no longer clearly distinguishable, this areola is called a circle of confusion.

If we do not consider the lens as an ideal object, then the sharpness and sharpness of the lens itself usually expressed with the Modulation Transfer Function "MTF" (Alfio et al., 2020) must also be taken into account. By means of advanced algorithms, it is possible to calculate MTF values for each edge; tools integrating these algorithms automatically convert the image to be analysed into a 16-bit greyscale and measure the values of the modulation transfer function, concluding that the greater the depth of field, the worse the MTF, i.e. ultimately that the spatial sampling step of the image is greater than the blurred detail.

1.2.2. UAV camera sensors

In recent years, progress and technological innovation have enabled the development of numerous types of Unmanned Aerial Vehicle (UAV) equipped with good quality and affordable cameras. These remotely piloted systems have in fact, for some applications, replaced classic aerial (or helicopter) photography. Through such

systems, it is possible to photograph from above and carry out a series of photogrammetric missions, thanks also to the possibility of having appropriate applications that make the acquisitions automatic. In addition to the possibility of equipping these systems with reflex or mirrorless cameras, the market also offers UAVs with integrated, high-performance cameras. In fact, these are almost always cameras suitable for both capturing digital images and shooting video, and their cost is a function of the features and performance of the photographic compartment. The main characteristics that distinguish UAV cameras are the size of the sensor and the possibility of capturing images in RAW or DNG format. If the resolution of the sensor makes it possible to obtain images of excellent quality, the possibility of capturing images in RAW or DNG format is particularly important as it allows for post-production modifications and a significant improvement in the dynamic range, compared to a JPEG file.

From a technical point of view, however, in integrated cameras, one does not have the possibility of managing optics or shooting parameters as one does on sensors such as SLRs or mirrorless cameras. The cameras on drones, in fact, are basically comparable to those mounted on smartphones, so the changes to aperture, timing and focal length are not optical, but software.

Since we are dealing with photos from above, in which the elements of the scene being photographed may be small, it is necessary to set and configure several parameters that will enable high-quality images to be obtained. The main factors to be set correctly are listed below.

- Aperture: Variable between different camera models, it offers better sharpness with values around f/5.6;
- Shutter speed: this parameter is highly dependent on the stability of the drone and the anemometer condition during acquisition (generally recommended values are above 1/50s).
- ISO sensor sensitivity, in cameras integrated on drones the sensors are extremely miniaturised and therefore do not allow high ISO values to be set without generating digital noise (standard values are below 400 ISO).

- White Balance: this parameter becomes unmodifiable when capturing in JPEG format, on the contrary, when capturing in RAW or DNG format, this can be changed in post-production.

Generally, to avoid problems, the AWB automatic mode can be used.

In order to maintain the tilt level and eliminate the problem of vibrations, all modern drones with a built-in camera are equipped with a device called a 3-axis Brushless Camera Gimbal. The gimbal of a quadcopter is an anti-vibration 3-axis camera stabiliser. It uses brushless motors (powerful, quiet and long-life) to adjust the position of the camera in all directions: up/down, left/right and forward/backward (3 dimensions or XYZ). There are two types of gimbals:

- as a separate unit on which an external camera (GoPro, for example) is mounted.
- As a complete unit with a built-in camera (drones with a built-in camera).

The most important mechanical parts of a gimbal (Fig. 1.4) are the camera mount and the 3 motors that work together to maintain the tilt level and to eliminate vibrations (Pepe et al., 2020). These new electromechanical devices are known by the acronym MEMS (Micro-Electro-Mechanical Systems) and are composed of more specific sensors such as IMUs, accelerometers and gyroscopes; their function is to receive a mechanical force and then convert it into an electronic signal that is processed by a computer. Most electronic components have no moving parts, they are purely electrical, with only electrons flowing through the labyrinth inside the chips. MEMS have structures that move and respond when pushed in one direction or another (g-force, gravity, etc.). A lot of computing power is required to operate the gimbal, which is often enclosed in a circuit board called a controller. This board also contains the MEMS and the firmware (embedded software), which is created directly by the drone manufacturer. This controller sends commands thousands of times per second to the 3 brushless motors that stabilise the camera. These commands are used to maintain the level of the camera and can be used to remove most of the vibrations caused by the propellers and motors of the drone itself.

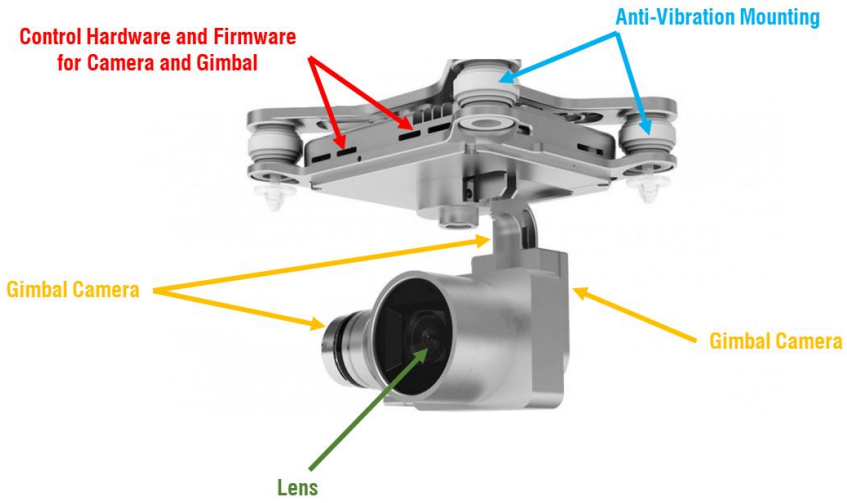


Fig. 1.4 - Cardan joint and mounting a camera.

1.2.3. Spherical cameras

In recent years, omnidirectional cameras, i.e. cameras with a field of view (FOV) of 360° in the horizontal plane, have become increasingly popular. Depending on the structure of the omnidirectional camera, it is possible to obtain 3 types of sensors:

- diopters, which use a combination of shaped lenses (e.g. fisheye lenses);
- catadioptric, which combines a standard camera with a shaped camera (parabolic, hyperbolic or elliptical);
- polydirectional, using several cameras with overlapping FOVs.

In images generated by these cameras, distortion increases non-linearly from the centre to the sides of the images. The equirectangular projection is generally used to represent the captured scene. In fact, captured images are projected onto a virtual sphere, which in turn is mapped onto a plane according to the “azimuth-zenit” or equirectangular projection, i.e. latitude-longitude, also called spherical panorama.

From a mathematical point of view, considering a sphere of radius r, spherical coordinates, latitude ϕ and longitude ϑ, this can be projected onto the map plane of x, y coordinates using the following relations:

$$x = r \cdot \vartheta$$
$$y = r \cdot \varphi \quad (5)$$

Therefore, the height of the map is equal to the development of a meridian (πr) while the base is equal to the circumference of the sphere (2πr), as shown below (Fig. 1.5).

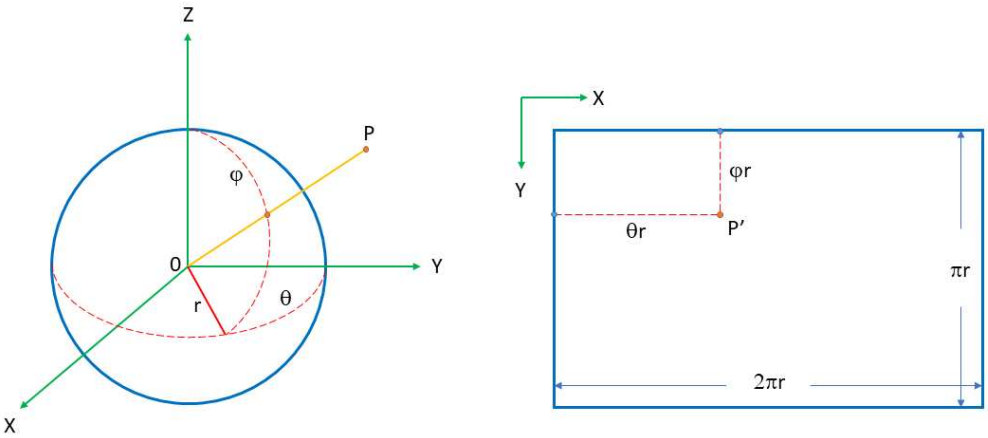


Fig. 1.5 - Relationship between spherical coordinates and image coordinates in the equirectangular image.

In fact, thanks to evolutions in surveying techniques and in particular spherical photogrammetry, it is now possible to use panoramic images or photos (spherical projections up to 360°) not only for classic monoscopic VR navigation (QTVR type) but it is also possible to georeference images, extract 3D geometries of the scene and above all project high-resolution textures onto model surfaces.

New 360° cameras are currently available on the commercial market, capable of capturing the entire surroundings in a single shot. Some 360° cameras are listed in the following Table 1.1 according to their resolution and relative cost. Therefore, the choice of camera comes from a trade-off between technical performance and price (Herban et al., 2022).

Tab. 1.1 - Relationship between spherical coordinates and image coordinates in the equirectangular image.

Product	Resolution [MP]	Cost [€]	Level
Ssstar	16	200	Entry
GoXtreme Dome	8	230	
Nikon KeyMission	23.9	280	
Xiaomi Mijia Mi Sphere	23.9	290	
GoPro Max	16.6	550	Medium
Ricoh Theta V	14.4	550	
Garmin VIRB	15	700	
Vuze+	14.7	990	
Ricoh Theta Z1	22	1,100	
Insta360 Pro2	59	4,000	High
iSTAR Fusion 360	50	5,900	
Insta360 Titan	55	17,000	
Weiss Ag Owl WAM2	230	Not available	

1.3. Multimedia devices

The use of digital images obtained from low-cost sensors for photogrammetric purposes has become increasingly common due to the recent development of software based on Structure from Motion (SfM) and Multi View Stereo (MVS), which allow three-dimensional models to be obtained. In particular, the SfM approach requires, for camera models based on collinearity equations, a minimum of three corresponding points per image captured at convergent angles. Consequently, it is necessary to capture images with a high overlap to cover an object or structure. The MVS approach allows the point density to be increased from the point cloud generated by the SfM process. In this way, a dense point cloud of an object or structure is obtained. This approach can also be used in low-cost cameras, which are characterised by a lower geometric stability than professional cameras, resulting in lower reliability of the cameras' internal geometries (Costantino et al., 2020). To remedy this, rapid calibration algorithms and

procedures have been developed in recent years that also allow the use of these cameras for 3D measurement purposes. After estimating the intrinsic parameters of the camera, such as the focal length (f) of the lens, the coordinates of the lens centre, the coordinates of the image projection centre and the radial distortion coefficients of the lens, a 3D model can be produced.

Various sensors are available on the market, such as Compact System Cameras (CSC), dSLR (digital Single-Lens-Reflex) cameras, bridge cameras, action cameras, but smartphones can also be a good choice as they can capture high-resolution images and most of them have a prime/fixed lens.

Actually, there are two main technologies that can be used for a camera's image sensor, CCD (Charge-coupled Device) and CMOS (Complementary Metal-Oxide Semiconductor). CCDs are sensors based on an array of passive photodiodes that integrate charge during the exposure time of the camera. The charge is then transferred to common electronics that reads the charges accumulated by the pixels and translates them into voltages. Since the CCD is a passive pixel device (i.e., without pixel-level electronics), its quantum efficiency is very high, making it an advantage in applications where light is rather scarce.

CMOS, on the other hand, are sensors based on a matrix of active pixels: the electronics at the pixel level (typically 3 or 4 transistors) translates the charge accumulated in the photodiode into a well-defined voltage; thus, the output of each pixel only must acquire and sample the output.

Although CMOS sensors were developed after CCD sensors, they soon gained a significant market share. Thanks to recent technological developments and their small size, CMOS detectors are integrated in all smartphones, which in turn are widely used in various applications, including photogrammetry for processing 3D models.

1.3.1. Smartphone with LiDAR and ToF sensors

Recently, smartphones equipped with depth sensors have come onto the market. These sensors were advertised as “LiDAR scanners” for iOS devices and “time-of-flight depth cameras” (ToF) for Android. The sensors were originally integrated to

improve the quality of photos (e.g. better focus, bokeh effect, etc.) and to enable augmented reality applications, but have also proven to be suitable for certain application purposes (Costantino et al., 2022).

As of 2016, the first smartphones to introduce depth sensors were Android devices, which were used for augmented reality applications. In order to improve these applications, thanks to a project funded by Google, a technology was developed based on three fundamental parts:

- Depth sensing.
- Movement mapping.
- Learning in the area.

The first part used an RGB-D sensor to estimate the depth of the images. The second part used inertial sensors (gyroscopes and accelerometers) while, the third part refined the position using simultaneous location and mapping (SLAM) technology. Despite the great potential of this technology, the project was abandoned in 2018 to be replaced with a new technology based on room depth sensing without active sensors (currently found in Android devices).

As far as the iOS market is concerned, in 2020 Apple released two devices (in mobile and tablet versions) equipped with LiDAR scanners, which are able to determine distances to and between objects with excellent accuracy, with the ability to reconstruct the detected object three-dimensionally. The LiDAR scanner measures the distance to surrounding objects up to 5 metres by taking a photon-level reading at nanosecond speeds. It is therefore a ToF technology that fires low-power lasers and using reflections calculates the distance to objects and points in the environment, creating a 3D depth map or accurate rendering. The depth frameworks in iPadOS combine the depth points measured by the LiDAR scanner with data from the two cameras and motion sensors, while also taking advantage of Computer Vision algorithms on the A12Z Bionic chip to more accurately detect details in the scene.

Raw data from LiDAR measurements are not directly accessible and can only be acquired via the API (Application Programming Interface) of the ArKit development library, which is dedicated to the design of iOS applications using augmented reality.

The functions provided by ArKit are very complex and jointly exploit both the cameras and the LiDAR sensor, essentially implementing 3D photogrammetry combined with laser measurement. The main functions that can be exploited for measurements are:

- "*sceneDepth*"*, which* creates a map of the distance between the objects in the scene and the iPad (or rather, between the scene and the cameras/LiDAR), this function forms the basis of many augmented reality applications.
- "*confidenceMap*" which associates each point on the distance map with an estimate of the accuracy of the data (accuracy is indicated on three values: low, medium, high).
- "*ArPointCloud*", which returns a cloud of points in the form of co-ordinates, which constitute the set of references useful for the device to recognise objects in a scene.

1.4. Survey techniques

Photogrammetry is a topographical technique that makes it possible to define the position, shape and dimensions of objects on the ground, using the information contained in appropriate photographic images of the same objects, taken from different points. This is possible thanks to the use of appropriate cameras that produce photographs (called photogrammetry to emphasise their metric value) on which precision measurements can be made, and which can be considered central views with sufficient approximation.

In general, photogrammetric surveys have several advantages such as the possibility of determining characteristics of objects without having physical contact with them, the rapid acquisition of information from many points (large masses of information), the repeatability of measurements as they are carried out a posteriori on the model, as well as rapid and inexpensive production of cartography and three-dimensional digital models, even of medium to large surfaces.

Photogrammetry is also classified according to the type of acquisition used, the type of processing and, finally, the type of photograph used, as summarised in the Fig. 1.6 below.

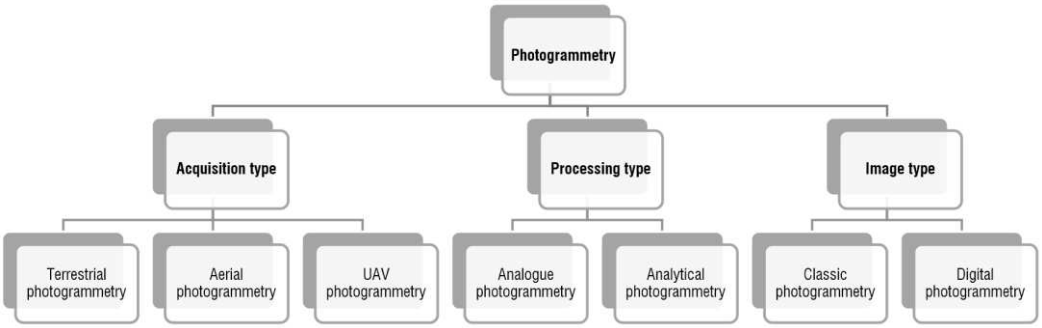


Fig. 1.6 - Classification of the photogrammetric technique.

In order to determine the positions of an object's points in the real environment using the positions of the corresponding points on the photograph, it is necessary to define the geometric relationships between the three-dimensional positions of the object points and those of their images on the plane of the photograph. Each three-dimensional object point (object space) then corresponds to a homologous point on the picture plane (image space). For this reason, it is possible to assimilate, with sufficient approximation, a photograph to a central perspective, on the basis of which the segments that connect the object points with their corresponding images (projecting star) all meet at a point O, a few centimetres away from the photographic plane, known as the centre of capture

The centre of capture O is a point on the camera lens, while the distance of O from the plate plane is called the main distance: the latter is kept fixed and can be considered (not considering distortion effects) equal to the focal distance of the same lens. The photogrammetric problem is solved by obtaining the coordinates of the intersection in space by solving a system of equations called 'collinearity equations' obtained from analytical geometry relations (analytical photogrammetry). In the past, the same operations were carried out with analogue mechanical (or optical) instruments, which made it possible to establish the position of the generic point A within a

reconstructed object space on a reduced scale known as a model (analogue photogrammetry).

In general, collinearity equations are the constitutive equations of photogrammetry, useful in the transition from image space to object space, and express the alignment between object point, image point and camera point. The most frequently used model is expressed by the *collinearity equations*, which, for a point of object co-ordinates (X, Y, Z) and image co-ordinates (x, y), take the following form:

$$x^0 = -c\frac{\hat{r}_{11}(\hat{X}-\hat{X}_0)+\hat{r}_{12}(\hat{Y}-\hat{Y}_0)+\hat{r}_{13}(\hat{Z}-\hat{Z}_0)}{\hat{r}_{31}(\hat{X}-\hat{X}_0)+\hat{r}_{32}(\hat{Y}-\hat{Y}_0)+\hat{r}_{33}(\hat{Z}-\hat{Z}_0)} \quad (6)$$

$$y^0 = -c\frac{\hat{r}_{21}(\hat{X}-\hat{X}_0)+\hat{r}_{22}(\hat{Y}-\hat{Y}_0)+\hat{r}_{23}(\hat{Z}-\hat{Z}_0)}{\hat{r}_{31}(\hat{X}-\hat{X}_0)+\hat{r}_{32}(\hat{Y}-\hat{Y}_0)+\hat{r}_{33}(\hat{Z}-\hat{Z}_0)} \quad (7)$$

where:

c	focal length;
$\hat{r}_{ij}$	element of rotation matrix which describes the rotations (ϕ, ω, κ) of the sensor with respect to the 3 main axes of the along-track system;
$\hat{X}, \hat{Y}, \hat{Z}$	object coordinates of the point;
$\hat{X}_0, \hat{Y}_0, \hat{Z}_0$	ground coordinates system.

Internal orientation parameters are commonly called the terms (f, x, y) and external orientation parameters the terms (X_0 , Y_0 , Z_0 , ϕ, ω, κ).

In digital photogrammetry, collinearity equations are used both on single frames and on images obtained from sensors consisting of linear arrays of CCDs. It should be noted that in the former case the equations are valid for the entire frame, while in the latter each scan line will have its own external orientation parameters.

1.4.1. Photogrammetric survey phases

The photogrammetric survey consists of the following phases:

- Acquisition that concerns the operations of taking photographic images, carried out with appropriate cameras and appropriate techniques.
- Orientation represented by the preliminary operations to determine the parameters relating to the centres of capture and the positions in space that the images had at the time of capture, thus reconstructing the shape and dimensions of the object captured.
- Restitution, i.e. the operations that allow measurements to be taken on the model of the reconstructed object, using restitutors or photogrammetric software, capable of producing, as an end result, a 3D model, a numerical set of coordinates or a straightened image.

For this last stage, the projecting rays of each frame must first be placed in the same position in space as they were when they were taken, these operations are provided for in the photogrammetric orientation stage. In general, photogrammetric orientation can be divided into two categories as follows:

- *Internal orientation*, i.e. the set of parameters that make it possible to reconstruct the metric of the central projection, hence of the projecting ray stars. They are the same for all the frames of the survey, as they are linked to the internal camera used in the capture and are provided by the camera manufacturer in special documents (calibration certificates).
- *External orientation* divided into relative and absolute. In the first case, the relative position of the two frames is determined, realising the intersection of the projecting rays, thus the three-dimensional model, but at an arbitrary scale and with a generic spatial location (thus released from the absolute reference system XYZ bound external to the surveyed object). In the second case, the model obtained in the previous phase is rototranslated and scaled, so that it is referred to the absolute reference system and scaled to the desired scale. At this stage, knowledge of the position of certain ground support points is required.

1.4.2. Photogrammetric acquisition techniques

To satisfy the basic principle of photogrammetry, i.e. to be able to obtain a 3D model from the frames taken, it is necessary for each generic point on the object or terrain to be taken from at least two consecutive frames. Depending on the type of survey to be carried out (Close Range Photogrammetry, Aerial or UAV Photogrammetry), there are suitable image-taking schemes or frames. In the case of aerophotogrammetric surveying, the aerial shot scheme consists of an aircraft (or UAV platforms) following a straight trajectory at constant speed, at a certain average height H above the ground (Fig. 1.7).

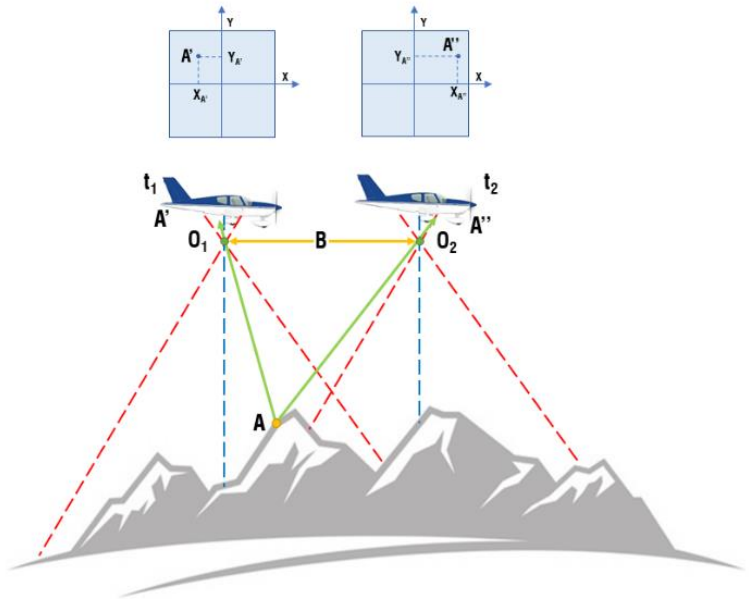


Fig. 1.7 - Scheme of nadiral aerial acqusition.

At instant t_1 the camera's pick-up centre is at O_1 and takes the first frame, and at instant t_2 (after having travelled a certain space that constitutes the pick-up base B) it will be at position O_2 where it will take the second frame. The aircraft flies over the ground repeating the previous pattern and performing the grasps within a sequence of side-by-side rectilinear paths along parallel directions.

The frames taken along the same straight path constitute a swipe. The set of several crawls is called a crawl block. In addition to the longitudinal (overlap) m (generally 70%), it is necessary that the spacing i, between two adjacent strips, is established so that there is also a lateral overlap f (sidelap) with a value in the range of 0.10-0.20 (10%-20%) of the frame's L embrace, in order to avoid having holes or a lack of information in the coverage of the terrain. Photographic coverage of the terrain must take place in such a way that the entire area to be surveyed is broken down into stereoscopic patterns such that every point on the terrain appears on at least two frames. This requirement is ensured by the longitudinal overlap of the L embrace of the frame; this produces two strips of overlap of 10% even in the stereoscopic models with respect to the adjacent models, which allows them to be chained together.

The parameters of an aerial photogrammetric survey (overlap, sidelap, etc.) are essential for correctly setting the flight planning; however, they must be considered as an outline programming of the flight itself. In fact, it is not possible to predict the exact unfolding of events throughout the flight, either due to the particularities of the terrain, which could cause the scale of the frame to vary, or due to the small but inevitable variations in the aircraft's speed, or the movements impressed on the aircraft by the winds. For the reasons previously described, it is not possible to entrust the acquisition operation completely to the automatisms with which the aerial cameras are equipped, to have specialised personnel on board the aircraft to manage the capture by control, and possibly correcting, certain critical aspects. Furthermore, if the terrain to be photographed is irregular and mountainous, a higher value of up to 80% must be adopted for the longitudinal coverage. Naturally, the number of frames required for the photogrammetric coverage of a certain portion of the terrain, due to the longitudinal and transverse overlap, is considerably higher than would be necessary for normal photographic coverage.

In general, photogrammetric flights are carried out under perfect weather conditions (no clouds, but also haze), and during the middle of the day, in order to take advantage of the maximum available illumination and to minimise the influence of

shadows on the ground. It follows that there are very few (on average 20-40) days in a year that are suitable for this activity.

Another important parameter to take into account is the ground sample distance (GSD), which is the distance between the centre points of two consecutive pixels of an image, measured on the ground.

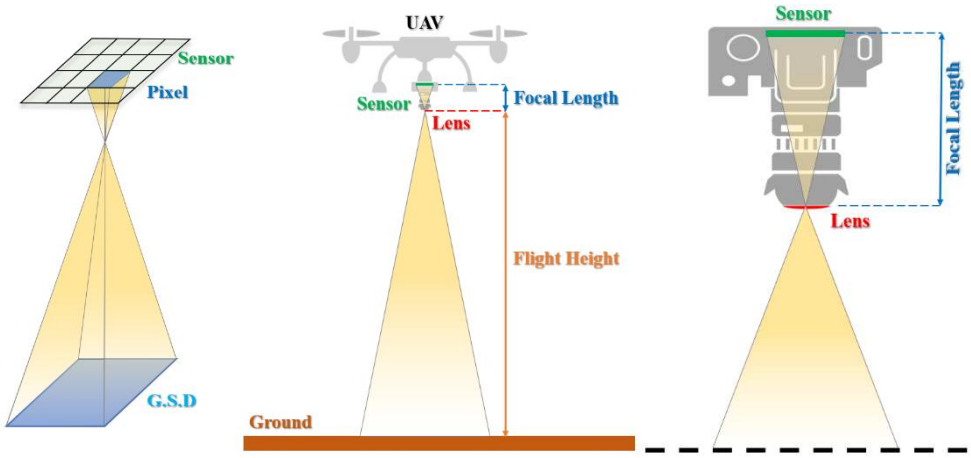


Fig. 1.8 - Schematisation and determination of the GSD.

Using a pinhole camera model (or sometimes called a projective camera model), i.e. a model that relates the physical image space to the physical work space using the projective transformation, it is possible to calculate the GSD value from the following relationship:

$$GSD = \frac{H}{f} s_x, \tag{8}$$

where:

H normal distance from the ground to the perspective centre,

f focal length of the camera and

s_x pixel size.

Graphically, these parameters are schematised in Fig. 1.8. For flight operations, it is necessary to calculate the absolute height above sea level (ASL) of the flight from

the knowledge of the mean height above ground (AGL), which means adding the height above ground (determined using regional and global digital terrain models) to that determined in relation to the GSD. The more accurate the digital terrain model is, the more the GSD is respected.

In conventional terrestrial photogrammetry, metric cameras are used, whose internal orientation parameters are known with stability. They are mounted on the same tripods and/or plinths and can be swivelled either horizontally or vertically by means of mechanical devices that allow predetermined click rotations. More rarely (due to very high costs) are dual-cameras configurations used, consisting of a pair of synchronised metric cameras mounted on a calibrated length bar.

Different scheme configurations can be used in close range photogrammetry:

- camera axes parallel and orthogonal to the base (normal acquisition).
- slightly converging camera axes (pseudo-normal acquisition).
- converging camera axes.
- oblique and parallel camera axes.

In the first case (Figure 1.9), the configuration involves an arrangement of the camera axes parallel to each other to allow stereoscopic processing of the acquired frames and in any case perpendicular to the acquisition base. This particular configuration translates into a very simple analytical solution, facilitating the planning of acquisition operations and the generation of uniform quality frames; it is therefore the reference terrestrial acquisition configuration.

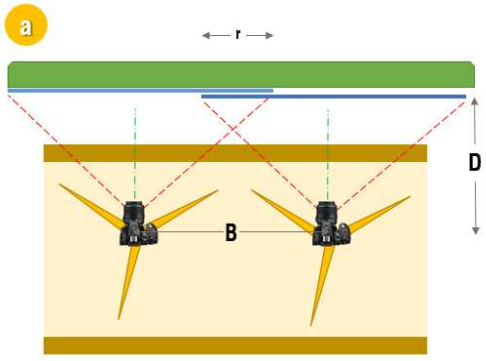


Fig. 1.9 - Parallel and orthogonal axis scheme.

In the second configuration, i.e. pseudo-normal acquisition, the camera is not precisely placed and the axes of acquisition only approximately follow the normal grip pattern. They are set slightly converging, allowing for faster grips, as they are freed from the exact positioning of the cameras, without however compromising the stereoscopic potential of the frames. Orientation operations in this case are enabled by a set of support points, of determined position, chosen on the object (but also outside it). This also allows a limitation of systematic errors, in orientation and restitution operations, which can be largely eliminated thanks to the support points themselves.

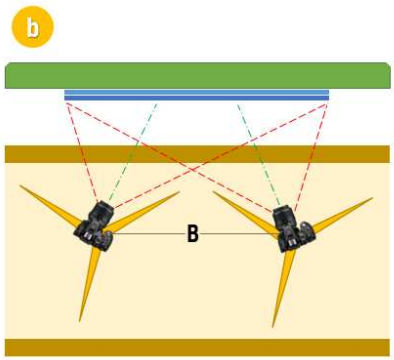


Fig. 1.10 - Converging axes scheme.

In the converging-axis scheme (Figure 1.10), the axes of the cameras form any angle with the acquisition base and are therefore convergent. The advantage is that the survey is carried out with a smaller number of frames (which can be completely covered by the next one); however, in conventional photogrammetry, as the stereoscopic aspect is missing, this type of acquisition is not used. On the contrary, it is a solution that is frequently adopted when the restitution does not make use of stereoscopic vision, but the processing is done on individual frames. When the distance between the camera and the object is very small, the oblique and parallel axes acquisition scheme is used (Figure 1.11); in this case, the axes of the cameras can be tilted with respect to the base, while retaining their parallelism. The value of the base taken, with this solution, is provided by the distance between the axes of the cameras themselves.

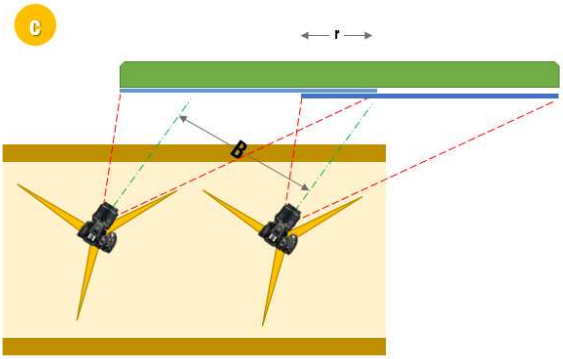


Fig. 1.11 - Scheme with oblique and parallel axes.

As far as the external orientation of the acquired images is concerned, support points will be used, which must be surveyed with traditional instrumentation and topographical techniques, obtaining their coordinates based on an appropriate reference system. In general, a forward intersection is made on each point, adopting the same base as the known base for taking the frames. However, other surveying techniques can also be used, which must nevertheless accurately provide the position of the footholds. The support points, preferably materialised by means of appropriate signs, will be appropriately numbered for their recognition. Their dimensions, so that they are clearly visible on the frames, must be established in relation to the scale of the frames themselves. When it is not possible (or very inconvenient) to materialise the footholds with signs, it is possible to identify, in their place, architectural details on the building, clearly visible on the photograms and easily collimated (corners of windows and door lintels, etc.).

2. COMPUTER VISION TECHNIQUES AND SfM APPROACH FOR PHOTOGRAMMETRY IN BUILDING 3D MODELS

As human beings, we perceive the three-dimensional structure of the objects around us with apparent ease. We are able to distinguish the shape and translucence of any object through the different light and shadow conditions surrounding the environment. Differently, if we observe an image, we are not able to define, through our visual system, the three-dimensional structure of what is depicted. For this reason, researchers in the field of Computer Vision have developed mathematical techniques to recover the three-dimensional shape and appearance of objects in images. Today, we have a number of reliable techniques to accurately calculate a partial 3D model of an environment from thousands of partially overlapping photos, or to reconstruct dense and accurate 3D models using stereo correspondence from a set of views of a particular object or façade. With recent technological developments, it is also possible to track a person moving against a complex background or automatically recognise all the people in a photograph using a combination of face, clothes, and hair. Modelling the visual world in all its rich complexity in 3D is very difficult; the predictive models used in Computer Vision are usually developed in physics (radiometry, optics, and sensor design) and computer graphics. Both these fields model how objects move and animate, depending on how light reflects off their surfaces, is scattered by the atmosphere, refracted by camera lenses (or human eyes) and finally projected onto a flat (or curved) image plane.

In Computer Vision, an attempt is made to describe the world we see in one or more images and to reconstruct properties such as shape, illumination, and colour distribution (Szeliski, R.,2022). Today, Computer Vision algorithms are used in a wide variety of real-world applications, including:

- optical character recognition (OCR) such as, for example, automatic reading of telephone numbers or automatic number plate recognition (ANPR);

- mechanical inspection i.e. rapid inspection of parts for quality assurance, using stereo vision with specialised lighting to measure tolerances on aircraft wings or body parts, or the search for defects in steel castings using X-ray vision;
- fully automated 3D model building (photogrammetry) from aerial photographs;
- medical imaging, i.e., pre-operative and intra-operative imaging (or long-term studies on the brain morphology of ageing people;
- automotive safety by detecting unexpected obstacles such as pedestrians on the road, in conditions where active vision techniques such as radar or lidar do not work well for examples of fully automated driving;
- match move by merging computer-generated imagery (CGI) with live action footage by tracing characteristic points in the source video to estimate 3D camera movement and the shape of the environment.

2.1. Computer Vision

Computer Vision is the field of Artificial Intelligence (AI) and the set of processes that aim to create an approximate model of the real world (3D) from two-dimensional (2D) images, giving computers and machines the ability to see, understand and interpret the visual world around us through machine learning techniques.

Such processes apply machine learning techniques to recognise patterns and deduce meaningful information from digital images, videos or other visual input; in this way, images can be recognised and interpreted in the same way as humans, distinguishing, classifying and ordering them according to their characteristic features, such as size, colour, etc. Since we are dealing with images, one CV-related problem is to determine whether or not the image contains certain objects or activities. This problem can be solved effectively and without difficulty for specific objects in specific situations, for example, the recognition of specific geometric objects such as polyhedra, face recognition or handwritten characters. Things become more complicated in the case of

arbitrary objects in arbitrary situations. To (partly) solve this limitation, there are different approaches to the problem (Fig. 2.1), which in the literature are subdivided into:

- recognition where one or more pre-specified or stored objects can be traced back to generic classes usually together with their 2D or 3D position in the scene.
- identification to identify a specific instance of a class.
- detection in which the image is punctuated until a specific condition is identified.

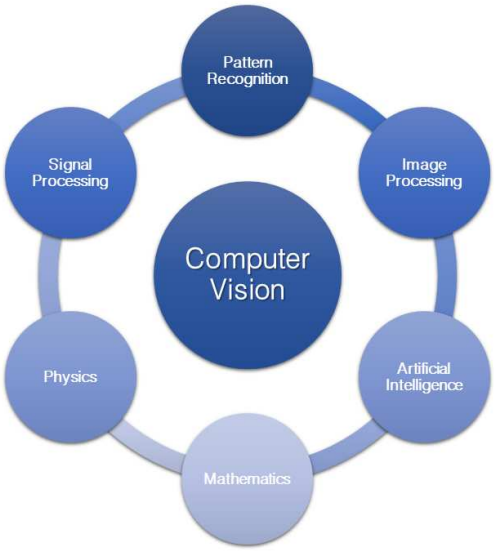


Fig. 2.1 - Computer Vision and related disciplines.

Another typical task is scenery reconstruction: given 2 or more 2D images, an attempt is made to reconstruct a 3D model of the scenery. In the simplest case, this is a set of individual points in a 3D space or entire surfaces. Generally it is important to find the fundamental matrix representing the common points from different images.

2.1.1. Principles and functionality of Computer Vision

In recent years, there has been a sharp increase in the focus on Computer Vision, thanks to the advent of increasingly advanced Machine Learning techniques,

which have made it possible to achieve performance comparable to human beings, and the spread of digital images and videos.

Computer Vision algorithms can perform more or less in-depth investigations of an image, depending on the techniques used, the type of image and the type of task performed. Possible tasks include:

- **Image Classification** i.e. the analysis of image content and attribution of a label.
- **Object Detection** i.e. the identification of one or more entities within an image.
- **Image Segmentation** that divides the image into sections.
- **Face Recognition**, which consists of the recognition of people's faces.
- **Action Recognition** which identifies one or more entities and their relationship in time and space to identify and describe specific actions.
- **Visual Relationship Detection** includes the relationship between objects in an image.
- **Emotion Recognition** detecting the *sentiment* of an image.
- **Image Editing** modifies an image by obscuring sensitive data.

Of these, the most currently used are those related to Image Classification and Object Detection solutions. The development of Computer Vision solutions entails quite a few difficulties, for example, in the case of Image Classification the image may be ambiguous and may be associated to several options, so the algorithm must be able to attribute the most appropriate label; furthermore, in the case of Image Segmentation it is necessary not only to identify the entities within the image, but also to delimit and analyse them individually. More generally, among the main criticalities in the implementation of Computer Vision projects are those related to the creation of a sufficiently large dataset for the training of the algorithm and the subsequent phase, i.e. the phase in which the algorithm is taught to understand the image even in the presence of transformations (e.g. non-optimal brightness conditions, deformation or partial coverage of the subject, scale variations). Once the dataset has been acquired, CV algorithms and tools are used to analyse the data several times until the images are distinguished and

recognised. To train a computer to recognise the tyres of a car, for example, it is necessary to feed it with a large quantity of tyre images and tyre-related elements in order to learn the differences and recognise a tyre, especially one without defects.

Two main technologies are used to realise automatic object recognition: a type of Machine Learning (ML) called Deep Learning (DL) and a Convolutional Neural Network (CNN). ML uses algorithm models that allow a computer to learn the context of visual data by itself. If enough data is entered into this model, the computer will 'look' at the data and learn by itself to distinguish one image from another. Algorithms allow the machine to learn on its own, without someone programming it to recognise an image. A CNN helps an ML or DL model 'look' by breaking down images into pixels that are given tags or labels. It uses the tags to perform convolutions (a mathematical operation on two features to produce a third feature) and makes predictions based on what it 'sees'. The neural network performs convolutions and checks the accuracy of its predictions over a series of iterations until the predictions begin to come true. It is thus able to recognise or see images in a similar way to humans.

Like a human being distinguishing an image from a distance, a CNN first distinguishes sharp contours and simple shapes, then adds information as it performs iterations of its predictions. A CNN is used for analysing single images. A recurrent neural network (RNN) is similarly used for video applications to help computers understand how images in a series of frames are connected.

2.1.2. Scope of application

Today, it is easier to create a 3D model of an object with millimetre precision than to build an algorithm that is able to recognise objects automatically. In fact, the latter is still a very difficult problem to tackle, even though scientific progress in recent years is increasingly improving algorithms towards human precision.

Computer Vision is therefore a very difficult approach because there is a huge difference between pixels and their meaning. In fact, what the computer sees in an RGB image, for example, with a resolution of 200x200, is a set of 120,000 values. The process that leads from these numbers to meaningful information is very difficult.

It is possible to divide the information obtained from images in Computer Vision into two categories: measurements and semantic information.

In the first case, an example would be a robot navigating in an unknown place with the ability to scan its surroundings to calculate the best route. Using CV algorithms, we can in fact measure the space around a robot and create a map of its environment. Stereo cameras will provide depth information, like our eyes, through triangulation, while stereoscopic vision will produce an accurate depth map from the stereo images. If we increase the number of viewpoints to cover all sides of an object, we can then create a 3D map of an object and produce a 3D surface representing the object.

Another approach could be to reconstruct the 3D model of a monument through all the results of a Google image search or, for example, a search where Computer Vision can help to understand the 3D geometry of an object. In addition to measurement information, an image contains a very dense amount of semantic information; it is possible to label objects in an image, to label the entire scene, to recognise people, actions, gestures, faces (Krishna, R., 2017).

Computer Vision algorithms are used in the analysis of images in various contexts in order to extract a range of content and information; some areas of application are listed below.

- **Special effects**: shape and motion capture are new techniques used in films to animate digital characters by recording the movements performed by a human actor. To do this, we have to find the exact positions of the markers on the actor's face in a 3D space, and then recreate them on the digital avatar.
- **3D urban modelling**: by photographing a city with a drone, the images can be used to render a 3D model of the city. Computer vision is used to combine all the photos into a single 3D model.
- **Scene Recognition**: It is possible to recognise where a photo was taken. For example, the photo of a landmark can be compared with billions of photos on Google to find the best matches.

- **Face detection**: Face detection has been used for many years in cameras to take better photos and focus on faces, allowing cameras, for example, to automatically take photos when the subject smiles. Finally, we can also use Computer Vision for biometrics, using iris or fingerprint pattern recognition.
- **Optical Character Recognition (OCR)**: one of the oldest successful applications of Computer Vision is character and number recognition. This can be used to read postal codes or car number plates.
- **Visual search**: perform a Google search using an image as a query.
- **Self-driving cars**: autonomous driving is one of the most widely used applications in the field of CV. Companies such as Tesla, Google or General Motors are competing to be the first to build increasingly self-driving cars.
- **Augmented Reality**: AR is also a much sought-after field at the moment, and several companies are competing to provide the best mobile AR platform solutions, such as Apple, which released ARKit and produced impressive applications.
- **Virtual reality**: VR uses Computer vision techniques similar to AR. The algorithm must know the position of the user and all surrounding objects, when the user moves, everything must be updated realistically and smoothly.

2.2. The SfM and MVS algorithms

At the beginning of the 1990s, with the spread of the first digital cameras, the focus in the field of Computer Vision was on the analysis of image sequences, addressing the reconstruction of Structure from Motion (Dhond et al., 1989; Faugeras, 1993). The main problem was to estimate the structure of the scene framed by the cameras with a limited amount of information available. In particular, even without having precise information regarding the orientation of the cameras, it is possible to extract directly

from the acquired images some important aspects regarding the geometry by reconstructing the entire structure of the scene filmed.

The methods for solving Structure from Motion are numerous, as are the algorithms involved in solving the various stages of the process. All resolution methods are based on similar methodologies where, in the first step, detectable features are extracted from different images (features). Features are a series of elements (usually points) that, with a certain degree of probability, can be found on different frames. Once a sufficiently large number of features have been identified, by means of appropriate geometric and radiometric criteria, correspondences are estimated that form a first attempt to catalogue conjugate points on the various images. At this point, it is possible to estimate a preliminary geometry acquisition and eliminate erroneous correspondences. At the end of the estimations, the identified geometry does not strictly verify the real conditions of the photogrammetric block, in the sense that it provides a projectively equivalent geometry to the real one. To obtain a real (metric) reconstruction of the block and the scene, knowledge of the internal orientation parameters is sufficient. These parameters can be known a priori, e.g. through a self-calibration procedure, or they can be estimated a posteriori during the reconstruction process through a methodology known as self-calibration. Finally, it is possible to obtain an optimal estimate using the results obtained as approximate parameters of a system solved successively to least squares (bundle adjustment) in which additional information on the block (support points, known interdistances, orientations, etc.) can be included. It remains to complete the reconstruction process by means of a pixel-by-pixel triangulation procedure in all available images, at this stage referred to as dense matching. At the end of this phase we will have the point cloud on which it will then be possible to eventually model a solid structure formed from mesh.

The main steps of the procedure, summarised in the following diagram (Fig. 2.2), will be analysed in detail below, with particular emphasis on the 3D reconstruction procedures, which represent the starting step in subsequent analysis and experimentation.

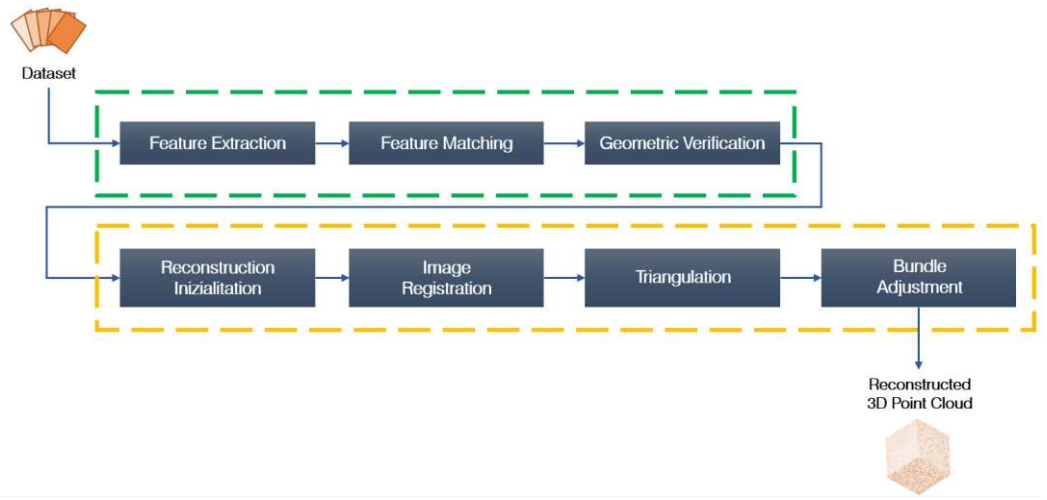


Fig. 2.2 - Main stages of the reconstruction process.

The stages of the SfM-based photogrammetry processing pipeline could be summarised as follows:

- image pre-processing,
- extraction and matching of key points,
- calculation of an initial solution for external camera orientation parameters and 3D positions of constraint points
- beam adjustment leading to the generation of dense point clouds; subsequently, mesh models can be reconstructed.

The main steps of this procedure are typically performed using the SfM and MVS approaches. This processing pipeline has become popular mainly due to its ability to assess the external orientation of the camera without any knowledge of the camera position and 3D points.

2.2.1. SfM and MVS approach for 3D reconstruction

The SfM pipeline allows a 3D object to be reconstructed from a set of images acquired from different vantage points through the following two steps:

- Searching for matches.

- Reconstruction phase.

The first phase, the match-finding phase, can in turn be divided into three sub-phases:

- Feature extraction.
- Matching of characteristics.
- Geometric verification.

Feature extraction (Pepe et al., 2020), using a suitable algorithm, such as the Scale-Invariant Feature Transform (SIFT) feature detector, allows for the identification and description of local features and image points of interest in images (key-points).

Feature Matching searches for matches between features by finding the most similar ones and identifies overlapping parts of the images. The key points and the features extracted from them are used to determine the images that portray common parts of the scene and are therefore at least partially overlapping. If two points in different images have the same description, then it is possible to consider these points to be the same in the scene. Various strategies to efficiently calculate correspondences between images can be used (Bianco et al., 2018). The output of this step is a set of images that overlap at least in pairs, identifying feature correspondences between the different images acquired.

The geometric verification is necessary because the previous matching step only verifies that the pairs of images apparently have points in common; there is no guarantee that the matches found are real as outliers may be included. It is necessary to find a geometric transformation that correctly maps a sufficient number of points in common between two images. If this is done, the two images are considered geometrically verified, which means that the points also correspond in the geometry of the scene. Depending on the spatial configuration with which the images were acquired, different methods can be used to describe their geometric relationship. For example, a homography can be used to describe the transformation between two images of a camera acquiring a planar scene. Epipolar geometry, on the other hand, can be used to describe the movement of a camera through the essential matrix if the camera's intrinsic calibration parameters are known. Alternatively, if the parameters are unknown, the

uncalibrated fundamental matrix can be used. Furthermore, since the matches obtained from the matching phase are often contaminated by outliers, it is necessary to use robust estimation techniques such as RANSAC (RANdom SAmple Consensus) during the geometric verification process (Hartley et al., 2003). In addition to RANSAC estimation, certain optimisations can be used to reduce the execution time. The output of this stage of the pipeline is the so-called scene graph, a graph whose nodes represent the images and edges join the pairs of images considered geometrically verified.

As for the reconstruction phase, this in turn can be divided into four stages:

- Initialisation,
- Image recording,
- Triangulation,
- Beam adjustment (BA).

The initialisation of incremental reconstruction is an important step because bad initialisation leads to bad reconstruction of the three-dimensional model.

To achieve a good reconstruction, it is preferable to start from a dense region of the scene graph, so that the redundancy of the correspondences provides a solid basis for the reconstruction. In the case where the reconstruction starts from an area with few images, the bundle adjustment process does not have enough information to refine the position of the reconstructed poses and camera points, which leads to an accumulation of errors and a poor final result. If more than one image pair can be used as a starting point, the one with the highest number of geometrically verified matching points is chosen. The points in common between the two images are used as the first points of the reconstructed cloud. Subsequent steps iteratively add new points to the reconstruction by considering one new image at a time.

Then, using the feature matches, a new image is added to the reconstruction and, as a result, the position and orientation of the corresponding camera is calculated (image registration phase). For the newly recorded image, it is necessary to calculate the camera pose (position and rotation) that captured it; this can be achieved by using correspondence with known 3D reconstruction points. Therefore, this step utilises the 2D-3D correspondence between the key points of the newly added image and the

points of the 3D reconstruction that are associated with the key points of the previously recorded images. To estimate the camera pose, it is necessary to define the position in terms of 3D co-ordinates in the reference system and the rotation (pitch, roll and yaw axes), for a total of six degrees of freedom. This is possible by solving the Perspective-n-Point (PnP) problem. Various algorithms can be used to solve the PnP problem (Fraser et al., 1997), which together with RANSAC (or its variants) allows for a robust estimation of the camera pose. The newly recorded image has not yet contributed to the addition of new points, this will be done in the next step, the triangulation phase.

In the triangulation phase, the newly recorded image can observe other new points, these points can be added to the three-dimensional reconstruction if they are observed by at least one previously recorded image. A triangulation process is used to define the 3D coordinates of the new points that can be added to the reconstruction and thus generate a denser point cloud.

In this step, a pair of recorded images with points in common is taken and the respective camera poses are estimated. The algorithm attempts to estimate the 3D co-ordinates of each point in common between the two images. To solve the triangulation problem, an epipolar constraint is placed. It is necessary for the positions from which the images were acquired to identify the acquisition position of the counterpart in the image, these points are called epipoles. In the ideal case, epipolar lines can be used to define the epipolar plane on which the point whose position is to be estimated lies. However, due to inaccuracies in the earlier stages of the pipeline, it is possible that the point will not lie at the exact intersection of the epipolar lines; this error is known as the reprojection error. In order to solve this problem, characteristic algorithms that take the inaccuracy into account are required.

Since camera pose estimation and triangulation can generate inaccuracies in the reconstruction, a method must be adopted to minimise the accumulation of such errors. The purpose of the Bundle Adjustment (BA) phase is to prevent inaccuracies in the camera pose estimation from propagating into the triangulation of the cloud points and vice versa. The BA can thus be formulated as a reconstruction refinement that produces optimal values for the reconstructed 3D points and camera calibration

parameters. The algorithm used for the Levenberg-Marquardt (LM) BA, also known as Damped Least-Squares, allows the least-squares method to be solved for the non-linear case. This step has a high computational cost and must be performed for each image added to the reconstruction. To reduce processing time, BA can only be performed locally (i.e. only for a small number of images/cameras).

In general, although BA is not strictly part of the SfM approach, it is a very common step used to refine and refine the initial SfM model. Given a set of camera parameters and a set of tracks the BA minimises the following non-linear least squares error (Furukawa et al., 2015):

$$E(P,M) = \sum_{j} \sum_{i \in V(J)} \left| P_i\left(M^j\right) - m_i^j \right|^2 \tag{9}$$

where:

P_i camera parameters

M^j 3D co-ordinates of a track

m_i^j 2D co-ordinates of the projection of its image

$V_{(j)}$ list of camera indices where the point is visible M^j

$P_i\left(M^j\right)$ coordinate of the projected 2D image of the 3D point M^j in the camera i using camera parameters P_i.

The value of $E(P,M)$ is typically measured in pixels squared, but a more common metric to express estimation accuracy is to use the root mean square error or RMSE defined as:

$$RMSE(P,M) = \sqrt{\frac{E(P,M)}{N}} \tag{10}$$

where:

N number of residual terms.

Typical RMSE values before bundle adjustment are of the order of several pixels, while values after bundle adjustment are often less than one pixel.

The BA solution includes estimates of the external orientation parameters of each image and the point coordinates obtained by collimating a number of homologous points in all images. Pre-calibration, on the other hand, requires the introduction of the camera's internal parameter values as priors into the BA optimisation.

In 1971, Brown established a camera model widely used in photogrammetry; this model implements the collinearity condition to establish a mathematically rigorous relationship between image and object (Brown, 1971). It utilises 10 additional parameters (Aps) relating to internal camera orientation (Δxp, Δyp, Δc), pixel shape uncertainty (skew factor Sx), reference system non-orthogonality (cutoff factor ∧), radial symmetrical lens distortion (k1, k2, k3) and tangential lens distortion (p1, p2). The development of so-called 'self-calibration' of cameras has made it possible to obtain high-precision 3D models even with low-cost (non-professional) cameras. Two different approaches are typically used in the photogrammetric workflow: camera self-calibration and pre-calibration.

In the “self-calibration” approach, the internal parameters, external orientation and object coordinates of the points are unknown. In this approach, “additional parameters” (AP) are used to account for systematic errors due to the frame acquisition process. Self-calibration with BA can be described by the extended collinearity model (Fischler et al. 1987):

$$\begin{pmatrix} x - x_P + \Delta x \\ y - y_P + \Delta y \\ -c \end{pmatrix} = \lambda \boldsymbol{R} \begin{pmatrix} X - X^c \\ Y - Y^c \\ Z - Z^c \end{pmatrix} \tag{11}$$

where:

X, Y, Z	Object point coordinates;
X^c, Y^c, Z^c	Perspective centre;
x_P, y_P	Principal point offsets;
$\Delta x, \Delta y$	Image coordinates perturbation;
$\boldsymbol{R}$	Rotation matrix;
x, y	Image space coordinates;
c	Focal length (or principal distance);

λ Scale factor for the image radius;

In addition, the image coordinates perturbation can be calculated as:

$$\begin{aligned}\Delta_x &= -\Delta x_0 + \frac{\Delta_C}{C}\bar{x} + \bar{x}S_x + \bar{y}A + (k_1 r^2 + k_2 r^4 + k_3 r^6)\bar{x} + p_1(r^2 + 2\bar{x}^2) + 2p_2\overline{xy} \\ \Delta_y &= -\Delta y_0 + \frac{\Delta_C}{C}\bar{y} + \bar{y}S_x + \bar{x}A + (k_1 r^2 + k_2 r^4 + k_3 r^6)\bar{y} + p_2(r^2 + 2\bar{y}^2) + 2p_1\overline{xy}\end{aligned} \quad (12)$$

where:

$\bar{x} = x - x_0$

$\bar{y} = y - y_0$

$r^2 = \bar{x}^2 + \bar{y}^2$

while:

k_1, k_2, k_3 radial distortion coefficients

p_1, p_2 tangential distortion coefficients

S_x the shape factor (uncertainty about the shape of the pixel)

A shear factor (non-orthogonality of the reference system)

An example of camera calibration software is Agisoft Lens, which supports the estimation of the entire camera calibration matrix, including non-linear distortion coefficients based on Brown's model. The camera calibration procedure is typically achieved by correctly processing a series of photos of a calibration model (checkerboard) taken from different viewpoints. Typical results of the calibration procedure are the estimated values of the camera parameters and their uncertainties, expressed as standard deviation (SD) values.

MVS algorithms are very sensitive to the accuracy of the estimated camera models. In fact, they use the epipolar geometry (defined by the camera models) to shrink the 2D matching problem into a 1D matching problem. If the reprojection error is large, a pixel may never be matched to its true correspondence, significantly degrading MVS performance.

The robustness of MVS against camera reprojection error depends mainly on the tolerance of the mismatch criterion. The larger the Ω domain of the photocoherence measurement, the more robust the measurement is. Unfortunately, large domains tend

to produce an excessively smooth geometry, so a trade-off between accuracy and robustness is necessary.

Since MVS is so sensitive to reprojection errors, bundle adjustment is often a requirement for MVS, with the goal of achieving sub-pixel reprojection errors. Since the reprojection error is measured in pixels, it is possible to reduce the sampling of input images and resize the camera parameters until the reprojection error falls below a certain threshold. This approach will work as long as the subsampled images still contain sufficient texture and detail to make the MVS work. The origins of multi-view stereoscopy can be traced back to human stereopsis and early attempts to solve the stereoscopic matching problem as a computational problem. To date, two-view stereoscopy algorithms have been a very active and fruitful area of research. The multiview version of stereoscopy arose as a natural improvement of the two-view case. Instead of capturing two photographs from two different viewpoints, multi-view stereoscopy would capture multiple viewpoints to increase robustness, e.g. with respect to image noise or surface structure.

What originally started to improve two-view stereoscopy has now evolved into a different kind of problem. Although MVS shares the same principles as classical stereo algorithms, MVS algorithms are designed to handle images with more variable viewpoints, such as a set of images surrounding an object, and also to handle very large numbers of images, even on the order of millions of images. The difference in the nature of the MVS problem ends up producing significantly different algorithms than classical stereoscopy. As an example, industrial applications for 3D mapping process millions of photographs over hundreds of kilometres, effectively reconstructing large metropolitan areas, countries and eventually the entire world. Pixel matching between images is a challenging problem that is not unique to stereoscopy or multi-image stereoscopy. Optical flow is another very active field in Computer Vision, which addresses the problem of dense matching between images. The main differences with MVS are that optical flow is typically a two-image problem (similar to two-view stereo), the cameras are not calibrated, and its main application is image interpolation rather than 3D reconstruction. Note that in the MVS case, where the camera parameters are known, the solution for

the 3D geometry of the scene is exactly equivalent to solving the input image matching problem. By considering a 3D point belonging to the scene and projecting it into the set of visible cameras, a unique correspondence is established between the projected co-ordinates in each image.

Given a pixel in an image, two elements must be considered in order to find the corresponding pixels in other images:

- An efficient way to generate possible candidate pixels in other images,
- A measure indicating the probability that a given candidate is the correct match.

If the geometry of the camera is unknown, as is typically the case in optical flow, each pixel in one image can correspond to any other pixel in another image. That is, for each pixel, a 2D search must be made in the other image. However, when the camera parameters are known (and the scene is rigid), the image matching problem is simplified from a 2D to a 1D search. A pixel in an image generates a 3D optical ray that passes through the pixel and the camera centre of the image. The corresponding pixel on another image can only lie on the projection of that optical ray in the second image. The different geometric constraints that are generated when multiple cameras look at the same 3D scene from different viewpoints are known as epipolar geometry (Furukawa et al., 2015).

2.2.2. Images Acquisition in SfM and MVS Approach

SfM and MVS approaches have become very popular because they can provide fully automated 3D modelling for arbitrary images without any prior knowledge or on-site measurements. In fact, a 3D point cloud can be constructed from a collection of stereoscopic images. The standard error $\sigma(X,Y,Z)$ of the coordinates X,Y,Z of a generic 3D point can be evaluated by the following relationship (Fraser 2017):

$$\sigma_{X,Y,Z} = \frac{q\ Z}{c\sqrt{k}}\sigma_{p_\xi}, \tag{13}$$

where:

q design factor expressing the strength of the camera network (generally between 0.4 and 2);

c focal length;

k number of images used to determine the same point;

σ_{p_ξ} measurement precision of image coordinates.

In order to obtain a high-quality three-dimensional model, it is necessary to increase the base-to-depth ratio (B/D) during image acquisition and to obtain a series of images with converging rather than parallel optical axes.

In general, the SfM-based approach offers enormous redundancy to compensate for the potentially modest loss of geometric strength.

In UAV photogrammetry, to create a robust and reliable network, it is useful to create parallel datasets and where necessary at different height levels (Fraser 2019).

To address SfM problems in Computer Vision, projective geometry is a fundamental tool, especially in multiple view geometry. For this reason, a brief introduction to epipolar geometry is given before explaining the SfM and MVS process.

In a pin-hole model, a 3D point X is projected into an image point x as:

$$\lambda x = PX \tag{14}$$

where:

λ is the scalar that represents the inverse depth of the 3D point, and P is called the camera matrix and can be decomposed as:

$$P = K[\boldsymbol{R}|t] \tag{15}$$

where:

$\boldsymbol{R}$ is the rotation matrix relative to the camera orientation, t represents a 3D translation vector relative to the position of the camera centre, and K represents the intrinsic calibration matrix representing the projection properties of the camera (Solem 2012).

Epipolar geometry thus describes the geometric relationships that exist between stereo images and that depend only on the internal parameters of the cameras and their poses.

Considering the point X in three spaces taken in two views, a x_1 in the first and a x_2 in the second, it is necessary to stabilise the relationships between these two points. Figure 3 shows a representation of the epipolar geometry; the distance between the two centres of the camera C_1 e C_2, is called the baseline; the intersection of the baseline with the planes of the image generates the epipoles, e_1 e e_2. The lines l_1 e l_2 are called epipolar lines and the plane containing the baseline and epipolar lines is called the epipolar plane.

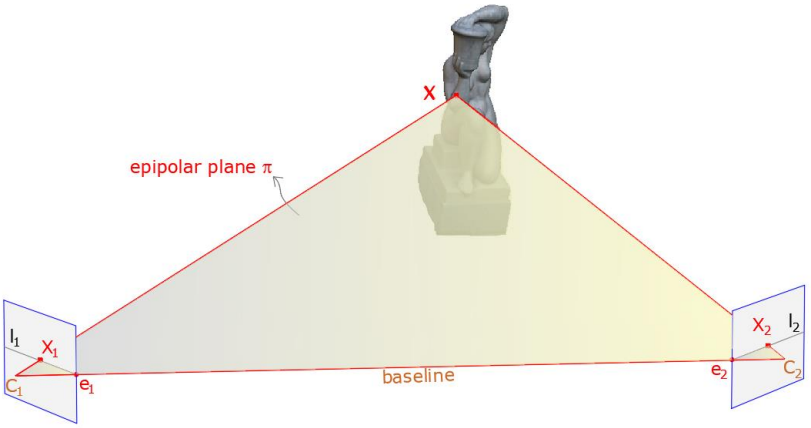


Fig. 2.3 - Fundamentals of epipolar geometry.

Two-view geometry can always transform cameras with homography to simplify the treatment, and the homographic transformation is a rigid translation. One approach is to place the origin of the reference system and axes in the first camera. Under this condition, the epipolar constraint equation can be written:

$$x_2^t \, F \, x_1 = 0 \tag{16}$$

where F, the fundamental matrix, is $k_2^{-t} \, S_t R \, k_1^{-t}$. S_t represents the skew-symmetric matrix.

The camera arrays, if the internal calibration parameters (k_1 and K_2) are not known, they can only be defined up to a projective transformation (angles and distances not respected), otherwise, with a known calibration, the reconstruction will be metric.

In the latter case, the fundamental matrix for the coordinates normalised by calibration is called the essential matrix and is usually denoted by E instead of F to emphasise that this is the calibrated case; the image coordinates are normalised and the equation of the epipolar constraint becomes $x_{k_2}^t \, E \, x_{k_1} = 0$.

2.3. Point cloud processing and 3D reconstruction

Processing and reconstruction of point clouds are essential steps in 3D reconstruction. Point clouds are sets of data points in a 3D coordinate system that represent the surface of an object or scene.

3D photogrammetric reconstruction is used in various applications, such as cultural heritage conservation, architecture, engineering, and entertainment. The process involves several steps, including image acquisition, feature detection and matching, camera calibration, point cloud generation and mesh creation. Each of these steps plays an essential role in the overall success of the reconstruction process and utilises the SfM and MVS algorithms discussed in the previous sections.

In the following, the steps that lead from the acquisition of an image dataset to the processing of the three-dimensional point cloud will be analysed.

The point cloud represents the starting point for the subsequent stages of creating a 3D mesh, i.e. a continuous surface, to represent the object or scene being surveyed.

2.3.1. Phases of photogrammetric processing

Image processing takes place through four distinct and successive stages:

1. Structure-from-motion (SfM) and Multiview Stereo Reconstruction (MVS).
2. Scaling of the 3D model.
3. Mesh reconstruction.
4. Colour of the mesh.

The first is the key phase in the processing and is generally the longest of the entire process from a software processing time perspective. In this phase, the taking geometry of the photographs is reconstructed and the dense point cloud is processed, i.e. the raw data on which subsequent processing is based is obtained. In order to be able to reconstruct the three-dimensionality of a scene, it is necessary to reconstruct the shooting position of the individual photographs (capture geometry) so that the position of the objects in them can subsequently be deduced by triangulation. Whereas traditional photogrammetry uses GNNS data from the images or control points of known coordinates, automatic photogrammetry is based on the identification of clearly recognisable key points in three or more images, which are used to create correspondences between the images and link them together (*image matching*).

Starting from the obtained key points, through a projective star photogrammetric triangulation procedure (*bundle adjustement*) the camera is automatically calibrated (internal orientation i.e. calculation of focal length and main point) and the shooting position of the individual photographs is reconstructed (external orientation i.e. determination of the coordinates of the centres of the frame and rotations), and for each key point, the actual x,y,z coordinates are obtained, which are materialised three-dimensionally in a sparse points cloud (*sparse reconstruction*).

In the next step, the dense point cloud is densified by increasing the number of points (dense reconstruction); the data on the x,y,z position of the key points are used as starting points to extend the image analysis to the areas proximal to them and to recognise and extract the x,y,z position of the surrounding elements. The result will be a dense point cloud. A file will be obtained in which each point is associated with a precise position in space, in each reference system, and each point will be associated with a colour in the standard RGB encoding used in the digital domain.

The dense point cloud obtained in the previous processing step is geometrically correct but to scale, this is because the software has no way of calculating the dimensions of the elements in the images. The 3D model must, therefore, be scaled up to the real dimensions. This scaling process can be achieved by indicating to the processing software the actual size of elements appearing in the model, or by inserting a series of

elements of known dimensions, such as rulers or metric bars of which the interdistances are known a priori, into the scene to be acquired. In order to obtain high-precision models and depending on the geometric and topological characteristics of the object or area to be represented, a series of targets (coded or not) are arranged and surveyed using geomatic surveying techniques (Total Stations, GNSS receivers). In this way it will be possible to determine the coordinates of the targets, in a suitable reference system, and to scale and georeference the model to be reconstructed in 3D.

In the last two steps, a continuous surface composed of polygons, known as a mesh, is reconstructed from a dense point cloud, the vertices of which are the points in the cloud and make up a triangular or tetrahedral surface that correctly fits this cloud. Using an appropriate triangulation process, a given set of points is converted into a coherent polygonal model. Such triangulation can be performed in two or three dimensions, depending on the geometry of the input data.

A popular two-dimensional interpolation method is Delaunay Triangulation (DT). Based on Dirichlet's intuition for the decomposition of a domain into several adjacent (and non-overlapping) convex polygons, Delaunay's method allows us to define three-way points such that the circle surrounding each triangle contains no other elements of the original set of points (Tsai et al., 1993). A 3D Delaunay triangulation is also called a Delaunay tetrahedral mesh. A tetrahedral mesh is a discrete representation of a continuous 3D space, both in topology and geometry (Si, 2015). Indeed, to generate unstructured meshes in 2D or 3D spaces, given a point cloud, Delaunay triangulation provides a good set of triangles to be used as polygons in the model. In the case of 3D triangulations (Fig. 2.4), tetrahedral regions are drawn based on a geometric constraint that establishes the shape and distance between the nodes that make up the mesh elements. The points of the mesh are selected in such a way as to respect the properties described below:

- Circumference condition: if a sphere is circumscribed by any tetrahedron of the triangulation, that sphere must contain only the points that make it up. All other grid points that make up the mesh must lie outside

this 3D volume. This condition is equivalent to limiting the mesh density for a given length scale.

- Angle maximisation condition: the smallest solid internal angles in the tetrahedra of the resulting Delaunay triangulation attempt to maximise the minimum solid angle for all angles in the triangulation. Consequently, the tetrahedra of the mesh may not be equilateral.
- Order-independent and unique: a uniqueness property of Delaunay triangulation states that the calculated triangulation will be independent of the order in which new points are added to the mesh.

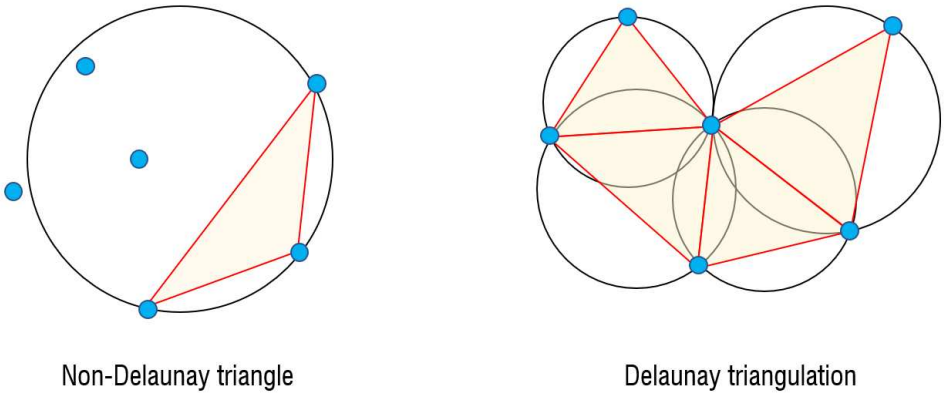


Fig. 2.4 - Example triangles used to construct 3D Delaunay triangulations.

Some meshing algorithms are iterative and add new points as the algorithm proceeds. The above-mentioned properties of Delaunay triangulation ensure that the same mesh is generated regardless of the order in which points are added.

In general, the conversion from point clouds to polygonal surfaces is based on four steps (Remondino, 2006):

- Pre-processing where erroneous or noise-affected data are eliminated or reduced.
- Determination of the global topology of the object surface in which the relationships between the various parts of the object are established.

- Polygonal surface generation, i.e. triangular (or tetrahedral) meshes are created that meet specific quality requirements;
- Post-processing, which consists of editing and refining the 3D model.

The post-processing phase of the polygonal surface model may be necessary because the mesh created may contain triangles with undesirable sizes and shapes, so it is important to perform an optimisation of the mesh to create a network of uniform and regular triangles. This can be done for example by using smoothing or decimation algorithms on the mesh.

The colour attribute can also be applied to the resulting 3D model. In fact, since the vertices of each polygon correspond to the points of the point cloud, the colour of the latter can be transferred to the corresponding polygons (obviously each polygon will have a colour mediated between that of the three vertices). This method is commonly referred to as *colour-per-vertex*. The overall rendering of the surface colour is closely related to the resolution of the model: the higher its resolution (high number of small polygons), the higher the "visual" quality of the surface colour will be; on the contrary, if the model resolution is low (low number of large polygons), the colour rendering will be very approximate, with a blurred or fuzzy visual effect. Another method for colouring the 3D model involves the use of source images. Since these are already oriented with respect to the model, they can be projected onto the model and applied to the polygons of the mesh. The images are then brought together into a new square image called a *texture* (a kind of mosaic of all the images), which covers the mesh. The advantage of this method is that the visual quality of the colour is equal to the quality of photographs, and that it is unrelated to the resolution of the model, as the texture literally covers the mesh.

3. PHOTOGRAMMETRIC SOFTWARE AND OUTPUT

Using software based on SfM-MVS algorithms, it is possible to obtain 3D point clouds useful for constructing realistic 3D models, digital surface models and other derived technical representations such as orthophotos, features, contours, etc.

In recent years, there has been an increase in the development of new commercial photogrammetry software; as a result, more and more high-performance software packages are available on the market, in addition to the open-source software that the scientific community is continuously developing. The development of open-source software is an important research direction, for:

- Maximise efficiency in the construction of 3D models;
- Make interfaces user-friendly;
- Reduce the need for further processing in other software to obtain the desired output;
- Improve the performance of 3D information reconstruction in the case of objects with complex geometries.

There are some common factors that can be used to compare photogrammetric software: in this chapter, we will explore some of the main features of the photogrammetric software available on the market, distinguishing them according to the type of licence (commercial or open-source).

Some factors that influence the choice of one photogrammetric processing software over another are listed below.

- **Functionality**

  The first thing to consider when comparing photogrammetric software is their range of functionality. For example, some software may only offer 3D model creation, while others may also offer the creation of orthomosaics, digital terrain models (DTM) and more.

- **Ease of use**

  Ease of use is another important factor to consider. Software with an intuitive and easy-to-use user interface may be preferable to software with a complicated and difficult user interface.

- **Processing speed**

  Processing speed is another factor to consider. Some photogrammetric software can process images faster than others, which can be important for projects that require fast data processing.

- **Accuracy**

  The accuracy of results is another important factor to consider. Some photogrammetric software can produce more precise results than others, which can be important for projects that require a high level of accuracy.

- **Support and training**

  Finally, the support and training offered by the software retailer can make all the difference. Software that offers extensive documentation and technical support can be easier to use and solve any problems that may arise. There are in fact several reviews and comparisons of photogrammetric software available online. For example, a comparison of Agisoft Metashape, Pix4Dmapper and RealityCapture published by GIS Geography compared the three software packages based on processing speed, accuracy, quality of results and ease of use. The comparison showed that Pix4Dmapper is the fastest and RealityCapture the most accurate, while Metashape offers a good balance between speed and accuracy.

In general, the comparison of photogrammetric software depends on the specific needs of the user and the type of project. However, considering the factors mentioned above, it is possible to make an informed choice between the available photogrammetric software.

3.1. Commercial software

3.1.1. Agisoft Metashape

Agisoft Metashape is one of the most popular and widely used photogrammetric software in the industry. The software enables the acquisition, processing and analysis of data from images to create 3D models, orthomosaics and more. Agisoft Metashape is a powerful and versatile photogrammetric software that offers a wide range of features and tools, is able to process large datasets, supports various data sources and allows you to create accurate and detailed 3D models. The process of processing acquired images for the creation of 3D models, orthomosaics and other final products consists of several steps including camera calibration, image alignment, 3D model creation, orthomosaic generation and the creation of other final products. The classic pipeline used for the production of 3D models in Metashape consists of several steps, the first of which is the import of the acquired images into the software. Metashape supports several file formats, including JPEG, TIFF and RAW. The next step is camera calibration, which is a process that is used to determine internal camera parameters such as focal length, lens distortion and optical axis position. These parameters are used to correct image distortions and improve the accuracy of 3D models and for further processing of orthomosaics.

Using the SfM and MVS algorithms, the next steps involve the alignment of the images and the construction of the dense point cloud. The alignment process is of paramount importance as it allows the acquired images to be placed in a correct position in space. The software uses various alignment techniques, such as feature correlation and reprojection error analysis, to align the images. After the creation of the dense point cloud, a 3D model of the acquired scene can be generated using the aligned images. The software uses a triangulation technique that consists of using the images to triangulate the position of the 3D points in the scene.

In addition, Agisoft Metashape allows the production and export of high-resolution orthophotos using rectification techniques to correct image distortions, contour

lines, digital terrain and elevation models, depth maps, and textures, as well as the ability to filter and classify the point cloud, and measure distances, areas and volumes using tools and instruments integrated in the software.

3.1.2. 3DF Zephyr

3DF Zephyr is a photogrammetric software for creating 3D models, orthomosaics, point clouds and meshes from images. The software is produced in Italy and distributed by 3Dflow, a private consulting and software production company operating in the field of computer vision and image processing. It was founded in 2011 as a spin-off of the University of Verona and was recognised as a spin-off of the University of Udine in 2012. 3DF Zephyr's pipeline is classic for creating 3D models and orthomosaics from the import of images in different file formats, including JPEG, TIFF and RAW. For the alignment phase, 3DF Zephyr uses a combination of feature recognition and correlation techniques to align images. Furthermore, after the creation of the dense point cloud and the construction of the mesh, in order to remove any unwanted points or areas from the mesh, the software offers a wide range of tools, including single point removal, point density reduction and hole filling.

Other end products such as contour lines, depth maps and 3D model textures can also be created here. 3DF Zephyr also offers a wide range of advanced features, such as batch processing, multi-scale reconstruction, LiDAR data fusion and 3D model creation from video. In addition, the software offers an intuitive and user-friendly interface that allows users to easily access all programme features and tools. 3DF Zephyr is available in several versions, including a free version with limited functionality and paid versions with advanced features.

3.1.3. Pix4D

Pix4D is advanced photogrammetry software designed to create precise 3D models and maps using aerial or terrestrial images. Thanks to its innovative technology, Pix4D has become an essential tool for various industries, including agriculture, engineering, construction, architecture, and environmental surveillance.

To use the software, a series of images of an area must be captured using a drone, aircraft or ground camera. The software processes these images, finds common landmarks and calculates the position and height of each point in the image.

In addition, Pix4D offers several powerful features to create accurate 3D models and detailed maps. Some of the key features include automatic image processing, providing an automated workflow in the calibration, alignment, and 3D model creation processes. From the 3D model information, high-resolution orthomosaic maps can be generated, which combine the images to create a panoramic view of the mapped area. These maps can be used to analyse terrain, measure distances and areas, and obtain detailed landscape information as well as providing advanced tools for data analysis and precise measurements. Users can measure distances, areas, volumes and even make accurate height measurements by calibrating the generated 3D models. Another important aspect is the various integration options with other software and platforms, allowing users to work efficiently and use the data generated by Pix4D with other tools.

3.1.4. Autodesk ReCap

Autodesk Recap is a software tool developed by Autodesk that allows users to capture, import and convert real-world data into high-quality 3D models. The software is designed to work with a wide range of data capture technologies, such as 3D laser scanning and aerial imagery, allowing users to obtain an accurate representation of real-world objects and environments.

AutoDesk ReCap Photo is part of the broader ReCap Pro subscription programme, which from 2019 is bundled with ReCap Pro. It limits the amount of processing a single user can perform by providing a limited number of 'cloud credits'; if the user needs additional credits for processing, he or she must purchase more.

Autodesk ReCap supports a variety of scan data formats, including Autodesk's RCP (ReCap Point Cloud) format, which was developed specifically for processing scan data in it. Once the data has been acquired, it can be imported into Recap to begin the 3D model creation process. The software supports both raw data files, such as point cloud files, and more structured data formats, such as laser scan files.

After importing the data, the SW offers a number of tools for cleaning and preparing the data. This is an important step, as the scan data may contain noise or unwanted points that could affect the quality of the final 3D model; this will remove unwanted points, filter out noise and improve the overall quality of the data. Once the scan data have been cleaned and prepared, you can proceed to create the actual 3D model using a variety of tools and features to create accurate and detailed 3D models.

One of Recap's key tools is the automatic alignment function, which allows multiple scans to be combined into one coherent pattern. This is particularly useful when scanning complex objects or environments.

3.1.5. Reality Capture

RealityCapture (RC) is photogrammetry software for creating 3D models from unordered photographs (terrestrial and/or aerial) or laser scans.

Reality Capture is a 3D processing software developed by the Slovak software house Capturing Reality, which was acquired in 2021 by the giant EPIC Games. Reality Capture enables the development of point clouds, 3D models, orthoprojections and much more from photographic images or laser scans.

In general, the most common fields of use are cultural heritage (art and architecture), whole-body scanning, games, surveying, mapping, visual effects (VFX) and virtual reality (VR) in general. Its functions include image registration (alignment), automatic calibration, polygon mesh calculation, colouring, texturing, parallel projections, georeferencing, DSM, coordinate system conversion, simplification, scaling, filtering, measuring and various exports and imports. Unlike other software, RC does not create orthomosaics, but creates orthoprojections from a textured 3D model (i.e. it will be necessary to develop a 3D model and texturise it to generate, for example, the elevation of a building). Thanks to the generation of very high quality 3D models and textures, the elevations generated by the 3D model's orthoprojections are of a very high standard and hardly distinguishable from a classic orthomosaic, which puts this software in a leading position in the creation of accurate and true-to-life 3D models.

3.1.6. Other photogrammetric software

In addition to the most performing software analysed in the previous paragraphs, several platforms exist on the market that are able to process 3D models from photogrammetric datasets, with some limitations related to the processing of final products (orthophotos, digital models, etc.). Among the most important of these is PhotoModeler, which is essentially a photo-triangulation programme capable of performing image-based modelling to produce 3D models and measurements. The software is also used for proximity, aerial and UAV photogrammetry (at relatively low altitudes) to perform measurements and modelling in agriculture, archaeology, architecture, biology, engineering, mining, storage volumes, etc.

Other commercial software are Trimble Inpho and WebODM. The former is suitable for high-precision transformation of aerial images into point clouds and surface models, orthophotos and 3D digital models, using various modules that can be integrated into any photogrammetric workflow and production. It is a standard software solution for aerial photogrammetry that has been used for large-scale metric mapping. The second, WebODM, is a user-friendly platform for the production of point clouds, elevation models, structured models and georeferenced maps using aerial digital images acquired from UAV systems. The software is a project of OpenDroneMap and supports several processing engines, currently ODM and MicMac. Purchase of the licence is free, but there is a one-off cost for installation and technical support.

3.2. Main open-source software

The scientific community is continuously developing open-source software. Indeed, the development of open-source software is an important research direction to maximise efficiency in the construction of 3D models, make interfaces user-friendly and reduce the need for further processing in other software to obtain the desired output, as well as to improve the performance of 3D information reconstruction in the case of objects characterised by complex geometries [104]. This type includes Regard3D, which is an SfM programme capable of creating 3D models of objects using a series of photographs taken from different viewpoints. The software, which is free and open-

source, is available on various platforms (Windows, OS X and Linux) and uses third-party tools and libraries. Various parameters allow models to be processed in a controlled and accurate manner. Furthermore, the actions required to process the 3D model are implemented in the software in a logical and sequential manner.

Another software that is widely used, especially in the academic world, is MicMac (Multi-Images Correspondances, Méthodes Automatiques de Corrélation) consisting of a free and open-source photogrammetric suite (Cecill-B licence) that can be used in various 3D reconstruction scenarios. The software is very versatile and can be used in various fields: cartography, environment, industry, forestry, heritage, archaeology, etc. In addition, it is possible to georeference end products in local/global/absolute coordinate systems and export orthophotos. Meshroom is a free and open-source 3D reconstruction software based on the AliceVision framework. AliceVision is a photogrammetric computer vision framework that provides state-of-the-art 3D reconstruction and camera tracking algorithms that can be tested, analysed and reused. The project is the result of a collaboration between academia and industry to provide state-of-the-art algorithms with the robustness and quality required for use in production.

3.2.1. Implementation of SfM pipeline

Over the years, several implementations of the SfM pipeline have been proposed, and among the most popular ones, there are dedicated platforms with publicly available source code, which can thus allow customisation of the SfM and MVS pipeline. Available pipelines include COLMAP, Theia, OpenMVG, VisualDFM, Bundler and MVE. In accordance with Bianco et al., 2018, a brief description of the pipelines is proposed below, followed by a detailed description of their implementations with the algorithms used in each processing block in Table 3.1.

COLMAP

COLMAP is a general-purpose Structure-from-Motion (SfM) and Multi-View Stereo (MVS) pipeline with a graphical and command-line interface. It offers a wide range of functions for reconstructing ordered and unordered image collections.

The tool has a simple graphical interface and an automatic reconstruction tool; however, it includes command-line options for more advanced users. The programme is also equipped with numerous tools and settings that make it suitable for various reconstruction scenarios. The C++ implementation has an intuitive graphical interface that also allows the configuration of pipeline parameters.

Theia

It is an open-source library of incremental and global SfM that contains many commonly used algorithms for feature detection, matching, pose estimation and 3D reconstruction. In addition, it is possible to extend the library with new algorithms using its software interfaces. The implementation is in the form of a C++ library, the executables can be compiled and then used to construct the reconstructions. The resulting sparse reconstruction can be exported in Bundler or VisualSFM NVM format that can be used by most MVS pipelines.

OpenMVG

An open-source library for solving multiple-view geometry problems, a Structure from Motion pipeline implementation is provided for both the incremental and global case, as well as several options for feature detection, matching, pose estimation and 3D reconstruction. It is also possible to use geographical data and GPS coordinates for the pose estimation phase. The library is written in C++ and can be included in a larger project or can be compiled into several executables, each for a specific set of algorithms.

VisualSFM

GUI application for 3D reconstruction using structure from motion (SfM). The reconstruction system integrates several of my previous projects: SIFT on GPU (SiftGPU), Multicore Bundle Adjustment and Towards Linear-time Incremental Structure from Motion. VisualSFM runs quickly by exploiting multicore parallelism for feature dedetection, feature matching and bundle adjustment. For dense reconstruction, this programme integrates the execution of Yasutaka Furukawa's PMVS/CMVS toolchain. VisualSFM's SfM output works with several other tools, including CMP-MVS by Michal Jancosek, MVE by Michael Goesele's research group, SURE by Mathias Rothermel and

Konrad Wenzel and MeshRecon by Zhuoliang Kang. Compared to other solutions, this one is less flexible because only one set of algorithms can be used to perform reconstructions. The software has an intuitive graphical interface that allows SfM to be configured and run.

Bundler

It is one of the first successful incremental implementations of SfM pipelines. It also defines a Bundler 'out' format that is commonly used as an exchange file between SfM and MVS pipelines.

MVE

Multi-View Environment is an incremental implementation of SfM, designed for multi-scale scene reconstruction; equipped with a graphical interface, it also includes an implementation of the MVS pipeline.

Linear SFM

It is a new approach to SfM reconstruction that decouples linear and non-linear components. The proposed algorithm starts with small reconstructions based on Bundle Adjustment that are then merged hierarchically. Table 3.1 and Table 3.2 below describes and compares in detail the implementations with the algorithms used in each processing block.

Tab. 3.1 - SfM pipelines algorithm comparison (extraction and matching features, geometric verification).

Software	Feature Extraction	Feature Matching	Geometric Verification	Image Registration
COLMAP	SIFT	Exaustive Sequential Vocubalary Tree Spatial Transitive	4 Points for Homography 5 Point Relative Pose 7 Points for F-matrix 8 Points for F-matrix	P3P EPnP
OpenMVG	SIFT AKAZE	Brute Force ANN Cascade Hashing	Affine Transformation 4 Points for Homography 8 Points for F-matrix 7 Points for F-matrix 5 Point Relative Pose	6 Point DLT P3P EPnP
Theia	SIFT	Brute Force	4 Points for Homography	P3P

		Cascade Hashing	5 Point Relative Pose 8 Points for F-matrix	PNP (DLS) P4P P5P
VisualSFM	SIFT	Exaustive Sequential Preemptive	n/a	n/a
Bundler	SIFT	ANN	8 Points for F-matrix	DLT based
MVE	SIFT + SURF	Low-res + Exaustive Cascade Hashing	8 Points for F-matrix	P3P

Tab. 3.2 - SfM pipelines algorithm comparison (triangulation and BA process).

Software	**Triangulation**	**Bundle Adjustment**	**Robust Estimation**
COLMAP	sampling-based DLT	Multicore BA Ceres Solver	RANSAC PROSAC LO-RANSAC
OpenMVG	linear (DLT)	Ceres Solver	Max-Consensus RANSAC LMed AC-Ransac
Theia	linear (DLT) 2-view Midpoint N-view	Ceres Solver	RANSAC PROSAC Arrsac Evsac LMed
VisualSFM	n/a	Multicore BA	RANSAC
Bundler	N-view	SBA Ceres Solver	RANSAC
MVE	linear (DLT)	own LM BA	RANSAC

3.3. Outputs and intermediate products

3.3.1. Digital models

A digital elevation model (DEM) is a 3D representation of a surface in digital format. The main characteristic of a DEM is therefore that it contains information on the heights of individual points on the earth's surface, represented as pixels in an image. Three different types of digital models can generally be distinguished:

- DSM (Digital Surface Model) is a digital surface model consisting of the topographical surface with all elements, natural or man-made, rising from the ground (e.g. buildings, trees, bridges, etc.).
- DTM (Digital Terrain Model) is a digital terrain model consisting of the topographic surface. A DTM can be described as a three-dimensional representation of a terrain surface consisting of X, Y, Z coordinates stored in digital form. It includes not only heights and altitudes but also other geographical features and natural characteristics such as rivers, ridge lines, etc., so it is as if what is seen from above is dissected at ground level.
- DEM (Digital Elevation Model) is usually used in a generic sense and includes all elevation models, regardless of the surface they consider (DSM, DTM).

The globally most widespread and well-established method for recording elevation information consists of arranging the elevations within a regular grid (raster or grid), which can be represented with a numerical matrix, an ideal input for management within a computer. Usually the grid has a square mesh, more rarely triangular or rectangular, whose side size provides the cell size (cell size or pixel size), which corresponds, once the projection is fixed, to the spatial resolution of the DEM (mesh size). The height resolution, on the other hand, defines the minimum numerical range with which height values are represented.

The progression from conceptual model to DEM requires the selection of suitable procedures but also the application of appropriate measurement and statistical

processes. The numerous steps required for the construction and analysis of a digital elevation model lead to the formation of a series of errors related to the measurement and/or data processing processes and contribute to the formation of a series of errors or model uncertainties (uncertainty). Errors can be attributed to momentary malfunctioning of the acquisition equipment or defined as the result of a deterministic system which, if known, can be represented by a functional relationship. For example, the contour ghosts found in many DEMs produced by isohypses can manifest themselves as a terraced slope effect, or artefacts due to the parallel striping typical of various DEMs (e.g. SRTM). Other random errors may result from a wide variety of operations performed during the construction of the DEM and are found on the model according to a random distribution. The most commonly used quantitative description of the error is the calculation of the Root Mean Square Error (RMSE) according to the following formula:

$$RMSE = sqrt\left(\frac{\Sigma\left(z_i - z_j\right)^2}{n}\right) \tag{17}$$

where:

z_i share of the DEM.

z_j measurement to be used as a reference.

n number of samples considered.

If the number of values considered is large enough to be significant, the measurement error of the original dataset can be statistically quantified. The resulting value provides a measure of how far the values of the two datasets deviate. Assuming that the mean of the errors is close to zero, its significance is close to the standard deviation of the error distribution; the higher the value taken by the RMSE, the more the two datasets differ.

In general, therefore, a digital elevation model (DEM) is a 2.5D model of a surface represented as a regular grid, with height values stored for each grid cell. Within photogrammetric software, DEMs can be rasterised from a dense point cloud, a sparse cloud, a mesh or generated directly from depth maps.

By means of point cloud classification algorithms, such as *ATIN - Adaptive Triangulated Irregular Network, PTIN - Progressive Triangulated Urregular Network* or *MCC - Multi Curvature Classification*, it is possible to classify the point cloud (Łącka, M., 2021) into other semantic classes and process the digital terrain model (DTM), which represents the bare ground surface without any objects or man-made elements such as plants and buildings.

3.3.2. Orthophotos

The orthophoto is one of the most important basic geographic information products, i.e. it is a photo that has the properties of constituting an orthographic projection.

The orthophoto, also called orthophotomosaic, or orthophotomosaic, is one of the 2D outputs that can be obtained as a result of terrestrial or aerial photogrammetry. It is a rather complex composition of photograms, geometrically corrected through a process known as orthorectification. In addition to correcting it, it also georeferences it, so that the scale of representation becomes uniform throughout. At the end of this process the orthophoto, in the case of a UAV survey of vast areas, can be given the same value as a real map. The construction, processing and generation phases of an orthophoto can be summarised in the following pipeline (Fig. 3.1).

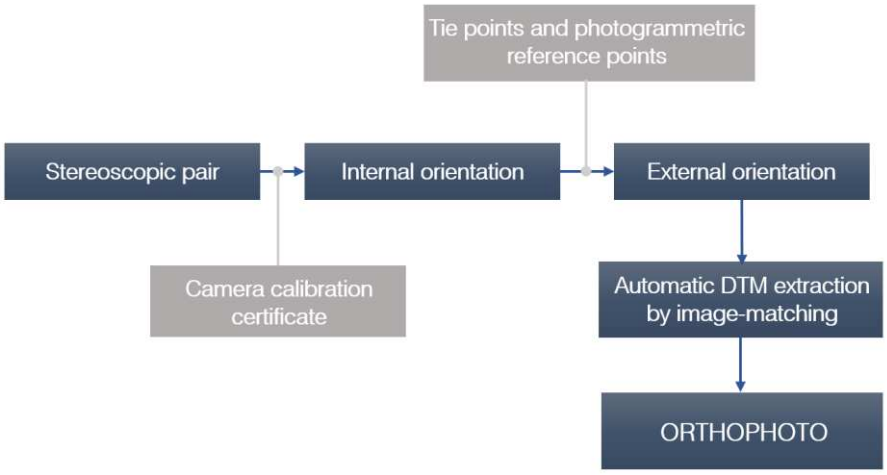


Fig. 3.1 - Processing steps of a digital orthophoto.

For the production of a digital orthophoto it is therefore first necessary to define the ortho-image matrix, in which each pixel corresponds to an element of the ground plane; for each pixel of the ortho-image, an elevation value is extracted from the relative digital elevation model (DEM). Then, the internal and external orientation parameters being known, the ground point is projected in the 3 coordinates (X,Y,Z) through the collinearity equations obtaining the coordinates of the *xy* frame. These *xy coordinates* are transformed into the image coordinates *ij* by means of a two-dimensional affine transformation. The density value corresponding to the calculated coordinates *ij is* found on the image by interpolation and this density value is assigned to the pixel on the ortho-image by one of the resampling methods. It should be noted that the extraction of an elevation value from the DEM is based on interpolation, since the resolution of the ortho-image is usually greater than that of the DEM; however, this solution has the disadvantage that the collinearity equations must necessarily be solved for each pixel on the orthophoto. In the process of orthophoto generation, the factors concerning the source digital image that most influence the quality of the final product are the scale of the image and the scanning interval, i.e. the resolution of the image if it was acquired directly.

3.3.3. The True Orthophoto

With the rapid development of photogrammetric technology and the increasing demand for high-precision graphics, real orthophotos have become a major research topic. Conventional orthophotos are differentially corrected using a 2.5-dimensional digital elevation model (DEM), ignoring the three-dimensional ground objective. The problem in this case, which is especially noticeable in the production of large areas, where there are a large number of buildings or large variations in height, is that the upper part is corrected and does not coincide with the lower part, resulting in a difference in projection in the corrected orthophoto. Due to the central perspective that gives rise to the frame, there is, in the area hidden by the building, an effect known as double mapping, as the image point is represented on the orthophoto by two points. This error

can only be eliminated with the precision orthophoto generation procedure or True Orthophoto, which involves orthorectification from several images (Fig. 3.2); missing details in the hidden areas can then be taken from frames acquired from other viewpoints.

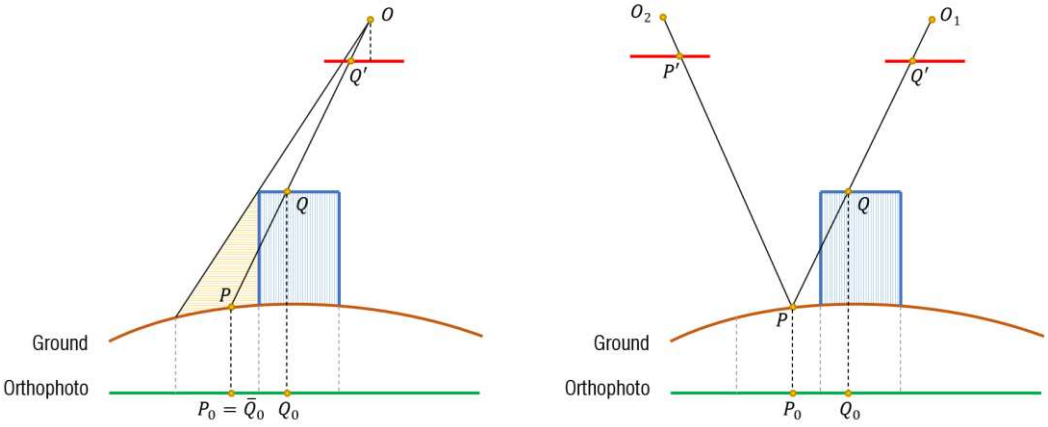


Fig. 3.2 - True Orthophoto: problem of hidden areas and the image orthorectification process.

To correct these projection difference issues, a 'true orthophoto' has been developed over the years that eliminates the tilt error and projection difference, requiring a direction and side overlap of at least 80 per cent and 60 per cent and a base ratio (B/H) of less than 0.3. The production of a true orthophoto can be divided into the following stages: i) acquisition of high-resolution data, DSM and DBM; ii) detection and identification of occlusions and shadows appearing in repeated mapping; iii) elimination of edge effects and outliers.

In general, the difference from traditional orthophotos is that the DSM replaces the DEM in the correction of traditional orthophotos when digital differential correction is performed. Perspective distortion caused by ground fluctuation and the difference in height between the object on the ground and the ground surface are corrected. In fact, differential correction using the DEM will produce a projection difference and perspective distortion, while differential correction using the DSM will cause repeated mapping in the occlusion area of the object (Li et al., 2020).

In order to make the orthophotos highly performant and without projection errors, there are several algorithms for rectification that are briefly described below.

Z-Buffer algorithm

The point of the feature on the photographic light itself is only one ground point closer to the centre of the photograph is not occluded, and the rest are occluded. The occlusion situation is determined by calculating the distance of each point on the photographic light from the centre of the photograph. First, based on the DBM and DTM, respectively, the correction is performed separately and then the two corrected images are combined to obtain a true orthophoto. The advantage is that the algorithm is simple and easy to implement.

Angle-based shadow detection algorithm

In the same direction of projection, based on the ground point, the angle between the photographic light and the vertical is proportional to the distance between the ground point and the photographic point; when the occlusion area is encountered, the elevation value due to the occlusion becomes smaller, causing the angle to become smaller, and up to the visible area, the projection angle is again restored. Occlusion is judged by comparing the change in angle between the photographic light and the plumb line. In this case, the calculation process is complicated, the amount of calculation is high and the efficiency is low.

Height-based shadow detection algorithm

The algorithm compares the height of the ground points on the search path with the height of the rays for shadow detection. In addition, there are several other algorithms such as *Shadow Detection Algorithm* Based *on Vector Polygon Inversion Imaging* based on the projection of polygons to obtain the image polygon or, for example *Based Texture Matching algorithm that* is based on a global variational model and texture matching, which further improves the interpretation capabilities and the aesthetic effect of the image. Finally, another algorithm that improves the final output of orthophotos is *object-oriented occlusion compensation algorithm that is* able to maintain the accuracy of geometric features, texture structure integrity and a good visual effect.

4. COMPUTER GRAPHICS: STRUCTURING AND CLASSIFICATION OF POLYGONAL MESH MODELS

A polygon mesh, in Computer Graphics, is a grid that defines an object in space, consisting of vertices, edges and faces. The term mesh in English literally means 'mesh', 'net'. Polygon meshes are increasingly being used in many different fields of computer graphics and geometry processing (a relatively new field of computer science involving algorithms and mathematical models for analysing and manipulating geometric data). The most commonly used meshes in computer graphics are *triangle meshes* and *quadrilateral meshes* (quad meshes). In general, when reference is made to 'mesh', this means a triangle mesh. Other mesh types, in which the basic elements are quadrilaterals or other polygons, are sometimes used in different contexts and applications. The processing of these types of mesh, if not properly controlled, may lead to possible approximations of the original three-dimensional model and may generate surfaces that are not easy to control (for example, it is easy to create a quadrilateral whose vertices do not all lie on the same plane, while there is always a plane passing through three vertices).

Furthermore, working exclusively with triangle meshes simplifies storage, and reduces the number of algorithms, but, in applications that go beyond simple three-dimensional representation, could generate various problems in the storage and management of models, as well as affect the processing in terms of computational resources and hardware memory usage.

4.1. Surfaces and categorisations

4.1.1. Definitions and main components

A mesh, unlike a real solid object, has no mass; it is therefore a kind of empty volume, devoid of thickness, whose faces are surface 'veils'. Intuitively, a polygonal mesh is the partition of a continuous surface into polygonal cells, such as triangles, quadrilaterals, etc. More formally, a mesh can be defined as a tuple $(V; K)$ where:

- $V = \{v_i \epsilon \mathbb{R}^3 | i = 1 \dots N_v\}$ is the set of vertices of the model (points in $\mathbb{R}^3$)
- K contains the adjacency information, i.e. how the vertices are connected to form the edges and faces of the mesh.

For example, a mesh composed of a single triangle would be:

$$\{v_0, v_1, v_2\}, \{\{v_0, v_1\}, \{v_1, v_2\}, \{v_2, v_0\}, \{v_0, v_1\}, \{v_0, v_1, v_2\}\} \quad (18)$$

i.e. the three vertices, the three edges and the triangle.

The visible components of a mesh, as shown in Fig. 4.1 below, are:

- vertex, i.e. the point in space with coordinates x, y, z which determine its position.
- edge, i.e. the segment joining two vertices in space.
- face, defined by connecting and closing at least three edges.

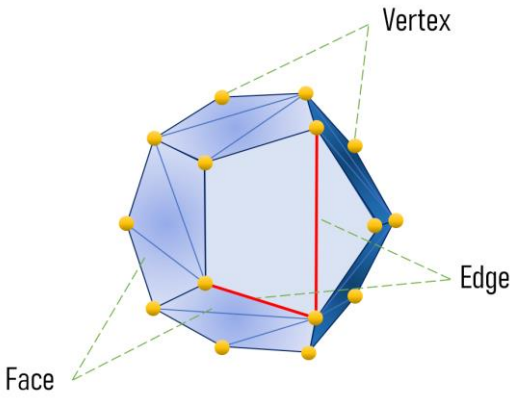


Fig. 4.1 - Representation of mesh components.

The set of all neighbours of a vertex is called the 1-ring of the vertex and is defined as:

$$v_i(i) = \{j | \{i, j\} \epsilon K\} \quad (19)$$

The cardinality of $v_i(i)$ is called the degree or valence of the vertex v_i.

A sequence of adjacent triangles that share the same vertex is called a triangle fan. A strip is a sequence of triangles that can be specified by listing its vertices unambiguously. To be more specific, given an ordered list of vertices:

$$\{v_0, v_1, \dots, v_n\} \tag{20}$$

the triangle i is represented by the vertices:

$$\{v_0, v_{i+1}, v_{i+2}\} \tag{21}$$

Strips and fans are used to compact mesh representations. A strip of triangles with n vertices represents $n-2$ triangles. Thus, for example, a strip of 100 triangles requires 102 vertices to be stored, rather than 300. The quantity of vertices increases with the number of triangles; the average number of vertices $\overline{v_t}$ required to represent a triangle in a strip with m triangles is $\overline{v_t} = 1 + 2/m$.

In the case of a fan, the triangle i is represented by the vertices $\{v_0, v_{i+1}, v_{i+2}\}$ assuming that v_0 is the shared vertex (Ganovelli et al., 2014).

4.1.2. Types and orientation

A finite 2D mesh is a manifold if the edges and triangles meeting a vertex v can be ordered in a cyclic order $t_1, e_1, t_2, e_2, \dots, t_n, e_n$ without repetition of the triangles t_i, t_{i+1}. This implies that for each edge, there are exactly two faces containing it.

A surface is said to be 2-manifold (two-dimensional-manifold) if the neighbourhood of each point p on the surface is homeomorphic to a disc. More simply, it means that if we have a disc, we can centre it in p and make it adhere to the surface around it. The definition of 2-manifold is extendable to the surface with frontiers, considering cutting half the disc and making it adhere to the frontier.

If the surface is a polygonal mesh, we can determine whether it is a manifold by checking whether the following conditions are true:

- **Edge Manifold** each edge is shared by one face (meaning it is on the mesh border) or two faces.
- **Vertex Manifold** if two-sided f_a e f_b share a vertex, then we can move from f_a a f_b traversing only the edges in the 1-ring of the vertex (i.e.

we can move around the vertex without passing through the vertex itself).

Furthermore, each face consists of a polygon and therefore has two sides. If we observe a face and follow its vertices in the order in which they are specified in Kthey can describe clockwise or anti-clockwise motion. Obviously, if the same faces are observed from the inside, these orientations will be reversed.

The definition of orientation is useful for the determination of the normal vector, which is necessary in 3D modelling processes; if the vertices of a non-degenerate triangle (i.e. one with a non-zero area) are at the points $P_i,\ P_j,\ P_k$, then it is possible to calculate the normal vector in the plane of the triangle as:

$$P_\perp = (P_j - P_i) \times (P_k - P_i) \tag{22}$$

The order relative to the position of the vertices is of fundamental importance, since if one swaps the position of the vertices $P_i,\ P_j,\ P_k$, the resulting vector is negative; this erroneous result could make the process of modelling and analysing the mesh critical, as the use of normals in this context is linked to the determination of the elements that are inside and/or outside the surface itself (see Fig. 4.2).

A mesh is defined as closed if there are no holes on its surface, otherwise it is customary to say that the mesh is open. In these cases, all elements (edges, vertices..) that are found on the edge are called *boundary*.

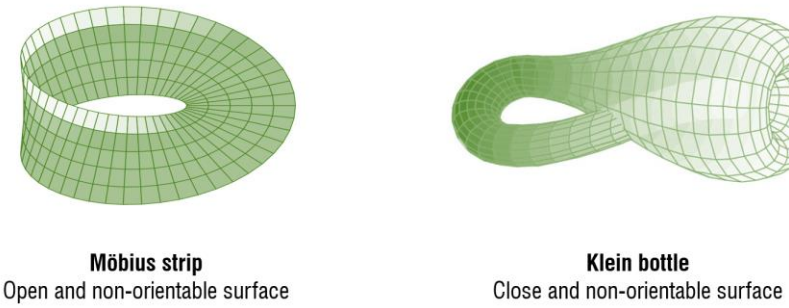


Fig. 4.2 - Example of non-orientable surfaces.

The local topology of a face, side or vertex of the mesh, on the other hand, describes how that element is connected to the rest of the mesh, i.e. which vertices, sides or faces are part of it or incident in it.

The mesh is of the 2-manifold type if locally every point on the mesh surface is topologically equivalent to a unit disc. If the mesh respects these characteristics, it is defined as *consistent*.

In the case of triangular meshes (see Fig. 4.3) this disc is a ring of triangles surrounding the point/vertex, if the latter is boundary (we are therefore talking about open meshes) then it only needs to have a half-ring around it. In a 2-manifold triangular mesh each side has exactly two incident faces. A 2-manifold with boundary allows for some sides with only one incident triangle. This type of mesh is relatively simple to handle.

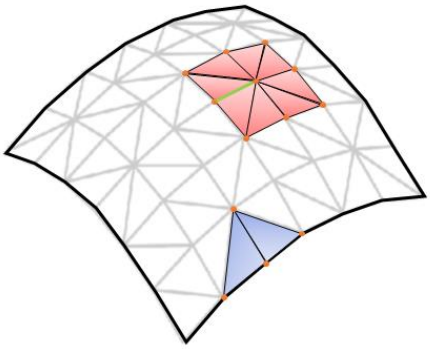


Fig. 4.3 - Structure of a mesh: in green the vertices, in red the inner edges, in blue the boundary zones.

There are of course also non-manifold meshes, i.e. meshes whose structure has incorrect characteristics, such as T-joints, edges shared by more than two faces or vertices shared by several surfaces.

In addition to topology, we also have other information associated with meshes:

- geometry, via the coordinates of the vertices,
- orientation, by ordering the vertices of each face.

4.1.3. Structuring and representation

The choice of mesh data structure is of paramount importance as topological and algorithmic factors must be considered (Botsch et al., 2010).

Among the topological requirements to be considered in the realisation of a mesh, one must take into account certain factors concerning the type of representation and the relative data structure, and consequently define what type of mesh is required to represent the data structure. In fact, it may be possible to refer only to 2-manifold meshes, or to need to represent complex edges and singular vertices. The Figure 4.4 below summarises the main topological requirements that must be considered when modelling a surface.

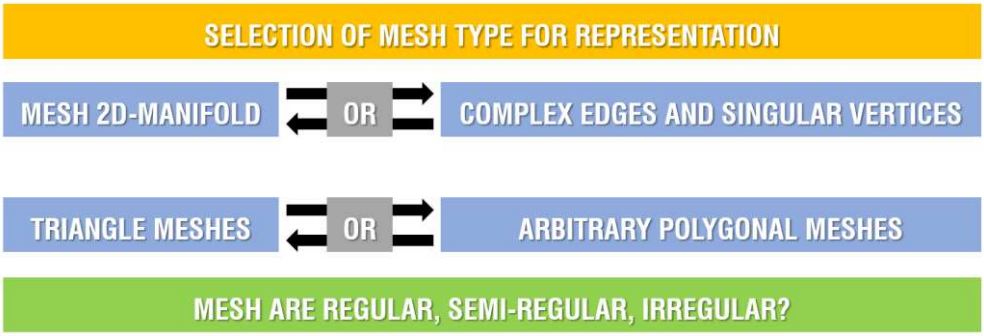


Fig. 4.4 - Topological requirements for modelling a surface.

In addition to the topology, other elements to consider are the algorithmic requirements, i.e. the set of algorithms that will operate on the data structure. This includes all aspects such as the rendering phase of the mesh, accuracy and thus efficient access to local contours of vertices, edges and faces. Furthermore, algorithmic requirements also include all operations related to the need to associate data with vertices, edges and faces, as well as defining whether the mesh will be static or whether its geometry and/or connectivity will change over time. Obviously, these factors influence the computational capabilities of computers, the specific requirements of the processors used in terms of memory consumption (i.e., the data sets are massive).

4.1.4. Polygon mesh representations

The ability to efficiently manage the information characterising the mesh object during the modelling process depends on the choice of data structure. If one only wants to represent the object, the information in a Wavefront OBJ file is sufficient. If, on the other hand, mesh editing operations are to be carried out, well-organised data structures are required, designed to contain additional information, striking a balance between the amount of data stored and the speed of access to this data and its updating. When representing an object by mesh, it is important to consider how the polygons are actually structured. In fact, there are many representations, and the choice of one type over another greatly influences the processing possibilities of the mesh. The simplest solution is to store each polygon with the geometric position of each of its vertices. However, this operation does not allow for any information on connectivity, resulting in a set of polygons that can be rendered but cannot be subsequently modified; in fact, all primitive operations start from an a priori knowledge of how polygons relate to one another. For this reason, there are different types of representation (Botsch et al., 2007), briefly described below.

Indexed face set

A more useful and simple representation is the indexed face set. An indexed face set stores the mesh in two arrays. The first contains all vertices indexed by a number (typically only in a linear array). For each vertex, its attributes are stored, i.e. at least its position in space, but frequently also other information such as a vertex normal, and some attributes needed for special purposes such as rendering. The second array is a list of faces containing at least one list of indices, where each index refers to the vertex array.

Edge-based data structure

The principle of the edge-based data structure is to represent connectivity by explicitly storing how edges relate to one another. For each face of the mesh, one simply stores a pointer to only one of its edges. Similarly, only one pointer to a single edge is stored for each vertex. If the next edge in the edge loop defining a face is to be defined, simply follow a pointer from the current edge. Consequently, it is possible to

circle a face, visiting all its edges even if the face only has one pointer. Since edge-based representations are based on edges that know their incident faces, only manifolds are usually representable using these data structures.

Winged edge data structure

A winged edge is an oriented edge connecting two vertices. For each of its two face loops, it has two pointers to subsequent edges, either clockwise or counterclockwise. The winged edge is a complicated representation type, it has the additional disadvantage that the edge itself is bidirectional but the winged edge is oriented. If we need to move from one edge to another, the pointer we have to use depends on the orientation of the edge (Fig. 4.5).

Half edge data structure

The half-edge representation presents two representations of a given edge. The half-edges are in pairs, and each element of the pair represents the edge from the point of view of one of the two polygons sharing the edge; hence there is no ambiguity.

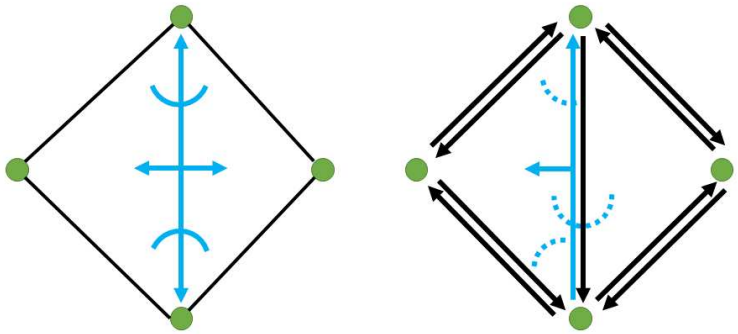


Fig. 4.5 - Comparison of Winged-Edge and Half-Winged-Edge Structures.

Each half-edge has a next pointer, which points to the next edge in the loop that corresponds to its face. By simply following the next pointers, it is possible to visit all the edges in the edge loop of a face.

A comparison of connectivity structures (indexed and half-edges) is shown in Table 4.1 below.

Tab. 4.1 - Connectivity structures compared.

INDEXED	HALF-EDGES
HW friendly	Complicated rendering
Complicated navigation i.e., requires an additional data structure	Simple navigation
Pure meshes formed from tri-mesh or quad-mesh only	Mixed polygonal meshes
Suitable for renderings	Suitable for mesh processing operations

4.2. 3D reconstruction algorithms

Triangular meshes are a common form of representation of 3D models used in computer graphics. Various algorithms and techniques exist for generating triangular meshes; in general, the process leading to the definition of a mesh model is based on the following process.

1. **Geometry definition**: in this phase, the geometry of the 3D object is defined and can be achieved by specifying co-ordinate points in three-dimensional space, (called vertices). Normals, which indicate the orientation of the surfaces, and texture co-ordinates can be defined if required.
2. **Creation of triangles**: once the vertices are defined, the triangles are processed using a triangulation algorithm (Delaunay's or Ear Clipping). These algorithms take vertices as input and connect them to form non-overlapping, non-intersecting triangles. The result is a 'network' of triangles called a mesh.
3. **Mesh optimisation**: The generated mesh may contain unwanted triangles or have an uneven density of triangles. In this case, optimisation techniques can be applied to improve the quality of the mesh. These techniques may include removing unnecessary triangles, splitting or re-arranging triangles to obtain a more uniform distribution.
4. **Assigning Properties**: Once the mesh has been generated, you can assign properties to the vertices and triangles. For example, you can assign materials, colours, textures, or other information that you wish to associate with the mesh.

It is important to note that the triangular mesh generation process may vary depending on the specific application or tools used.

In addition to Delauny triangulation, there are other methods for creating triangles; the choice of method will depend on the context and the specifics of the problem to be solved, the most commonly used ones are listed below.

- Subdivision of an existing mesh: This method starts with a starting mesh, which may be a simple grid or a mesh with a more complex structure. Subsequently, subdivision operations are iteratively applied to refine the mesh and generate smaller triangles. Examples of subdivision algorithms are the Catmull-Clark and the Loop.
- Procedural generation algorithms: These algorithms generate meshes based on predefined rules and parameters, allowing complex geometric structures to be created. For example, the fractal algorithm, such as Voronoi's fractal, can be used to generate triangular meshes of terrain or landscapes.
- Optimisation algorithms can be used to generate optimal triangle meshes for certain purposes. These algorithms attempt to minimise a cost function that takes into account factors such as triangle quality, mesh adaptivity and conformity to specific constraints. An example of this type of algorithm is free-form optimisation.

Some algorithms for 3D reconstruction from a point cloud will be illustrated below. These algorithms are also contained within the PCL - Point Cloud Library, an open-source library used for processing three-dimensional point clouds. PCL is designed to provide a set of tools and algorithms for the acquisition, processing, visualisation and manipulation of point cloud data from 3D sensors such as lidar, laser scanners and ground cameras. PCL is written in C++ and was developed to be highly efficient and modular. It offers a wide range of functionality, including filtering, segmentation, resampling, alignment, feature extraction, object recognition, surface reconstruction and more. PCL supports numerous point cloud data formats, allowing users to import and export data to and from different sensors and software. It is also

integrated with other popular frameworks and libraries such as OpenCV, providing simplified interoperability with other tools and components.

The open, autonomous and large-scale project was started in 2010 and is released under the terms of the BSD licence and is therefore free for commercial use and research applications.

For simplified use, PCL is divided into a series of modular libraries covering different topics, from simple point cloud input/output operations to more complex applications including visualisation, feature estimation, segmentation, etc. (https://pointclouds.org/).

4.2.1. Greedy Projection Triangulation

Greedy projection triangulation is an iterative algorithm (Marton et al., 2009) that starts by joining three points in an initial triangle and then connects other adjacent triangles until all points have been taken into account. It was developed in an attempt to optimise meshing processing times. There are different types of Greedy algorithms such as Ball-pivoting (Bernardini et al., 1999) and the algorithm developed by Petitjean and Boyer (Petitjean and Boyer., 2001); they all follow the same principle by prioritising different factors. Greedy relies on distance to work out whether a point belongs to the previous surface or to another, and only joins surfaces that reflect certain parameters that can be set by the user, such as the maximum angle between the normals of two surfaces and the maximum length of the sides of the triangle (De Loera et al., 2010); it may happen that there are points left over that are not part of the surface that was generated, in which case, if it is possible, the algorithm generates a new starting triangle and starts expanding it again. This process is repeated until the starting triangles can no longer be created. The maximum distance a point must be from the nearest triangle is one of the parameters to be set in order to obtain an optimal result.

One of the main features of this method is that it does not modify or delete the points it receives as input, nor does it interpolate with surfaces. This allows for maximum precision in the creation of surfaces. Furthermore, if the density of the points is variable, this is not a problem, but the algorithm is able to adapt to different densities.

If the point clouds are affected by noise or outliers, the precision in the determination of the normals to the triangles is altered and this may affect the final result.

In order to overcome this problem, it is necessary to increase the radius within which the vertices of the triangles are defined. In this way, an attempt is made to assign a normal to the triangle as close as possible to that of the neighbouring triangles. While this operation solves the problem of normals on a flat surface, it may not be efficient in the case of edges, which may be blunted after this process if not significantly modified.

Through the 'Maximum Number of Neighbours' algorithm, it is possible to improve the final result of mesh processing, in terms of storage memory, and it is possible to set the maximum number of neighbours that will be joined to the triangle, obviously giving priority to the points closest to the surface.

4.2.2. Organised Fast Mesh

The Organised Fast Mesh algorithm is a high processing speed for point clouds. It is able to reconstruct surfaces in real time from organised point clouds, i.e. set up as a structure (or matrix), where the points are ordered in rows and columns. Due to the speed of this algorithm and the limited possibility of setting parameters, the mesh obtained is of poor quality and in any case worse than when using other algorithms. The only parameter that significantly influences the final result is the maximum side length of the polygons.

4.2.3. Poisson surface reconstruction

Another method for reconstructing a surface model used in computer graphics is the Poisson method; in particular, this method is used to generate meshes of regular, smooth surfaces. It is named after the French mathematician Siméon-Denis Poisson, who contributed to the development of vector field theory.

The Poisson method for meshes starts with a set of sample points placed on the surface of interest. These sample points can be distributed uniformly or according to a specific criterion, such as the density of the desired detail. The first step is to calculate a scalar function defined on each sample point, which represents the distance

of the actual surface from that point. This function is called the sample distance function or sample distance field. Next, a mesh is generated from the sample points using triangulation techniques, such as Delaunay's delimiting triangle. This initial mesh may be irregular and not conform to the desired surface.

The next step is to apply Poisson's method to approximate the real surface on the generated mesh. This is done by solving a partial differential equation (EDP) problem called the Poisson equation, which involves the sample distance field as a known term. The objective is to find a harmonic solution of the scalar function on the mesh, which represents a smooth surface conforming to the sample points.

The solution of the Poisson equation can be achieved by using various numerical approaches, such as discretizing the equations and applying iterative methods to solve the resulting system of equations.

Once the solution of the Poisson equation is obtained, the mesh is updated by changing the coordinates of its vertices according to the values of the solution. This process is iterated until the mesh converges to a regular, approximate surface of the desired shape.

The Poisson method for meshes is widely used in various applications, such as 3D modelling, reconstruction of surfaces from scan data and creation of surface details. It offers an effective way to generate smooth meshes that are consistent with the input data, ensuring accurate representation of complex surfaces.

In the most general form (Kazhdan et al., 2006), given as input a set of points S consisting of samples of s.p. points, with $s \in S$ and normal $s.N$ inwards, which are assumed to lie on or near a surface ∂M of an unknown solid model M unknown; the surface is reconstructed by solving a standard Poisson problem:

$$\Delta \tilde{X} = \nabla \cdot \vec{V} \tag{23}$$

where:

$\vec{V}$ is the vector field and gradient of the smoothed indicator function X_M of M is equal to the vector field obtained by smoothing the field normal to the surface:

$$\nabla(X_M * \tilde{F})(q_0) = \int_{\partial M} \widetilde{F_p}(q_0)\vec{N}_{\partial M}(p)dp \tag{24}$$

where:

$\vec{N}_{\partial M}$ normal to the surface at point $p \in \partial M$;

$\tilde{F}(q)$ smoothing filter;

$\widetilde{F_p}(q_0) = \tilde{F}(q - p)$ translation to the point p.

The surface integral cannot be solved because the geometric surface is not known, so the integral is approximated to a discrete summation. The approximation method consists of using the set of input points S to divide ∂M into discrete patches $Ps \subset \partial M$and then approximate the integral on a patch Ps to the value of the s.p. sample point, scaled by the area of the patch:

$$\nabla\left(X_M * \tilde{F}\right)(q_0) = \sum_{s \in S} \int_{Ps} \widetilde{F_p}(q)\vec{N}_{\partial M}(p)dp \approx |P_s|\widetilde{F_{s.p}}(q)s.\vec{N}_{\partial M} \equiv \vec{V}_q \tag{25}$$

A reworked graphical representation (Kazhdan et al., 2006) of the Poisson surface reconstruction method applied to a generic figure is shown in the next figure.

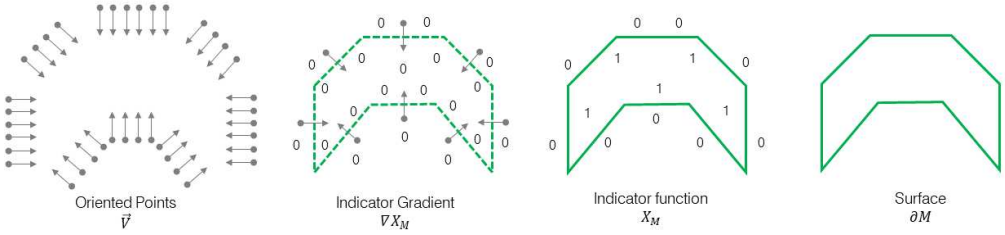


Fig. 4.6 - Illustration of Poisson reconstruction adapted from Kazhdan et al. 2006.

Poisson's method and Delaunay triangulation (cfr. 2.3.1) are two distinct approaches used in mesh reconstruction, with some significant differences.

The Poisson method is an algorithm used to reconstruct meshes from a set of points sampled in three-dimensional space. The main objective of Poisson's method is to create a smooth mesh surface that approximates the input data. The algorithm uses

the theory of partial differential equations to estimate the normal vector field of the input data and then generates a mesh with this information. In fact, this method is particularly suitable for reconstructing objects with smooth, well-defined surfaces.

Poisson's method focuses on creating a smooth surface based on an estimated normal vector field, while Delaunay triangulation focuses on generating non-overlapping triangles with circumscribed empty circles. Delaunay triangulation is, in fact, particularly useful for mesh generation where it is important to obtain a good distribution of triangles and maintain the quality of geometric shapes.

The choice between the two methods depends on the type of input data and the objective of mesh reconstruction.

4.2.4. Marching Cubes

Marching Cubes is an algorithm (Lorensen et al., 1987) implemented in the PCL library (Kazhdan et al., 2007), which requires an isosurface as input. Before using it, it is necessary that the point cloud has been previously processed by another algorithm that outputs an isosurface such as the Poisson Surface Reconstruction Algorithm. In this case, a 'divide and conquer' approach is used to arrive at the final mesh: the surrounding area or isosurface is divided into cubes called voxels, which will contain part of the surface inside. Initially it only recognises how the isosurface intersects the sides of the voxels; the algorithm reconstructs the surface as it is positioned within the cube. At points where the surface changes with a greater gradient, the size of the voxels is halved one or more times, depending on how quickly the point normals change. If the voxels are too large, there is a risk of losing much of the information they contain, which is why the so-called "octree" is used, i.e. a tree that keeps track of the number of subdivisions the voxels have undergone and thus their relative size. Each node in the octree corresponds to a voxel and contains information on the number of points within it.

Based on this assumption, it can be said that a greater number of subdivisions leads to greater accuracy in reconstruction, but at the same time also to greater computational complexity. For large quantities of points, the time may increase considerably. The possibility of setting the input parameters may allow a more accurate result: it

is possible to increase the number of subdivisions in the vicinity of edges and conversely, decrease it at faces where the variation of the surface normal changes little from one point to the next.

The intersection of the surface with a cube can take place in many ways. Starting with the voxels, the algorithm identifies the interpolation mode by placing the voxels side by side and attempting to join the various surfaces within the cubes into a single final surface, which will then be the output of the algorithm. The most important parameters at this stage are the 'Depth' which allows the maximum height of the octree to be modified: higher values provide a more accurate reconstruction conversely lower values improve processing time. Another parameter is the "Samples per node" which allows you to set how many points will be contained in each cube before it is divided into smaller cubes. Decreasing the value of this parameter allows the isosurface to be subdivided into many more cubes, thereby increasing the relative processing time and somewhat decreasing the effect of noise. The following Table 4.2 summarises the advantages and disadvantages of the algorithms used to define mesh surfaces from a point cloud and proposed in this section.

Tab. 4.2 - Advantages and disadvantages of the algorithms used to define mesh surfaces.

Algorithm	Advantages	Disadvantages
Greedy Projection Triangulation	Can manage colours Can manage variable dot densities High accuracy	Trade-off between accuracy and polygon size
Organized Fast Mesh	Extremely fast	Does not manage colours Needs point cloud organised
Poisson surface reconstruction	Smooth, consistent and well-defined surfaces	Does not manage colours
Marching Cubes	Allows choice of precision Final result without outliers and noise	Does not manage colours Used in conjunction with the Poisson algorithm

4.2.5. Smoothing algorithms

In order to improve the final quality of the mesh, several algorithms exist in the literature that are capable of performing smoothing operations consisting of changing the position of nodes or sometimes, eliminating some of them altogether. In some cases, node positions may also be fixed, either per specific node or per geometric entity, in order to limit the application of smoothing to non-fixed nodes. The objective of smoothing is to change the position of vertices to obtain a better mesh by eliminating noise. The application of such processes also aims to preserve the volume and features of the mesh, and to make this operation independent of the mesh type.

As with the meshing algorithms, a variety of smoothing algorithms are available (Erten et al., 2009), some of which apply to multiple types of geometric entities and others only to a specific type. Some of the most commonly used smoothing algorithms are listed below.

SMOOTHING LAPLACIAN

In Laplacian smoothing (Field et al., 1988), a free vertex is simply repositioned to the centroid of the vertices connected to that vertex, as described in the following equation.

$$x^* = \frac{1}{k} \sum_{x_j \in \Omega_i, x_j \neq x_i} x_j \tag{26}$$

where Ω_i is the vertex star x_i with k points and x^* is the new position.

This technique is widely used for its simplicity and effectiveness. However, it does not guarantee an improvement in quality when inverted elements are generated. Continuous iterations of Laplacian smoothing collapse or degrade the mesh. For this reason, many studies have introduced different versions of Laplacian smoothing or combinations with other methods.

SMART LAPLACIAN SMOOTHING

One of the variants of the standard Laplacian smoothing algorithm is smart Laplacian smoothing (Freitag et al., 1997) also called constrained Laplacian smoothing.

The main advantage of this algorithm is that points can only be re-cloned if there is a real improvement in the neighbouring contours. In other words, a vertex is repositioned when the new candidate position improves the quality of the star elements. Otherwise, the node remains in its current position. Note that the candidate position is the centroid of the surrounding points, as in the previous algorithm.

This method is quite simple from a computational capacity point of view and the quality of the final metric can be chosen according to the application.

VORONOI TESSELLATION

A Voronoi tessellation is called a centroidal Voronoi tessellation (CVT) when its generating points are the centroids of the corresponding Voronoi regions. This special structure has applications in many fields, such as image processing, clustering, cell division and others (Du et al. 1999).

A procedure that reduces calculation time is to apply Lloyd iteration, iteratively calculating Voronoi regions for each point and updating their positions to the centroid of each region until convergence. In general, the new position can be defined by the following relationship:

$$x^* = \frac{\int_{V_i} x\rho(x)dx}{\int_{V_i} xdx} \tag{27}$$

where:

V_i Voronoi region generated by the vertex x_i;

ρ density function.

Note that Lloyd's iteration is not the only technique for determining CVTs.

However, this method is naturally suitable for smoothing meshes, even though the Lloyd iteration converges slowly and is difficult to analyse. Therefore, there are several smoothing techniques in the literature based on Voronoi's concept of centroidal tessellation, differing mainly in density function and regions leading to less computation time. For this purpose, variants have been proposed such as the Weighetd Centroid of Circumcenters (WCC) where the vertices are placed in a weighted average of the

corresponding circumcentres and the weights are based on the physical size of each simplism (Alliez et al., 2005). This is equivalent to calculating a weighted centroid of the corresponding Voronoi region as expressed by the following relation:

$$x^* = \frac{1}{|\Omega_i|} \sum_{\tau_j \in \Omega_i} |\tau_j| \, c_j \tag{28}$$

where:

Ω_i star of the point of interest;

c_j circumcenter of the simplex τ_j;

$|.|$ area in two-dimensions.

The circumcenter can be defined as the intersection point of the bisectors of a triangle, which is also the centre of the circle that passes through all three vertices of the triangle. Using the same density function, the centroid of each simplex can be used instead of the circumcentre (Chen et al., 2004). This algorithm allows the surrounding points to be calculated more quickly:

$$x^* = \frac{1}{|\Omega_i|} \sum_{\tau_j \in \Omega_i} |\tau_j| \, x_{\tau_j} \tag{29}$$

where:

x_{τ_j} centroid of the simplex τ_j.

Furthermore, for non-uniform domains, considering the density function of the mesh in the calculations, the smoothing algorithm can generate an appropriate gradation for the modulation regions. Thus, considering the density function ρ, the previous equation becomes:

$$x^* = \frac{\sum_{\tau_j \in \Omega_i} |\tau_j| \, x_{\tau_j} \rho_{\tau_j}}{\sum_{\tau_j \in \Omega_i} |\tau_j| \, \rho_{\tau_j}} \tag{30}$$

OPTIMAL DELAUNAY TRIANGULATION (ODT)

ODT is a new adaptation of mesh smoothing based on the concept of Delaunay triangulation and aims to distribute the edge lengths equally according to the function to be approximated (showing that when $f(x) = x^2$, Delaunay triangulation can be defined as optimal in terms of minimising interpolation error). In essence, the energy function of the interpolation error is exactly solvable, resulting in the new position of each query point. For the generation of a uniform mesh of regular shape, the new position can be calculated as:

$$x^* = -\frac{1}{2|\Omega_i|}\sum_{\tau_j \in \Omega_i}(\nabla|\tau_j(x)|\sum_{x_k \in \tau_j, x_k \neq x_i}\|x\|^2 \tag{31}$$

WELL-CENTRED TRIANGULATION (WCT) SMOOTHING

Another iterative smoothing method is that which takes into account well-centred triangulations (WCT), i.e. sharp triangulations in the plane. Considering a mesh M, this algorithm tends to maximise the maximum angle of the mesh, while penalising smaller angles, as expressed by the following relationship:

$$\psi_{WCT} = \sum_{\theta \in M}|\cos(\theta) - 1/2|^p \tag{32}$$

where:

p finite number with $p \in \{4, 6, 8\}$);

θ each corner in triangulation M.

This algorithm requires very high computational times but is capable of preserving topological connections, unlike the methods mentioned above, which incorporate edge flip operations into the smoothing operations.

4.3. Polygonal meshes

Polygon meshes are representations that play an important role in geometric modelling applications, computer graphics, engineering, simulation, architecture, etc.

These representations are based on the idea of cell decomposition: a complex object is represented by an assembly of (possibly many) simple polygonal cells. Triangles and quadrilaterals are the most commonly used cells for surfaces (see Figure 4.7).

Quad meshes, i.e. meshes composed entirely of quadrilaterals, have been widely used in CAD and simulation for many years, as some applications are better suited to meshes composed of quadrilaterals than triangular meshes.

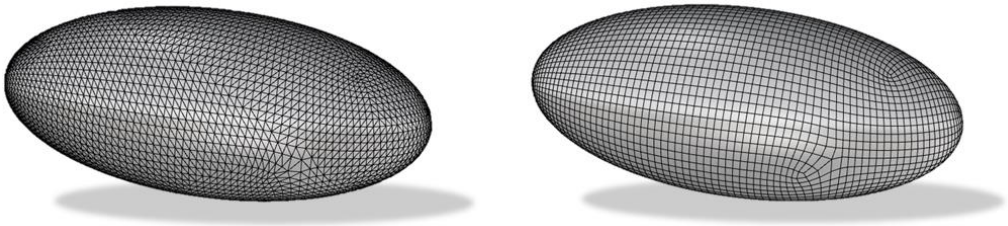


Fig. 4.7 - Illustration of triangular and quadrangular mesh.

While triangle meshes are much more common in computer graphics, and most research in the field of geomatic processing has focused on them, the advantages of using squares for graphics applications are well known and have numerous benefits.

As in the general case described above, the components of a two-dimensional polygonal mesh are vertices, edges and facets (Figure 4.8). Vertices are points in space, edges are line segments bounded by pairs of vertices, and facets are polygons bounded by cycles of edges that may share vertices and edges (Bommes et al., 2013).

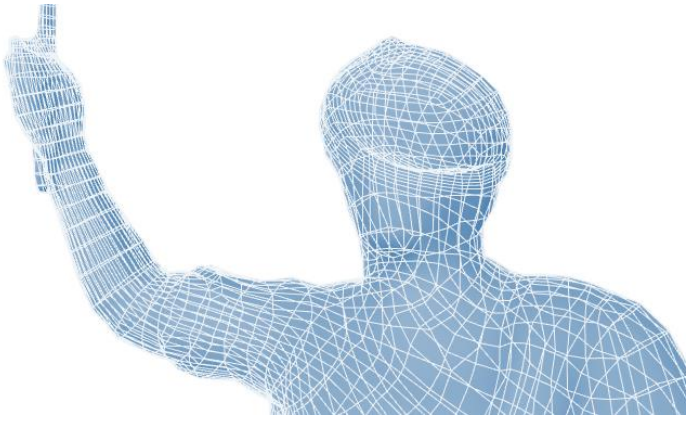


Fig. 4.8 - Example of modelling in quad-mesh.

A further element characterising quadrangular meshes are cross-fields, consisting of a pair of orthogonal vectors assigned to each mesh point.

Each cross-field consists of two components: the angular and the dimensional. These are stored in two separate structures, called orientation field and sizing field respectively. From the former, information about the orientation of the quadrangles is obtained, and from the latter, the dimensions of the quadrangles.

4.3.1. Classification and characterisation of quad meshes

Square meshes can be classified into different classes, based on degree of regularity.

a. A regular mesh, or a geometric image, can be globally mapped onto a rectangular subset of a square tiling. Regular meshes have a limited field of application, as they are suitable for disc or torus surfaces (a toroidal topology mesh can be obtained by identifying opposite sides of a regular mesh without introducing irregular vertices) (Gu et al., 2022).

b. A quad mesh is semi-regular if it is obtained by gluing several 2D regular quad arrays conformally from one side to the other. Each of these regular sub-matrices is called a patch, and the number of patches is assumed to be much smaller than the total number of facets. In a semi-regular square mesh, all the vertices that are inside the patches or lie along their edges are regular, while only the vertices that lie at the corners of the patches can be extraordinary. Regular semi-regular meshes represent the most important class in terms of applications.

c. A quadruple mesh is semi-regular if most of its vertices have valence 4. All semi-regular meshes are valence semi-regular, but not all valence semi-regular meshes can be partitioned into a small number of patches. The differentiation of these two classes of quad meshes makes it possible to more precisely differentiate algorithms that aim to produce meshes with a patch structure from those that minimise the number of irregular vertices.

d. A quad mesh is unstructured if a large fraction of its vertices are irregular. An unstructured mesh is obtained, for example, by subdividing each facet of an arbitrary triangular mesh into three quads (Bommes et al., 2013).

Although a mesh composed of triangles brings considerable computational advantages, there are several reasons why quadrangles may be more performant. These reasons are to be found in the intrinsic topological properties of quadrangle meshes. Some of these properties are related to the ability to more accurately approximate desired shapes, while others relate to the suitability to perform subsequent processes or analyses.

Quadrangles are also more suitable for depicting the shapes of objects as they can more naturally approximate curvature. Generally, curvature zones are locally characterised by two main directions, indicating the maximum and minimum degree of curvature present; this information is encoded by means of curvature tensors, and quantifies how far the geometry of the object deviates from the completely flat shape. The quadrangle, due to its geometric nature, is able to align itself to both directions.

Again, with reference to the quadrangle, the dimensions of its sides are inversely proportional to the degrees of curvature: along the direction of maximum curvature, the sides will be shorter, and vice versa along the direction of minimum curvature, the sides will be longer. This topological characteristic is called anisotropic quadrangulation, whereby the characteristics of the quadrangles depend on the directions of curvature considered (Kovacs et al., 2010). Otherwise, the quadrangulation is isotropic. By virtue of these properties, quadrangle meshes are of higher quality and much more versatile and perform better in various fields.

In some cases, it is indispensable to use both triangular and quadrangular meshes: in this case we speak of 'quad-dominant meshes'. Conversely, meshes with more than four sides known as 'N-gons' are hardly ever used as they cause somewhat unnatural deformations and shading problems during subdivision operations.

A polygonal mesh can easily contain hundreds of thousands of elements. Therefore, it is important to understand how to represent this information in data structures that allow for less processing time and less occupied memory space. In general,

the more relationships between vertices, sides and faces are exploited, the less processing time is required, but the more space is occupied. The most common representations are listed below.

- **Vertex-vertex mesh**: this is the crudest representation method, as it consists of representing the model by means of a simple list of vertices. Each vertex is identified by its own co-ordinates, and the relationships between them to identify faces and sides are ignored.
- **Face-vertex mesh**: the existing relationships between faces and vertices are exploited. There is a list of vertices, identified by their coordinates, and a list of faces pointing to them. Each face is then defined by a list of four pointers to the list of vertices.
- **Winged-edge mesh**: a list of vertices, a list of sides and a list of faces are created. Each face of the face list is represented by pointers to the side list, and each side of the side list by pointers to the vertex list. This is the most explicit representation possible.

Due to the discrete nature of polygon meshes, any three-dimensional object is represented with a certain degree of approximation. Objects with flat surfaces are reproduced correctly; for curved or complex objects, in order to achieve greater fidelity and accuracy in reconstruction, it is necessary to increase the number of faces and vertices, thereby increasing the computational capacity of computers.

4.3.2. Tri-to-Quad mesh conversion

Many datasets, such as those from photogrammetric processing or TLS surveys, are originally provided as triangular meshes.

In order to convert meshes from triangular to polygonal, it is necessary to apply a series of simplification algorithms capable of reconstructing meshes composed of quadrilateral elements. Based on the edge collapse method for triangular meshes, Daniels et al., 2008, adopt an approach capable of solving the problem of maintaining quadrilateral connectivity during the creation of the level of detail. This method consists of a series of operations applied to the mesh that can maintain the topology of the mesh

itself and can be used to re-mesh generic polygons, i.e. triangle-dominated meshes, into quad-only meshes. Applying this technique, the quadruple mesh obtained has the advantage of preserving all the original edges, but also has several disadvantages: in terms of complexity, it has three times as many quadruples as the triangles of the initial mesh; in terms of quality, no more than 50% of its vertices are regular.

It is also possible to construct polygonal meshes consisting initially of both tri and quad hybrid meshes that are subsequently converted to purely quad meshes. Each face of the tri-quad mesh is divided into triangles by barycentric subdivision; subsequently, pairs of such triangles are joined to form quads by deleting all edges that existed before the subdivision (Velho et al., 2001). This method is better than the Catmull-Clark operation on the input triangle mesh, both in terms of quality and complexity, but still produces a non-negligible increase in the total number of quads.

Tarini et al., 2010 proposes a methodological approach that, starting with a quadruple mesh surface, produces a mesh with a number of quadruples equal to half the triangles of the starting mesh. This method requires as input a (connected) mesh with an even number of triangles. If the mesh is closed (and two-manifold), this condition is guaranteed; otherwise, each edge can be divided in half, increasing the number of triangles by one. This approach involves a first step where most triangles are merged in quad pairs, dissolving their shared edge. The edges of the triangle mesh can be flagged as dissolved with any heuristic, with the constraint that no triangle can have more than one flagged edge. The objectives are to maximise the number of reported edges and to prioritise the creation of quadrilaterals with (almost) right angles. Subsequently, the remaining triangles are made to 'crawl' over the mesh towards each other until they can be joined into quads.

In terms of the complexity of the resulting square mesh, this method is better than the direct use of Catmull-Clark subdivision as more regular vertices are produced, preserving all original edges (Figure 4.9).

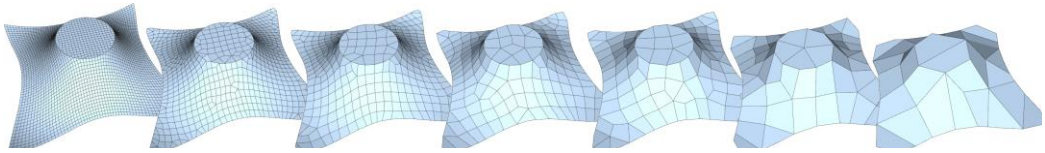


Fig. 4.9 - Simplification process of a quad mesh *(image taken from Tarini, M., Pietroni, N., Cignoni, P., Panozzo, D., & Puppo, E. (2010, May). Practical quad mesh simplification. In Computer Graphics Forum (Vol. 29, No. 2, pp. 407-418). Oxford, UK: Blackwell Publishing Ltd.).*

4.4. NURBS modelling

In cases where there is a need to represent the standard geometry of an object as accurately as possible, NURBS modelling is the best option to use. With this technique, high levels of accuracy and surface quality can be achieved despite the difficulties encountered in computer-aided modelling (CAM) processes.

NURBS (Non-Uniform Rational B-Splines) geometries are mathematical representations of 3D geometry, which precisely define any shape: from a simple line to a circle, arc or curve, to the most complex 3D free-form or organic solid or surface. Due to their precision and flexibility, NURBS models can be used in a variety of processes, from illustrations and animations to manufacturing.

NURBS geometries have five important characteristics that make them an ideal choice for computer-aided modelling.

1. There are several industry-standard solutions that can exchange NURBS geometries. Consequently, it is possible to use their important geometric models in various engineering modelling, animation and analysis programmes.
2. NURBS have a precise and well-known definition, based on mathematical and computer rules.
3. NURBS can accurately represent both standard geometric objects (such as lines, circles, ellipses, spheres or toroids) and free-form geometries, such as objects and structures.

4. The amount of information required for the NURBS representation of a geometric element is far less than the amount of information required to represent the same geometry using mesh approximations.
5. The estimation rule of a NURBS can be implemented on a computer efficiently and accurately.

4.4.1. NURBS definitions and geometries

NURBS curves and surfaces behave similarly, so the technical terminology relating to them is almost the same. A NURBS curve is defined by four characteristics: the degree, the control vertices, the nodes and the estimation rule.

There are two types of NURBS curves:

- **EP curves.**
- **CV curves.**

EP curves pass through points entered by the user which are called end points (EP - End Points), while CV curves only approach points entered which are called control vertices (CV - control vertices), or as they are called in some software, control points.

In the following, the characteristics of a NURBS geometry will be analysed.

Degree: NURBS geometries are defined by their degree, which is a positive integer number that is usually worth 1, 2, 3 or 5, although it can be any integer. As a rule, NURBS lines and polylines have degree 1, NURBS circles have degree 2 and most free-form curves have degree 3 or 5. Instead of indicating the degree of the curve with the respective number, the terms linear, quadratic, cubic and quintic may be used respectively. Sometimes, reference can be made to the order of a NURBS curve. The order of a NURBS curve is a positive integer equal to (degree+1). Accordingly, degree corresponds to (order-1). It is possible to increase the degree of a NURBS curve without changing its shape. On the other hand, it is generally not possible to reduce the degree of a NURBS curve without changing its shape.

Control points: are a row of points at least equal in number to (degree+1). One of the simplest ways to change the shape of a NURBS curve is to vary the position of

its control points. Each control point is associated with a weight (i.e., its ability to attract the curve). With a few exceptions, the weights are positive numbers. When the control points of a curve all have the same weight (usually 1), the curve is called 'non-rational'. Otherwise, it is called rational. The letter R in the NURBS acronym stands for 'rational' and indicates that a NURBS curve can be rational. In practice, most NURBS curves are non-rational. Some NURBS curves (circles and ellipses are a clear example) are always rational.

Nodes: are a row of numbers equal to (degree+N-1), where N represents the number of control points. This sequence of numbers that specifies the parametric definition of the curve is also called the 'vector of nodes'. The sequence of node numbers must fulfil several technical conditions. As a rule, to ensure that these technical conditions are met, one requires that the numbers are the same or greater as one proceeds towards the end of the list and limits the number of repeated values to a number no greater than the degree. The number of times the value of a node repeats itself is called 'node multiplicity'. A node has full multiplicity if its value repeats as many times as the degree. If a sequence of nodes begins with a full multiplicity node, follows with simple nodes, ends with a full multiplicity node and all values are equally spaced, its nodes are said to be uniform. The letters N and U in the NURBS acronym stand for 'non-uniform' and indicate that the nodes of a NURBS curve may be non-uniform.

Repeated node values within a sequence of nodes make the NURBS curve less smooth. A full multiplicity node in the middle of a sequence of nodes indicates that there is an area in the NURBS curve that may be associated with a discontinuity. For this reason, some designers prefer to add and remove nodes and then adjust the control points to obtain curves with smoother or more angular shapes. Since the number of nodes is equal to (N+degree-1), where N is the number of control points, adding nodes implies adding control points, just as removing them implies removing control points. It is possible to insert nodes into a NURBS curve without changing its shape. In general, removing nodes will change the shape of the curve.

In general, the number of end points $nEP$of control vertices nCV and the degree of the curve are linked by (equivalent) formulae:

$$nCV = nEP + degree - 1$$
$$nEP = nCV - degree + 1 \quad (33)$$

Estimation rule: is a mathematical formula that assigns a number to a point. The NURBS estimation rule is a formula involving degree, control points and nodes. It calculates the so-called B-spline basis functions. The letters B and S in the NURBS acronym stand for *basis spline*. The number with which the estimation rule is started is called the parameter. We can think of the estimation rule as a "black box" that receives a parameter and generates the position of a point. The degree, nodes and control points determine how this "black box" works.

Extremely flexible and suitable for all modelling processes, NURBS follow the model theory of Bezier Curves generated through the following mathematical formulation (Ji et al., 2021):

$$C(u) = \sum_{i=1}^{n} \frac{N_{i,n}(u) \cdot w_i}{\sum_{j=1}^{k} N_{j,n} \cdot w_j} P_i \quad (34)$$

where:

w_i weights;

P_i control points;

$N_{i,n}$ normalised B-Spline basis function of n degree.

$$N_{i,0}(u) = \begin{cases} 1 & if\ u_i \le u \le u_{i+1} \\ 0 & otherwise \end{cases} \quad (35)$$

$$N_{i,n}(u) = \frac{u - u_i}{u_{i+n} - u_i} N_{i,n-1}(u) + \frac{u_{i+n+1} - u}{u_{i+n+1} - u_{i+1}} N_{i+1,n-1}(u) \quad (36)$$

where u_i are the knots forming the knot vector:

$$U = \{u_0;\ u_1; \ldots;\ u_{n+n+1}\} \quad (37)$$

In addition to describing curves (Figure 4.10a), NURBS can also be extended to the two-dimensional case to describe surfaces (Nguyen et al., 2013).

In this case we speak of control vertices, surface patches, grade and parameters u, v.

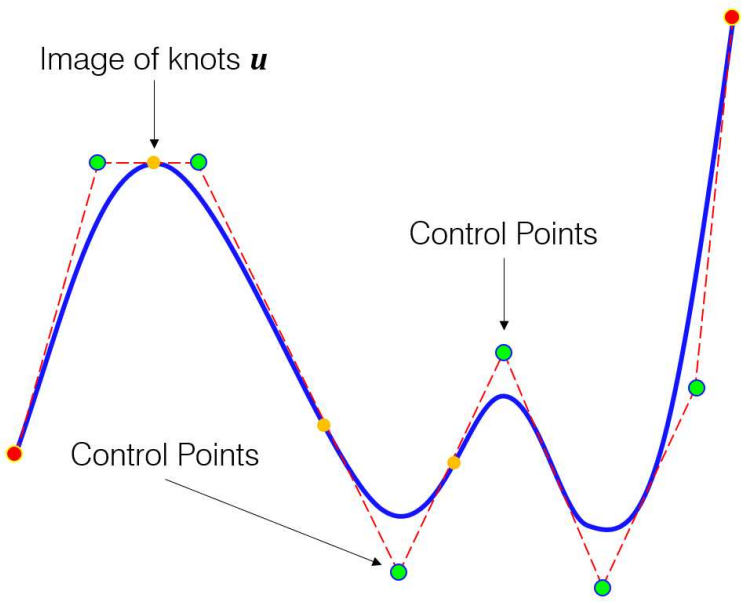
Since these are two-dimensional surfaces, the coordinates of the points belonging to them are determined by two independent parameters called u, v.

$$S(u,v) = \sum_{i=1}^{k} \sum_{j=1}^{l} R_{i,j}(u,v) \cdot P_{i,j} \tag{38}$$

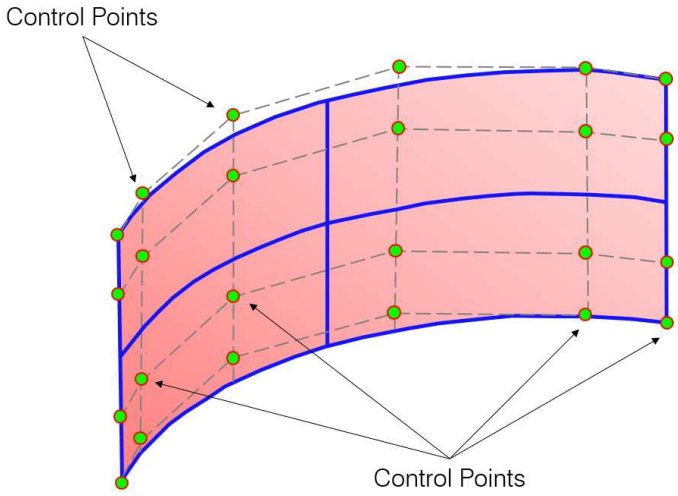
where the rational basis function holds:

$$R_{i,j}(u,v) = \frac{N_{i,n}(u) N_{j,m}(v) w_{i,j}}{\sum_{p=1}^{k} \sum_{q=1}^{l} N_{p,n}(u) N_{q,m}(v) w_{p,q}} \tag{39}$$

Surface patches are equivalent to EPs for a NURBS surface (Figure 4.10b).



(a)



(b)

Fig. 4.10 - NURBS geometries: example of NURBS curve (a) and NURBS surface (b).

The number of surface patches nSP is related to the number of CVs in a similar way to EPs:

$$nSP_u \cdot nSP_v = (nCV_u - degree) \cdot (nCV_v - degree) \quad (40)$$

All NURBS surfaces are derived from a spatial deformation of a grid of square surfaces (surface patches) placed side by side and, the images of the nodes (i = 1, . . . , n) divide the curve into segments that play the role of finite elements in an analysis context.

4.4.2. Differences between polygonal and NURBS models

There are some substantial differences between polygonal and NURBS modelling. Among the first noticeable differences, the modelling workflow is certainly the main aspect. In fact, following a polygonal modelling approach, making surfaces is easy as it is N-gon modelling used to manipulate and change the mesh. In contrast, in the case of NURBS, objects are always 4-sided, which poses certain limitations in the modelling workflow. In addition, NURBS objects are always separate and difficult to attach; this results in an additional assembly step between the different components.

Another difference concerns the file size and the import into appropriate modelling software. In some cases, polygonal mesh models can be distorted for various reasons; this cannot happen with NURBS modelling, as each object is represented by an easily readable mathematical model. Another aspect that should not be underestimated is that NURBS files are smaller in size and easier to store.

In the process of texturing mesh models, unfortunately NURBS modelling has a shortcoming in that it is not possible to assign a photorealistic content to the model.

In terms of modelling aspects and final resolution and accuracy, polygon modelling uses flat surfaces or polygons to create an object. Consequently, all these polygons, in addition to vertices and lines between points, are included in the calculation process. This process obviously cannot define or create a perfectly smooth curve, which is possible with NURBS modelling; where each surface or curve is represented by a well-defined mathematical function.

Although such modelling allows greater precision than polygonal modelling, NURBS calculations are more difficult to process and for this reason, their use is not yet widespread, except in a few contexts.

The Table 4.3 provides some advantages of NURBS modelling over polygonal modelling in different application areas.

Tab. 4.3 - Advantages of NURBS modelling.

Modelling	Scope	Background
NURBS surfaces are easy to construct.	NURBS surface types are applied in various fields such as vector graphics.	NURBS assists in creating curves and different types of organic 3D shapes.
Offers smoother opening, closing and tightening curves.	It is possible to import NURBS data into different modelling, rendering or engineering analysis software.	Less information is needed to represent NURBS geometry, unlike faceted approximations.

4.5. File formats for storing polygonal meshes

There are several file formats that store mesh information. Generally, these interchange formats are designed and developed by the different processing software. Currently, there are several interchange formats in use, more or less used, that are capable of storing different types of mesh information. Some of the most commonly used storage formats will be described below.

OBJ

It is a file format for defining 3D geometries developed by Wavefront Technologies for its Advanced Visualizer software. It is an open format that has been adopted by many 3D graphics applications for interchanging data with other programmes. It is a simple data-format representing only the 3D geometry, i.e. the position of each vertex, the position of each UV co-ordinate for the textures, normals and faces that make up the model. The vertices are by default stored in a counterclockwise order, making the explicit declaration of normals unnecessary. The coordinates of an OBJ have no units of measurement, but information on the scale of the model can be contained in an annotated line of code.

FBX

It is a proprietary file format (.fbx) developed by Kaydara and owned by Autodesk since 2006. It is used to provide interoperability between digital content creation applications. The FBX file format is a proprietary format, however, the description and export of the format is often directed towards the FBX SDK extension, which provides header files for FBX readers and writers.

COLLADA *(acronym for COLLAborative Design Activity)*

It is a 3D application interchange file format distributed free of charge together with the source code by the non-profit organisation Khronos Group Inc. The file format is realised in XML code. Among the main features of this format are the import of mesh geometries, transformation hierarchies. It also stores materials, textures and shaders. In the case of virtual and animated objects, it can also store virtual camera and lighting information, animations and physical simulations.

3DS

It is a 3D image format used by Autodesk 3D Studio. It contains mesh data, material attributes, bitmap references, smoothing group data, viewport configurations, camera positions and lighting information. 3DS files can also include object animation data. 3DS files consist of blocks of data called 'chuncks' that contain a description of ID and length. The chunks store shapes, lighting and information that together represent the three-dimensional scene. Currently, this format has been replaced by .MAX files in more recent versions of 3ds Max software. However, the 3DS format is still widely used.

PLY

It is a computer file format known as Polygon File Format or Stanford Triangle Format. It was primarily designed to store three-dimensional data from 3D scanners. The data storage format supports a relatively simple description of a single object as a list of nominally flat polygons. A range of properties can be stored, including colour and transparency, surface norms, texture coordinates and confidence values of the data. The format allows for different properties for the front and back of a polygon.

There are two versions of the file format, one in ASCII format and the other in binary format. A PLY file begins with the 'header' attribute, which specifies the elements of a mesh and their types, followed by the list of elements themselves. The elements are usually vertices and faces, but may include other entities such as edges, distance map samples and triangle strips.

MA (MB)

It is a 3D project file created with the Autodesk Maya application. It contains an extensive list of textual commands for specifying file information. This type of file can be opened and edited in any text editor to resolve any problems with commands in the event that a file becomes corrupted. These files contain information to define 3D scene information such as geometry, lighting, animation and rendering.

MA files are generally saved in ASCII text format in contrast to MB files, which are saved in binary file format.The header of the MA file begins with a comment section that provides information on the creation of the file and the date of modification. Maya

file readers ignore this block as it is only used for information purposes. However, a header must begin with the first six characters as '//Maya'.

The Table 4.4 lists some of the other formats available for storing and managing mesh models.

Tab. 4.4 - Some mesh model interchange formats.

FILE	Description
3DX	Rinoceros file format
BLEND	Blender file format
X	Direct X object
BVH	Behavioral file (mesh animate)
MD3	Quake 3 vertex animations
FIG	Used by REND386/AVRIL
FLT	MulitGen Inc.'s OpenFlight format
RWX	RenderWave Object
POV	Persistence of vision
QD3D	Apple's QuickDraw 3D Metafile format
TDDD	For Imagine & Turbo Silver ray
C4D	Cinema 4D file format
NFF & ENFF	Neutral File Format (Extended)
VIZ	Used by Division's dVS/dVISE
HDF	Hierarchical Data Format
Z3D	ZModeler File format

5. SCAN TO H-BIM PROCESSES: FROM POINT CLOUD TO PARAMETRIC 3D MODEL

The management of complex point clouds, especially in the case of cultural heritage, requires increasingly efficient modelling tools, methods and techniques, capable of being shared with all operators at all times and suitable for different analysis purposes.

In recent years, the BIM (Building Information Modeling) approach is spreading rapidly (Chen et al., 2019) and has become an indispensable process in the design of new structures and infrastructures in the field of engineering. The application of BIM processes to historical structures or those belonging to the CH field poses an interesting challenge both in the field of 3D modelling and in the management/valuation of buildings of special historical and cultural interest. With this approach it is therefore possible to represent existing structures with high detail and accuracy and to assign a range of geometric, design and semantic information to each element (Diara et al., 2018).

The application of BIM processes to historic structures was introduced in 2012 under the name H-BIM, or Heritage-BIM (Dore et al. 2012). In this environment, architectural elements are represented not only in their geometry, but also in the corresponding attributes that form an information database on a historical and architectural level; geometries, materials, state of deterioration, historical phases and conditions, as well as all design interventions on a building can be managed in a high-performance manner on a three-dimensional model.

Since 2015 and especially in recent years, the research and development of solutions in the HBIM environment has grown. Different methodological approaches have been developed and a large number of information models of historic buildings have been developed by the scientific community. Some applications have been tested, for example, on the historic building of the Four Courts in Dublin (Dore et al. 2015), the Scala Regia in the Vatican (Paris et al. 2016), the Pavilion of Charles V, a building with

a Renaissance character located in the outdoor areas of the Alcazar in Seville, Spain (Nieto et al.2016), and the Wang Temple in Karpacz, Poland (Sztwiertnia et al. 2019). This procedure, in the context of the enhancement and conservation of cultural heritage, can also be integrated for the construction of 3D information models linked to a specific context, in which the geometry of the objects can be linked to information about the state of knowledge of the historical asset and thus represent a technical digital and organisational support for future conservation operations (Coli et al. 2019).

In general, the procedure for the construction of an H-BIM for a structure of particular historical-architectural interest may consist of several steps, depending on the complexity and objectives to be achieved. The main steps for a correct definition of an H-BIM model are:

- Identification of information requirements;
- Determining the required information and quality of scan data;
- Acquisition of scan data.
- BIM (or H-BIM) reconstruction as-is.

The first step is to identify the LODs, or Level of Detail. LODs have the task of precisely defining the level of development of the various types of information contained in the model. In particular, the various LODs can be schematised as follows (Brumana et al. 2018):

- LOD100 represents an a-dimensional conceptual model.
- LOD 200 model elements are represented with generic quantity, size, shape, position and orientation; non-graphical information may also be attached to the element.
- LOD 300 a three-dimensional model in the executive design phase.
- LOD350-400 represents the model implemented for the construction phase.
- LOD500 the as-built upgrade after the construction phase.

Therefore, taking this classification into account, the HBIM project is generally oriented towards the LOD500 level.

With regard to the second topic, i.e. determining the quality of the required scan data, special attention must be paid to various aspects of the geomatic survey, such as spatial resolution, density and consistency of the point cloud. This also means identifying a suitable sensor that can meet the characteristics of the project. To obtain a dense point cloud, for example, the survey can be carried out with the use of an active sensor, such as a terrestrial laser scanner (TLS) or a passive sensor (cfr. 1).

The third phase involves the scanning of the structure under examination and the subsequent processing of the acquired datasets using appropriate photogrammetric processing software.

The last phase, also known as "Scan-to-BIM", consists of transforming the point cloud into parametric objects. This means that objects, such as walls, doors or windows of a structure, can be modified with a simple setting of its dimensions in a project database; the geometry, material properties and technical elements of the objects can be edited, queried and linked with other information in a simple, precise and intuitive way.

When the objects to be modelled are numerous, this operation takes on considerable importance. Computer-aided management, i.e. the description from a geometric-architectural and engineering point of view, is a modern and indispensable tool in the field of cultural property surveying, capable of supporting such modelling activities. Problems arise as the complexity of such objects increases and their geometry plays a fundamental role in the subsequent representation and analysis of the data. For this reason, methods and algorithms capable of automating 3D reconstruction processes are continuously being researched by the scientific community.

The aim is to make the 3D models highly performant from a computational and storage point of view, but without diminishing the degree of detail and accuracy levels of the architectural asset.

In order to achieve these objectives, simplification algorithms are taken into account in order to manage the large number of meshes obtained from the photogrammetric process more easily. Furthermore, another fundamental aspect not to be

underestimated concerns interoperability with the BIM software and platforms available today for the management of such 3D models.

In this chapter, a methodological approach will be shown that is able to transform the point cloud obtained from geomatic surveys into parameterised polygonal three-dimensional models. For this purpose, the three-dimensional modelling software Rhinoceros was used, which, thanks to specific tools and plugins, not only allowed the parameterisation of objects, but also enabled a bidirectional transformation of objects between the modelling software and the BIM software (Revit). As a result, any changes made to the model in Rhinoceros were automatically made to the objects in Revit and vice versa.

The potential, simplicity and efficiency of the developed method show the high degree of interoperability in parametric 3D modelling in scan-to-BIM processes. This methodological approach was applied to a religious building in Italy, belonging to the Cultural Heritage and dating back to the 14th century.

5.1. Integrated survey of historical architecture

5.1.1. Brief history of San Nicola in Montedoro

The church of San Nicola a Montedoro is one of the oldest in the city of Martina Franca, in the province of Taranto (Italy) and presumably dates back to the 14th century, the period of the city's Angevin foundation. The church is located in the Montedoro district, from which it takes its name "San Nicola di Montedoro" (Figure 5.1).

The church stands on a site that in the 14th century was occupied by the church of San Paolo dei Greci, whose name also indicated the district. This small church fell into ruin in the 16th century and was rebuilt in the 17th century. After the reconstruction, the church was named after St Nicholas of the Poor, because in the past, together with another building called "Ospedaletto", it was mainly dedicated to social welfare activities and was later named after St Nicholas in Montedoro.

The church retains its original structure, despite internal transformations in the 17th century. In fact, the church has undergone several changes over the centuries, the most significant of which occurred in the 17th century.

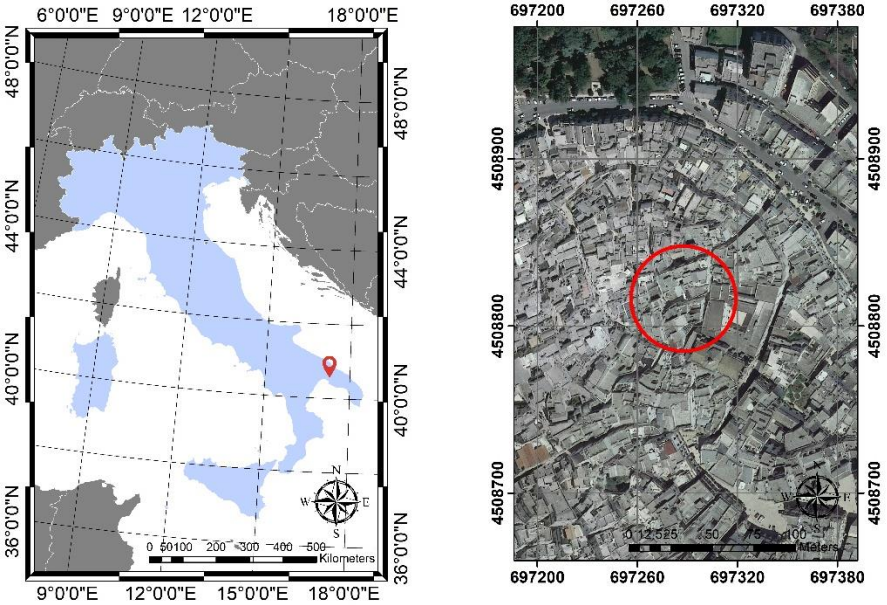


Fig. 5.1 - Keymap.

The structure is characterised by a modest rectangular hall and late-medieval architectural elements visible mainly on the exterior. The building, oriented roughly north-east/south-west, consists of a single hall measuring approximately 5.95 x 5.20 metres. The simplicity of the exterior façade is embellished only by the roof with raised pitches that intersect and form two gables with cladding made of typical limestone slabs (Figure 5.2a). The slightly ogival portal is surmounted by a lunette and a small radiating rose window, while a graceful bell tower rises above the tympanum of the main façade. The interior of the church, which has been remodelled several times, houses stone altars made in the 17th century and a vast repertoire of paintings made between the 16th and 19th centuries. Many frescoes have been repainted with obvious overlaps; in other cases, the paint film has been completely chiselled away, leaving

only a few traces of the sinopia on display. The central fresco depicts Our Lady of Grace with St. Nicholas, St. Benedict, St. Scholastica and a martyr saint, probably St. Comasia (Figure 5.2b). On the right wall, St. Anthony Abbot, La Pietas and Our Lady of the Rosary are depicted with clipei showing the most significant episodes in the life of Christ. On the wall where the entrance is located are images of St. Anthony of Padua and St. Francis of Paola, which flank the portal. The pictorial decoration on the left wall is rather deteriorated; an effigy of Saint Margaret and a depiction of the Massacre of the Innocents stand out.

(a) (b)

Fig. 5.2 - The church of “San Nicola in Montedoro”: entrance and main façade (a); central painting (b).

5.1.2. Integrated survey

The survey of the church was carried out through the use and integration of active and passive, ground and airborne sensors. In particular, the exterior façade was surveyed with a TLS, the interior part with a Digital Single-Lens Reflex (DLSR) camera with a fish-eye lens, and the upper part of the building (i.e. the roof and other architectural elements not visible with a terrestrial survey) with the use of a camera mounted on a UAV platform. About the generation of the model for the exterior of the church, the survey was carried out with a terrestrial laser scan. In this case study, the TLS FARO

FocusS 350 was used. HDR imaging and HD photographic resolution (overlay up to 165 megapixels in colour) guarantee detail-accurate scanning results with high data quality (3D position accuracy 10m: 2mm / 25m: 3.5mm). To cover the entire exterior surface of the church, three acquisition stations were constructed. The post-processing of the TLS scans was performed with Autodesk Recap software, allowing fully automatic recording of the scans. In the case of a partially successful procedure, the software allows manual identification of homologous targets and natural points, to reduce the distance between contiguous scans and improve their alignment with the iterative nearest point (ICP) algorithm. The photogrammetric survey from the UAV was carried out using a Parrot Anafi, a UAS (Unmanned Aerial System) quadcopter equipped with a Sony Sensor® 1/2.4" 21MP (5344 × 4016) CMOS (complementary metal-oxide semiconductor), which allows clear and detailed images to be obtained thanks to a 3-axis stabiliser. The distance between the UAV and the building was very close due to the presence of numerous obstacles in the historical centre where the church is located. Consequently, the images were acquired with a high geometric resolution. In any case, the photogrammetric survey was carried out with a high degree of overlap between the images. Furthermore, by varying the angle of inclination of the camera, it was possible to acquire images of every part of the building. In this way, it was possible to construct a network of 97 images with a high degree of overlap and a configuration of converging images. Considering 5 GCP, the root mean square error (RMSE) for spatial coordinates, evaluated on the cameras used in this dataset, was 0.009 m; notably, this RMSE refers to the georeferencing process of the images and not to the resolution of the model.

For the interior of the church, as there are also frescoes of great historical and cultural value and considering the rather confined environment, a photogrammetric survey was carried out using a Nikon D5000 reflex camera with a calibrated fisheye lens (10 mm focal length). In order to properly scale and reference the entire model, a series of GCPs were materialised and surveyed using a Leica TS30 Total Station. In this way, it was possible to obtain horizontal and vertical angular observations of the control points (GCPs) both inside and outside the building. Post-processing of the data was carried out in LGO (Leica Geo Office) developed by Leica Geosystem.

5.2. Building 3D model

To obtain a dense point cloud of a structure, a very common method is the photogrammetric method. In particular, the Structure from Motion (SfM) approach has become very popular in close-range photogrammetry (CRP) due to its ability to determine external orientation parameters without any a priori knowledge of the approximate camera positions and 3D points (Bitelli et al. 2018). The SfM technique requires, for the realisation of 3D models, a block of images with a high degree of overlap that captures the complete 3D structure of the scene as seen from different positions. Based on the estimated camera positions, depth information can be calculated for each image and combined into a single, dense point cloud (cfr. 2.2). In fact, based on the Multi-View-Stereo (MVS) algorithm, this task allows to increase the density of the point cloud generated in the SfM process (Visintini et al. 2019; Alessandri et al. 2019).

In this case, the small interior spaces of the building allowed the use of a fisheye lens. However, the geometry of the images taken with a fisheye does not conform to the central perspective projection. Therefore, the collinearity equation cannot be used to mathematically describe the imaging process. Indeed, fisheye projection is based on the principle that, in the ideal case, the distance between an image point and the main point is linear from the angle of incidence of the ray from the corresponding point. Therefore, the equations of the fisheye projection model are (Schwalbe 2005):

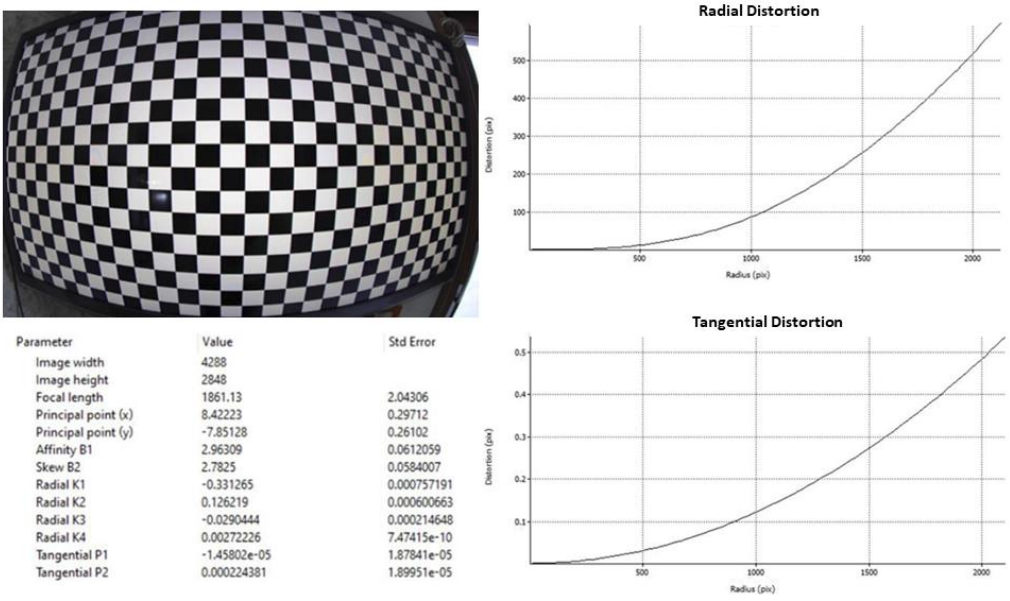
$$x' = \frac{\frac{2R}{\pi}\operatorname{atan}\left[\sqrt{\frac{(X_c)^2 + (Y_c)^2}{Z_c}}\right]}{\sqrt{\left(\frac{Y_c}{X_c}\right)^2 + 1}} + dx + x_h \tag{41}$$

$$y' = \frac{\frac{2R}{\pi}\operatorname{atan}\left[\sqrt{\frac{(X_c)^2 + (Y_c)^2}{Z_c}}\right]}{\sqrt{\left(\frac{Y_c}{X_c}\right)^2 + 1}} + dy + y_h \tag{42}$$

where:

x', y' image coordinates;
X_c, Y_c, Z_c object point coordinates in the camera coordinate
R image radius
$\mathrm{d}x, \mathrm{d}y$ distortion polynomials
x_h, y_h coordinates of the principle point.

The image modelling workflow to build an accurate 3D model involves camera calibration, dense matching and surface extraction. Therefore, the first photogrammetric step was the calibration of the camera. This made it possible to calculate the main parameters of the camera (focal length, main point coordinates, radial and tangential distortions). This process was carried out using Agisoft Lens, which uses the LCD screen as a calibration target. The chessboard was displayed on a 42″ LCD monitor and 12 photos were taken at different positions. The results of the processing, radial and tangential distortion curves are shown in Figure 5.3.

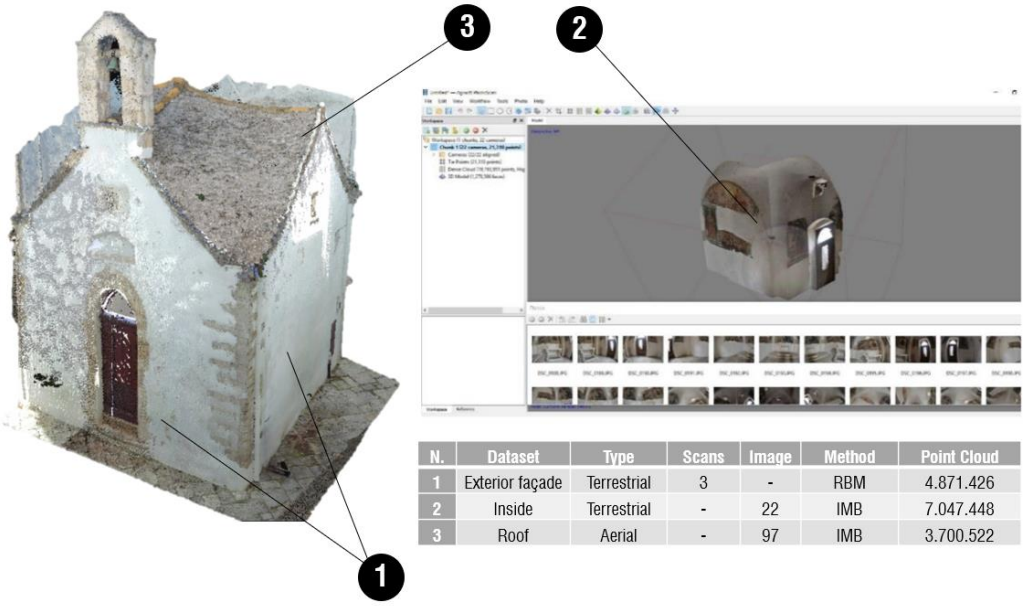


Parameter	Value	Std Error
Image width	4288	
Image height	2848	
Focal length	1861.13	2.04306
Principal point (x)	8.42223	0.29712
Principal point (y)	-7.85128	0.26102
Affinity B1	2.96309	0.0612059
Skew B2	2.7825	0.0584007
Radial K1	-0.331265	0.000757191
Radial K2	0.126219	0.000600663
Radial K3	-0.0290444	0.000214648
Radial K4	0.00272226	7.47415e-10
Tangential P1	-1.45802e-05	1.87841e-05
Tangential P2	0.000224381	1.89951e-05

Fig. 5.3 - Results of the camera calibration with fisheye generated in Agisoft Lens.

Next, the 3D model was constructed in the Agisoft Metashape software. After importing the images into the software and loading the camera calibration parameters, the software recorded the images. For the scaling and referencing phase of the photogrammetric model, a topographic survey was carried out. In particular, 8 points were surveyed of which 6 points were used as GCPs and 2 as CPs. The accuracy of the photogrammetric model assessed on the GCPs was 0.003 m, while on the CPs it was 0.004 m.

Once this phase was completed, the dense cloud relating to the building's interior and roof was constructed. In order to complete the entire point cloud model, the scans obtained from the TLS survey were merged with the point cloud obtained from the photogrammetric process. The different point clouds were then merged into a single point cloud on the basis of common points. Figure 4 below schematises the results obtained in terms of the point cloud.



N.	Dataset	Type	Scans	Image	Method	Point Cloud
1	Exterior façade	Terrestrial	3	-	RBM	4.871.426
2	Inside	Terrestrial	-	22	IMB	7.047.448
3	Roof	Aerial	-	97	IMB	3.700.522

Fig. 5.4 - Point clouds obtained in the several datasets.

After the reconstruction of the dense point cloud, it was possible to generate a polygonal mesh model based on the dense cloud data. In this way, the architecture

inside the church was constructed. On the basis of the three-dimensional surveys and after generating a referenced point cloud, it was also possible to start the production of 2D layouts, characterised by a higher level of detail than traditional restitution, and several very high-resolution orthophotos; the projection planes were chosen in relation to the planes at the positions of the frescoes. In order to achieve a high quality of the orthophotos and consequently highlight the church's precious frescoes, a pixel size of 0.001 m was used.

5.3. From SfM/TLS clouds to H-BIM models

5.3.1. Methodological approach

Once the point cloud has been acquired, it is necessary to transform the points of a structure into parametric objects. This can be done using 3D reconstruction through profiles; in fact, from a set of 3D point clouds, a parametric curve or surface can be described using the following two methods (De Luca 2011).

Approximation: the curve or surface is defined through points, but does not necessarily pass through them; the surface is therefore described by adjusting some control points.

Interpolation: the curve or surface passes through all entry points.

In the present case study, the NURBS (Non Uniform Rational B-Splines) was used for the parameterisation of curves or surfaces. A NURBS curve C(u), is a rational polynomial function value vector (cfr. 4.4.1).

Therefore, a set of NURBs makes it possible to describe the geometry of a generic (and complex) surface. In particular, a series of profiles are constructed from the point cloud acquired using geomatic techniques. These profiles are interpolated with NURBS. The three-dimensional modelling phase can be carried out in various commercial software, such as Rhinoceros and Blender. Rhinoceros (also called 'Rhino') version 7 was used in this work. The reason for this choice is due to the simplicity in the creation and modelling of objects and the implementation of certain tools and plugins in Rhinoceros. In this context, a very useful plugin for 3D modelling is Arena4D,

which allows point clouds to be imported into Rhino for modelling. All Rhino functions are controlled from the simple Arena 4D panel, presented in the style of Rhino; the functions included in this plugin are:

- overlapping of the point cloud;
- decimation of the point cloud;
- snap to one point one point;
- sections through a point cloud;
- sections between geometry;
- customisable shader.

In addition, this software, as of version 6.0, is able to interoperate with Grasshopper. Grasshopper, a programming language that, like Rhino, has become a solid development platform and provides the basis for numerous third-party components, from physical simulation to robotic control. Recent programming developments in Grasshopper have made it possible to transform objects modelled in Rhinoceros into Autodesk Revit.

Objects created in this way belong to both Rhinoceros and Revit. Therefore, the software are fully connected and, as a result, the user can make surface changes in real time in both Rhinoceros and Revit. In addition, surfaces can be managed through the Grasshopper programming language and, as a result, can be moved, rotated, etc.

The path between the programmes is two-way, so the surface imported into Revit, once modified, changes simultaneously in Rhinoceros. In the Grasshopper software, categories (architectural-structural elements) such as the following are chosen and assigned:

- walls;
- floors;
- pillars;
- generic masses.

Features assigned in Grasshopper are automatically recognised in the Revit software. Therefore, three-dimensional models are edited simultaneously in Rhinoceros and Revit, thus creating a link between the two software, and all objects are fully

parameterised. The main step pipeline for producing a model from scanning to HBIM can be summarised as shown in Figure 5.5.

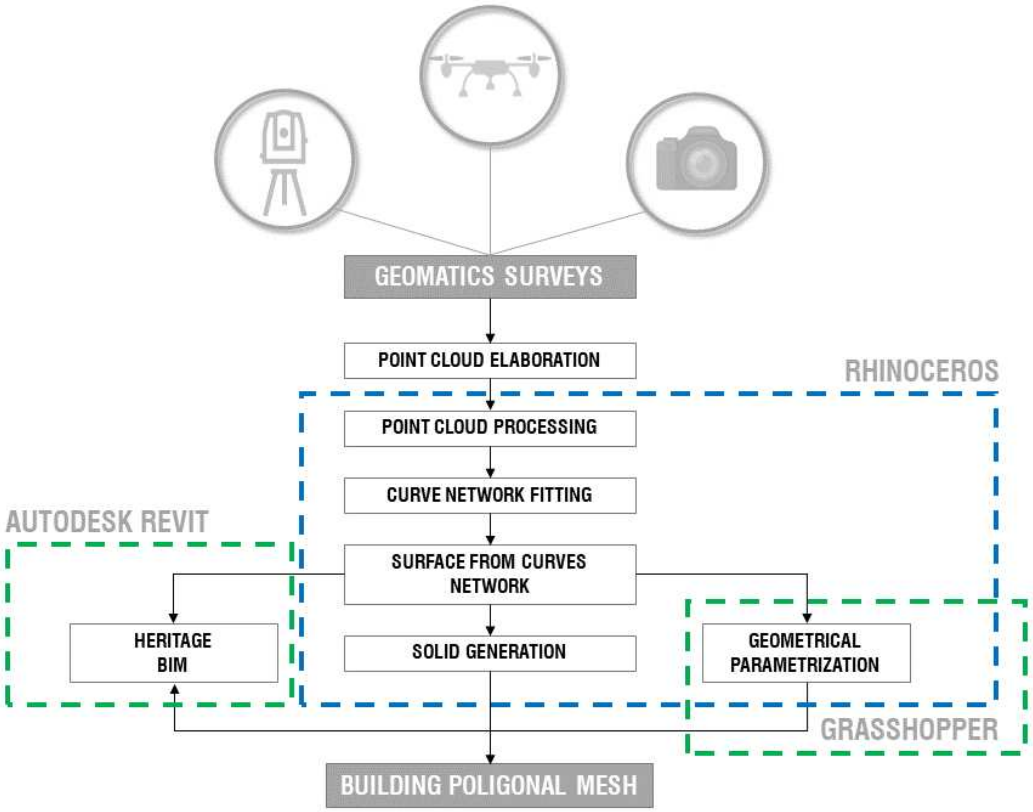


Fig. 5.5 - Pipeline of developed method.

In this way, it is possible to design a working environment for the creation, modelling and digital management of cultural heritage. In addition, semantic information from other environments can be linked. For example, historical or architectural information can be integrated into the BIM model within the Revit software through the use of a hyperlink field. In this way, the objects created can be continuously updated in terms of geometry, materials and semantic information.

5.3.2. Parametric modelling of the church

The HBIM model of the set of objects that make up the Church of St Nicholas in Montedoro was divided into two parts. The first part concerns the modelling of the structure, i.e. the modelling of the walls, arches, doors and windows, while the second, concerns the objects and elements of particular value inside the church.

In this way, objects such as altars and frescoes are placed on wall elements or arches and can be modelled and represented within the same parametric model as in reality. The point cloud obtained from the geomatic survey was imported into Rhinoceros software. To create the parametric objects of structural elements, such as walls, a particularly useful tool was Arena4D; it is a comprehensive software package capable of visualising and editing huge data sets of point clouds, images, panoramas, shape files, GNSS tracks and much more in a single contextual view. In addition, this software, developed by Veesus, allows point clouds to be marked, annotated, measured and even edited, and then exported into several common formats.

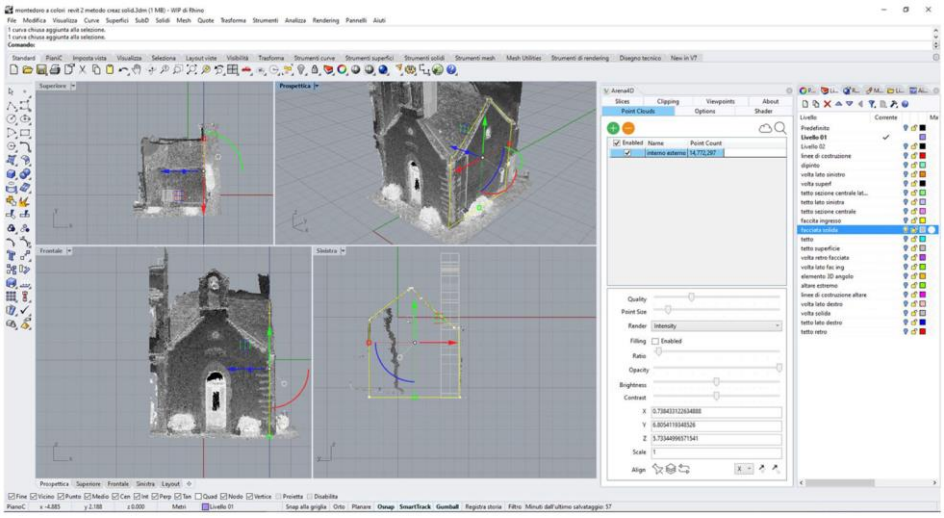


Fig. 5.6 - Profile on the point cloud in Rhino environment to build 3D objects using Arena4D tools.

Once the point cloud is loaded into the Rhino environment, the plugin also offers powerful tools for manipulating the point cloud, such as slicing, clipping, and smoothing algorithms. For the case analysed, a much-used command was slicing. In fact, this

tool allows the point cloud to be cut into various sections. Before performing this operation, it was necessary to transform the point cloud in LAS format into VPC format. This conversion operation is necessary because it is necessary to convert the point cloud into the optimised Veesus Point Cloud (VPC) format. By importing this point cloud into Rhino, profiles were created in certain sections, as shown in Figure 5.6.

Through this last plugin it was possible to control the point cloud, reduce noise, control shading, etc.; it was also possible to rotate the point cloud and adapt it to the desired directions. A series of section planes were set up, which were fully controllable in space, and as a result, the dense point cloud was sectioned. All realised sections were transformed into one-dimensional geometric elements such as splines and polylines. The 1D lines through two-dimensional modelling commands were subsequently transformed into 2D entities, i.e. NURBS surfaces. Finally, for some monumental architectural elements, it was possible to switch from 2D surfaces to 3D solid models using various available Boolean commands. In areas where the structure was characterised by a high degree of complexity, a very large number of sections had to be created in order to improve the quality of the final model. In addition, for the reconstruction of complex geometries, an additional, highly advanced modelling tool was used in the generation of NURBS surface entities, which is capable of reconstructing highly irregular NURBS surfaces by readjusting them. Using this approach, it is possible to model and reconstruct not only the structural elements, but also all architectural and interior elements of the structure. As far as topological aspects are concerned, the different 3D structures realised (walls, architectural elements, etc.) do not intersect each other. This condition of non-intersection is a consequence of the very method by which the structural elements were realised, such as the extrusion of splines (curves) obtained directly from the intersections of the planes with the point cloud. The intersection lines are thus generators of multiple surfaces that are in contact with each other but cannot intersect.

In general, in order to validate the final parameterised model, a geometric validation can be performed by comparing the mesh obtained through modelling in Rhino with the point cloud generated in the geomatics survey.

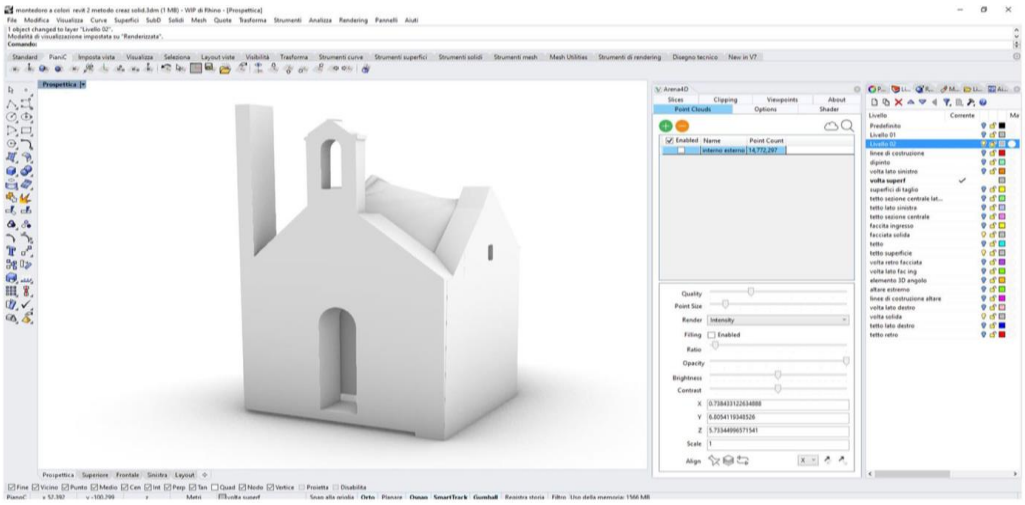


Fig. 5.7 - 3D parametric model of the church (external part) realised in Rhino software.

The parametric 3D model created, thanks to the "Rhino.Inside.Revit" plugin, made it possible to transform the model generated with Rhinoceros software into Autodesk's Autodesk Revit® software (Figure 5.7).

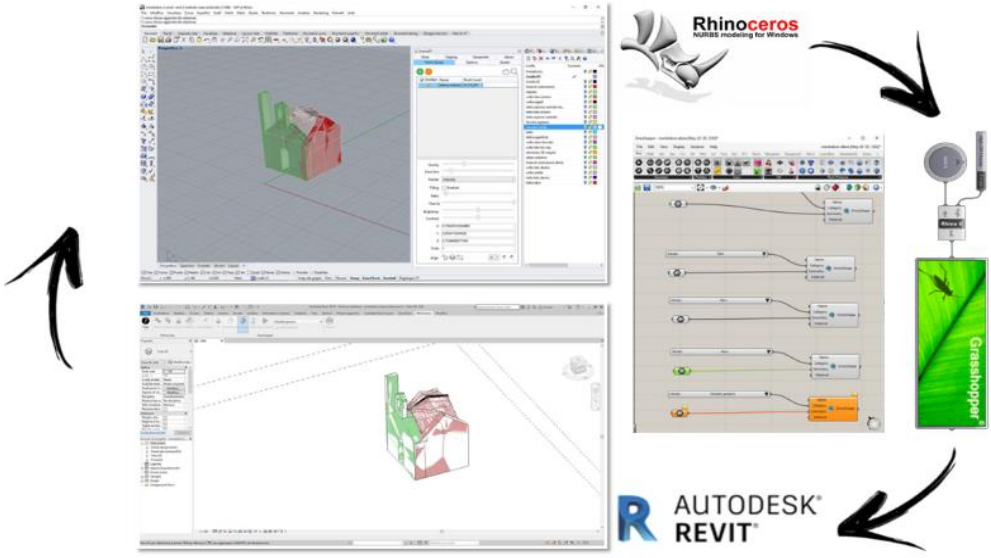


Fig. 5.8 - Simplified two-way workflow for HBIM model.

In fact, Grasshopper's programming language allows the transformation and modification of objects from one software to another (Figure 5.8). The workflow just described is reversible, i.e. a model created in Autodesk Revit® can be transferred to Rhinoceros via a different programming structure in Grasshopper.

5.3.3. Parametric modelling of the church

Regarding the modelling of the different elements in the church, it was necessary to carry out a more detailed modelling of the structure in order to construct an HBIM model. By setting a series of section planes along an object of which you want to reconstruct a 3D model, you can reconstruct a series of profiles. This approach was used for the reconstruction and modelling of all architectural elements within the church, such as the altar. Once the profiles to be analysed had been determined, the extracted sections were transformed into parametric three-dimensional objects using Boolean tools such as extrusion and lofts. In this way, it was possible to obtain a fully parameterised 3D model in terms of both geometry and material (Figure 5.9).

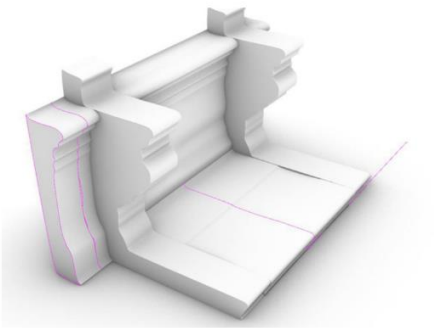


Fig. 5.9 - 3D representation of the altar in Rhinoceros environment.

In fact, each object reconstructed in 3D is defined in its dimensions and it is possible to associate the type of material of which it is composed. The surfaces thus generated in Rhinoceros are associated with Grasshopper, which is able to link them to Autodesk Revit® software (Figure 5.10). Adopting the same modelling approach as for the altars, the plaster layer containing the frescoes was modelled.

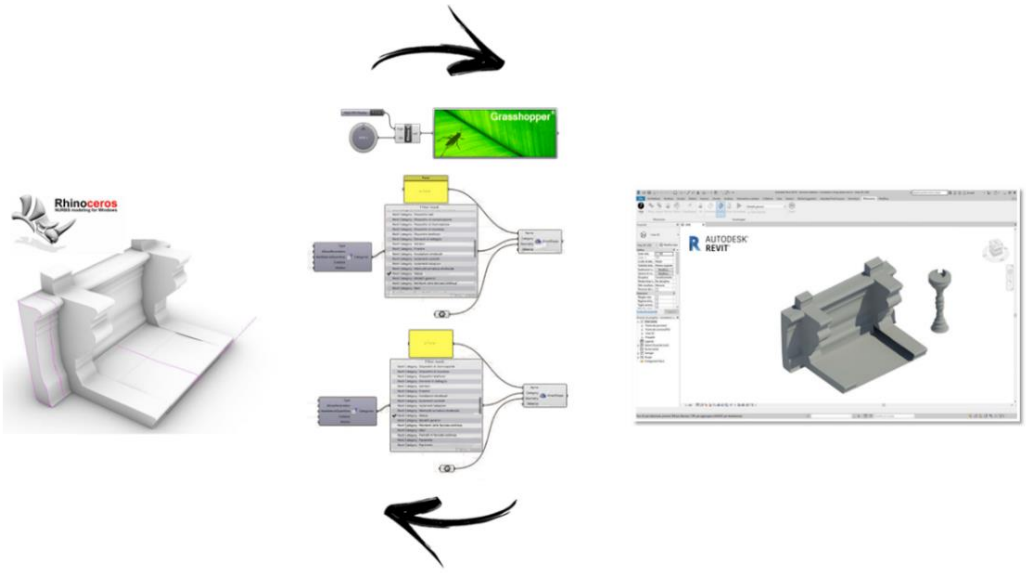


Fig. 5.10 - Transformation of the 3D model of the altar from Rhinoceros to Revit® with processing in Grasshopper.

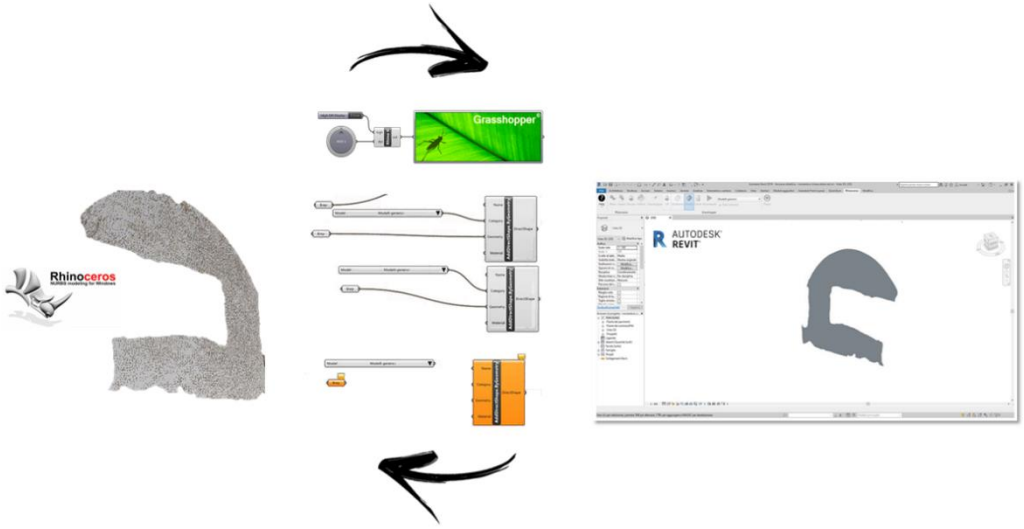


Fig. 5.11 - Transformation of the 3D model of the plaster of the frescoes from Rhinoceros to Revit® with processing in Grasshopper.

As with the altar, after parametric modelling through NURBS in the Rhino environment, the frescoes were also managed in an HBIM environment on the Autodesk Revit® platform. In fact, the modelling of the frescoes (Figure 5.11) in the HBIM environment made it possible to manage information on the characteristics of the materials and the geometry of the surfaces. This is particularly useful for archiving and sharing geometric and semantic information to manage elements of particular historical-architectural interest.

This can be achieved by linking objects to other external files via a 'hyperlink' field. Consequently, historical and architectural information can be associated with each object. This can be achieved through the construction of a simple local database or a centralised one, i.e. made with standard specifications in which all objects present on the national territory and recognised as belonging to the cultural heritage are included (Costantino et al., 2021).

In fact, a new General Catalogue Information System (SIGEC) has recently been created in Italy, which offers the opportunity to reflect on the current meaning of the Catalogue of Cultural Heritage as laid down in the Cultural Heritage and Landscape Code. SIGECweb was created with the aim of unifying and optimising the processes of cataloguing cultural assets, guaranteeing, through the control of the procedures applied, the quality of the data produced and their compliance with national standards. Therefore, through the URL (Uniform Resource Locator) link, it was possible to connect the information of each model to a central database. In this way, the homogeneity of information is guaranteed; the latter being a prerequisite for the immediate availability, correct use and sharing of cultural heritage information. However, the hyperlink command can only be activated on objects of the Autodesk Revit® family.

5.4. Model optimisation for structural analysis

A parametric polygonal 3D model thus realised has the advantage not only of being performant in BIM design processes, but also, if suitably processed, becomes supportive in structural analyses. To date, no software is yet able to perform this function starting directly from a geomatic survey, so a series of prior elaborations and a

series of simplification processes on the mesh (smoothing algorithms, simplification, noise reduction, etc.) are required to make the model fit for purpose.

The methodological approach used in this case study, which leads to the creation of an HBIM model, becomes more relevant when this model can be used in structural analysis software based on the FEM (Finite Element Methods) method. The Finite Element Method is the most widely used method for analysing and solving engineering problems and mathematical models, such as structural analysis, heat transfer, fluid flow, etc.

Thanks to the recent development of computer tools, FEM analysis has developed exponentially today, involving various sectors and fields to which its principles can be applied. In fact, the use of this methodology has established itself as one of the best tools for analysing complex systems, for which investigations and experiments in the laboratory would entail excessive expense, logistical difficulties and difficulties related to the physical measurement of the various quantities. If the first automatic approaches for solving the differential equations governing physical phenomena were established with finite differences, the FEM evolves the possibilities of solution by giving an application possibility that is unparalleled, thanks to its unquestionable flexibility. The generality of the method allows for many studies and applications, paving the way for new strands of research that currently address issues of considerable interest of a theoretical and practical nature.

In contrast to FDM, which sees the domain to be analysed as a series of points in a lattice, FEM sees the domain as the union of many sub-domains of elementary form. This finite element discretization procedure reduces the problem to a finite number of unknowns, dividing the domain into finite elements and expressing the unknown field in terms of approximating functions defined within each element. Approximating functions, also called shape functions, are identified by the values that the dependent variable takes on at specific points called nodes. Nodes are usually placed on the contour of elements, at points common to two or more elements.

The FEM method therefore subdivides the geometric model into many small elements that can be calculated easily, using a geometric grid or mesh model. The final

solution corresponds to the sum of all partial solutions calculated for each element. Thanks to an FEM analysis, displacements, possible deformations and/or stresses in a structural system can be revealed.

As shown in Figure 5.12, the FEM analysis process is divided into 3 phases (highlighted in red, yellow and green in the pipeline in the figure).

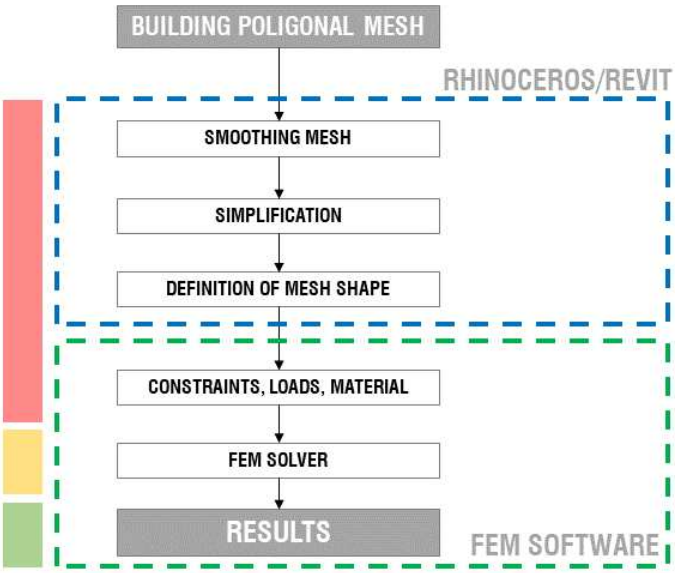


Fig. 5.12 - FEM analysis process pipeline.

1. Pre-Processing: In this first phase, the finite element model is defined. A series of elements are defined such as the type of analysis to be performed (static, dynamic, thermal, linear, etc.), the type of finite elements, constraints, loads and material parameters. This phase is very important as the model to be analysed is broken down into smaller elements in order to reduce the complexity of the system. It is therefore necessary to simplify the input parametric mesh model in order to guarantee an optimal result of the entire process. The choice of representative mesh shape and the degree of mesh simplification must be carefully evaluated in order to guarantee the best end result. Generally, the shapes chosen for decomposition are elementary shapes; in the present case, a triangular mesh was used. Then the behaviour of each individual

element is studied: how it reacts to a stress, how it deforms, etc.; On each individual element reduced to an elementary form, the solution to the problem is expressed by the linear combination of functions known as basis functions or shape functions.

2. Processing: In this central and decisive phase, the structural calculation software with integrated FEM solver intervenes, which has the task of solving systems of partial derivative equations (or PDEs) in discrete and approximate form on the previously discretised computational domain.

3. Post-Processing: the third and final phase corresponds to the processing and graphic visualisation of the results obtained.

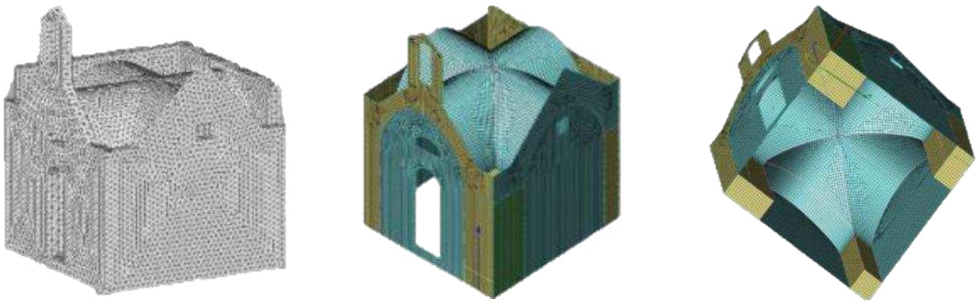
5.4.1. From parametric mesh to FEM analysis: a first approach

Finite element analysis (FEA) is performed on a mesh composed of several triangular and quadrilateral finite elements that subdivide an original body. The finer the mesh is, the more accurate the numerical results will be, but the calculation time will also be longer. A balance between mesh size, calculation time and accuracy of results is an important characteristic of a well-defined finite element analysis. The polygonal mesh model was imported into the FEM environment using the Midas GTS NX software developed by MIDAS Information Technology Co. This application is a complete software package for finite element analysis, capable of handling the full range of structural design applications. The procedure to generate structural information from the 3D model consists of several steps:

- import of surfaces;
- creation of structural meshes;
- assignment of materials and loading conditions.

In the first step, surfaces were imported into the software (Figure 5.13 a) and structural meshes were generated (Figure 5.13 b, c). Subsequently, the conditions of external weight, internal weight of the structural elements and accidental load were assigned to the structure. As far as materials are concerned, the customised information for each material can be previously assigned directly within Rhinoceros via the

"VisualARQ" plug-in. Object styles with these customisations can be exported, in IFC (Industry Foundation Classes) format, in one go into the Revit software.



(a) (b) (c)

Fig. 5.13 - Importing 3D model into structural analysis software: structural mesh (a) front axonometric view (b) and bottom view (c).

These objects, recognised on the basis of style and customised information (material properties, cost per unit and customised metric information associated with any object in the model), are further enriched through the Revit libraries with material characteristics useful for volumetric, thermal, computational and maintenance processing.

The parametric 3D model created in Rhinoceros was imported into Midas GTS NX using the "STEP" and "Parasolid" formats. The imported object is congruent, and all its structural parts are correctly connected. The parametric model, however, consists of a single solid of a single material. By means of specific Boolean operations, such as 'divide solid', it is possible to divide and self-connect the different surfaces. Therefore, each of the generated structural parts will receive the appropriate structural material characterised by the appropriate constitutive relationships (Mohr-Coulomb, Drucker-Prager, Von Mises).

The aim of this work was to identify a specific procedure that was deemed most suitable for moving from a 3D point cloud representation, obtained through a geomatic survey, to a 3D model that could be managed in both H-BIM and H-FEM environments.

6. SCAN TO H-FEM PROCESSES: MESH OPTIMISATION FOR SEMANTIC 3D MODELLING

The current exponential growth in computer processing capacity and constant technological evolution have made it possible to obtain photorealistic and metrically accurate 3D models (Kabady et al., 2020). Three-dimensional digitisation and modelling techniques enable the faithful reconstruction of the shape of objects or structures and is widely used in various fields of application, such as:

- the creation of prototypes to support museum activities and historical collections.
- Production of multimedia content and prototypes for web and app representation.
- Production of three-dimensional scale prints for cultural dissemination and tactile support for the blind.
- Study of an artistic asset from a topological and conservation point of view.
- 4D monitoring for conservation status assessment.
- Creation of a data archive to support museum activities and historical collections for the evaluation of the state of preservation and the analysis of all possible changes.

As seen in the previous chapters, three-dimensional modelling can be performed using modelling or image-based techniques (Baik et al., 2021). An example of range-based modelling (cf. 1.1) can be performed by the Terrestrial Laser Scanner (TLS), which allows us to quickly determine the position of millions of points that precisely define and accurately reproduce the surface and geometry of the surveyed objects (Artese et al., 2019).

An example of an image-based technique, on the other hand, is Structure from Motion (SfM), which allows us to extract many points from individual photos, calculate camera calibration parameters and thus identify recognisable points on several photos

(Schonberger et al., 2016), ultimately resulting in a dense, geometrically accurate point cloud capable of representing the detected object with a high degree of detail (cf. 2.2).

These techniques can also be integrated with each other in order to improve the quality of the resulting 3D model and accurately reconstruct even those portions of the object that would not be detectable using only one surveying technique.

In fact, the TLS method integrated with close-range digital photogrammetry, in the field of the existing historical-building heritage, makes it possible to elaborate an accurate 3D model and perform a series of structural analyses and to assess the effects of deformations (Bertolini-Cestari et al., 2013).

The use of such instruments or the integration of innovative and established investigation techniques for 3D metric modelling, also represent a great support in conservative restoration activities and provide useful and accurate information in the reconstruction of missing or disfigured parts, especially in activities related to the preservation and conservation of existing cultural heritage.

Many applications and researches have been conducted in order to improve the accuracy of the model and the results obtained, especially in the field of Cultural Heritage and in particular the evaluation of the mechanical behaviour of a marble statue characterised by localised fractures and cracks, the mapping of the state of deterioration of stone elements on 3D models (Adamopoulos et al., 2021).

In general, the choice of one of these techniques is not a simple process and is linked to the complexity of the structure to be analysed; for this reason, the choice of the best technique or the integration of different techniques is an important aspect to take into account during the planning phase of the survey.

In this chapter, the case study of a statue of particular historical-architectural interest will be dealt with, and an innovative methodological approach will be illustrated that makes it possible to obtain, starting from photogrammetric and/or TLS surveys, simplified and lightweight 3D models that are at the same time highly performing in the evaluation from a structural point of view.

In fact, this research aims to contribute to the identification of efficient processes in the application of the scan-to-FEM process applied to cultural heritage, by

identifying innovative and high-quality geometric methods and criteria for modelling structures. In particular, aspects related to the construction of a parametric model for FEM analysis will be addressed, starting from a model with a rather complex geometry, such as that of a statue.

6.1. An overview of methodology in the field of CH

The integration of the previously described acquisition techniques have been applied in different CH contexts by the scientific community.

In order to highlight on a digital 3D model the portions of a statue in need of restoration and map the metadata directly within the model, the Statue of Hope located in the English (Protestant) cemetery in Florence was surveyed (Barone et al., 2012). Given its sculptural form characterised by small details distributed over large areas, the integration of a 3D structured light scanner and a stereo photogrammetric sensor were used with the aim of reliably and accurately reconstructing the sculpture.

Once the point cloud has been processed, it is possible to generate a mesh model that defines a polygonal surface (triangular or polygonal mesh). In this way, it is possible to construct a high-resolution 3D model with photorealistic content suitable for structural analyses of structures.

However, this model, although geometrically accurate and very detailed, is very heavy from a storage point of view and at the same time, difficult to handle from a computational point of view. For this reason, it is necessary to transform the point cloud obtained from the survey into a 3D surface suitable for the subsequent processing and analysis phases. In particular, knowledge of the structural behaviour of particular structures plays an important role in the conservation and preservation of cultural heritage. The often-slender shapes and the masses involved generate high stresses not only related to the static field, but also linked to other natural and environmental factors, such as earthquakes or other vibrations.

It is therefore necessary to define a methodological approach that, starting from an accurate representation of the model, can provide structural (static and dynamic) information on the element examined.

This can be achieved by means of FEM (Finite Element Analysis), also known as FEA (Finite Element Analysis) since it is possible to study complex physical phenomena that could otherwise only be addressed by an experimental approach.

A method of transforming the point cloud into a 3D model suitable for structural simulation (based on Finite Element Analysis) was conducted on a castle in the city of Sondrio (Italy). Starting from the point cloud obtained from a TLS survey, they performed two different types of analysis on the tetrahedral mesh model: a static analysis to calculate plate stresses and a dynamic analysis simulating the effects on the historical structure due to an earthquake (Barazzetti et al., 2015).

A further application, in this context, was developed, using data acquired from a TLS survey, of the Revedin-Bolasco villa; starting from the point cloud, a portion of this structure was modelled and subsequently used to perform a finite element analysis in order to assess the effects of the external walls and the weight of the roof on the structure itself (Guarnieri et al., 2017).

Digitising and 3D modelling processes as a suitable tool for planning conservation measures were also used to interpret the results of natural stone sculptures by setting up a comparison between 2D and 3D surfaces (Pfeuffer et al., 2018) or to transform the point cloud into a FEM model. To perform this transformation, NURBS (Non Uniform Rational B-Splines) surfaces implemented in Rhinoceros software were constructed to generate the parametric model of a stone arch bridge from the late 1800s.

Further scientific research conducted in this field was to apply the methodology of non-destructive measurement design using an HBIM approach (Santini et al., 2021); starting from historical, structural and design information, it was possible to reconstruct and model the information obtained from the geometric and material survey, and then convey it into a finite element model built on Midas Fea NX.

Today, thanks to the potential offered by software capable of managing mesh models and defining a parametric modelling approach, it is possible to rapidly transform point clouds into geometric entities. In fact, starting from a 3D survey and through generative programming, it is possible to define a parametric Scan to FEM approach applied to architectural heritage (Funari et al., 2021).

Regarding the modelling and FEM analysis of cultural heritage statues, a very important example of the contribution of geomatics to the structural analysis of statues is the 3D model of Michelangelo's David. This model was obtained from a laser scanner survey carried out by ISTI-CNR in collaboration with Stanford University, which provided a 3D model composed of over 50 million triangles, subsequently simplified to 20,000, 50,000 and 100,000 triangles to facilitate structural analysis calculations (Borri et al., 2006). The latter model was imported into FEM software to analyse the injuries and stability of Michelangelo's David.

Another important contribution in the Scan to FEM process was addressed on a statue of Emperor Claudius preserved in the National Archaeological Museum in Aquileia (Visintini et al., 2014). The statue was scanned with laser scanner technology and modelled with MeshLab software. The structural analyses of the surface model were imported into Rhinoceros through the use of the Scan&Solve plugin: the most stressed areas were then evaluated, and a hypothetical model was analysed to assess the influence of a rear support on the equilibrium of the entire structure. Using a similar approach, a FEM analysis was carried out using the Rhinoceros plug-in to assess the possible risks of the marble statue of San Giovannino Martelli (St John the Baptist) preserved in Florence (Spangher et al., 2017).

In order to make the best use of computational resources for solver accuracy and congruence between geometric resolution and FEM, the three-dimensional model was simplified by considering only 7 per cent of the initial triangles.

Recently, other studies and research based on geomatic survey techniques and 3D modelling have been conducted on statues such as, for example, the statue of Tullio Lombardo (Riccardelli et al., 2014) or the central sculpture of the Fountain of Juno by Bartolomeo Ammannati (Bici et al., 2018). On the former, preserved at the Metropolitan Museum of Art in New York, static and modal analyses were carried out to identify the statue's weak points and define its correct positioning; in the latter case, in order to construct the FEM element on which to conduct the analysis, the 3D geometric model was obtained by laser scanner survey.

6.2. Workflow of the proposed method

The method proposed in this chapter involves, starting from a geomatic survey, the elaboration of an optimised 3D model for its handling in an FEM solver.

The first step is to identify a suitable geomatic surveying technique for the 3D point cloud reconstruction of the object to be analysed. Using dedicated software, once the dataset has been processed, it is then possible to reconstruct a 3D model characterised by a TIN surface. To improve and optimise the handling of this model in the FEM environment, it is necessary to transform it into a polygonal surface and then decimate it. Therefore, the 3 main steps in the process leading to the FEM analysis of the structure under consideration are as follows:

1. construction of the 3D model
2. transformation of the TIN into a quadrangolar mesh and NURBS model;
3. importing the model into an FEM solver.

6.2.1. Photogrammetric survey and point cloud processing

In the construction of the 3D model, the first step concerns the identification and planning of a suitable surveying technique. The 3D model of the statue can be constructed using both TLS and photogrammetric techniques; however, taking into account the inexpensiveness of the photogrammetric survey and the possibility of acquiring information on the geometry of the statue in a rather simple and detailed manner, this technique was chosen for the survey of the Colossus.

In addition, Close Range Photogrammetry techniques allow an accurate and detailed model to be obtained; in an SfM and MVS environment, it is therefore possible to obtain a dense and coloured point cloud capable of reconstructing a detailed and realistic 3D geometry of the structure.

In order to obtain an adequate coverage of the statue and an adequate overlapping of the frames, in order to have a perfect reconstruction of the finest details, a tripod extendable up to a height of 6m was used in the case study analysed. In this way it was possible to obtain an adequate number of images of even the tallest elements of the structure.

For the scaling and referencing of the model, a topographical survey was conducted with Total Station, which made it possible to detect a series of GCPs and CPs evenly distributed around the scene to be reconstructed.

The resulting dataset was processed with photogrammetric software (in particular, Agisoft Metashape v.1.5.1 was used for this case study).

In general, the various processing steps leading to the construction of the photogrammetric model in the SfM/MVS environment are (see 2.3.1):

- the alignment of images,
- the construction of a dense point cloud (PC),
- the application of a colour filter to remove noise due to the sky and the manual editing of points outside the statue,
- mesh construction and
- identification of the planes of the individual façade for the construction of a series of orthomosaics in the different projection planes.

The first step in the photogrammetric process is the calibration of the camera. Using the Agisoft Lens tool (v. 0.4.2), it was possible to calculate the camera's main parameters such as focal length, main point coordinates and the associated radial and tangential distortions. Subsequently, the calibration parameters were imported into the software in order to improve the quality of the photogrammetric process.

Following this step, image registration was performed, which, through the SfM algorithms, produces a sparse cloud that defines the model space. The reliability of the id-alignment process is assessed using GCPs and CPs.

To achieve this aim, the accuracy of the process was evaluated using the GCPs and CPs determined in the topographic survey phase. The total error, i.e. the mean square error for the x, y and z coordinates, can be calculated from the following formula:

$$T.E. = \sqrt{\sum_{i=1}^{n} \frac{\left(x_{i,est} - x_i\right)^2 + \left(y_{i,est} - y_i\right)^2 + \left(z_{i,est} - z_i\right)^2}{n}} \tag{43}$$

where:

x_i, y_i, z_i input values of the coordinates x, y, z referring to the i-th camera;
$x_{i,est}, y_{i,est}, z_{i,est}$ values of the coordinates at their estimated position.

6.2.2. Mesh simplification: from TIN to NURBS

From the point cloud, a triangular surface mesh can be generated; this surface can be textured to obtain a photorealistic digital representation of the object/scene. The irregular mesh (TIN) is a surface in which the known points in the three coordinates, however they are distributed in space, are joined by lines to form flat, adjacent triangles that allow the surface of the object to be represented with continuity. However, in some cases, the triangular model does not allow the object or structure to be represented according to a closed mesh. Moreover, the elements that make up the object cannot be treated (and thus modelled) as separate entities. Furthermore, the triangular mesh generated by the photogrammetric (or TLS) process leads to a difficulty in handling the model in relation to the number of triangles. This becomes particularly important when the geometry of the structure or object is rather complex. For this reason, the next step is to transform the TIN model into a square polygonal mesh model, referred to as a quad-mesh.

A quad mesh is semi-regular if it is obtained by gluing together, in a conformal manner, several regular 2D arrays of quads from side to side. Each of these regular sub-meshes is called a patch and the number of patches is assumed to be much smaller than the total number of facets (cfr 4.3.1).

Square meshes are often preferred in some numerical analyses, as they reduce both the approximation error and the number of elements compared to triangles. Consequently, taking into account the complexity of the statue geometry, the quad mesh is a valid solution for modelling this type of complex structure. In this way, it is possible to achieve accurate modelling of each component of the surveyed object and to avoid the problem of working with a mesh that is not properly structured and difficult to handle, both for modelling and for the computational aspects related to the subsequent FEM analysis. In the context of FEM, a semi-regular mesh is called a 'multi-block grid', i.e. it represents blocks corresponding to patches.

A modelling software that allows the transformation and management of square polygonal meshes is Rhinoceros. In order to further reduce the mesh to be analysed and decrease the calculation power required for the finite element analysis, the quadruple mesh was decimated using a tool contained within Rhinoceros.

To geometrically validate the model, a comparison was made between the quadruple mesh generated in the Rhino environment and the point cloud obtained from the photogrammetric survey.

A cloud to mesh (Cloud to Mesh - C2M) comparison was performed using the Cloud Compare software (see 2.11); using the C2M algorithm, it is possible to calculate the distance between the point cloud and the vertices of the mesh that make it up (Girardeau-Montaut, 2016). The C2M distance is calculated according to the equation:

$$D_{2M} = \sqrt{(X_M - X_C)^2 + (Y_M - Y_C)^2 + (Z_M - Z_C)^2} \quad (44)$$

where:

X_M, Y_M, Z_M coordinates of the vertices of the mesh polygon;

X_C, Y_C, Z_C coordinates of the point cloud.

Distance estimation (Figure 6.1) is, of course, strongly influenced by the quality of the mesh, which may have roughness or approximations related to the degree of detail of the 3D model reconstruction process.

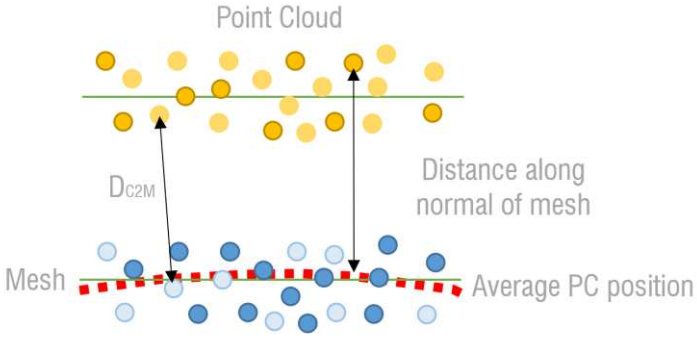


Fig. 6.1 - Illustration of the relationship between cloud and mesh (C2M), calculated between the closest points along the local normal of the mesh.

The conversion operation from a TIN model to a NURBS model may result in a non-manifold mesh, or a mesh with an edge shared by several faces that can be removed at a later stage by means of appropriate commands. This operation is of fundamental importance in order not to create errors in the subsequent 3D model import phase, and thus to produce a continuous model suitable for FEM analysis. Finally, the square-mesh model can be transformed into a NURBS surface, i.e. understood as a generalisation of a b-spline curve (composite curves connecting several Bezier curves) capable of faithfully representing the surveyed object (cfr 4.4.1).

6.2.3. Management of the semantic 3D model

Once the three-dimensional, parametric model has been generated, it can be imported into software based on the FEM approach. Various tools and software are available on the market for importing 3D meshes for FEM analysis. Midas FEA NX represents, for example, FEM analysis software designed for the analysis of complex geometries. Within the software, a series of parameters can be assigned to the model in order to define its material composition and consequently conduct the structural analysis. The construction of the FEM model requires knowledge of a parametric model, i.e. a model in which each object is described according to its dimensional and material characteristics. The methodological approach developed and applied to this case study of a heritage statue can be schematised in the following pipeline (Figure 6.2).

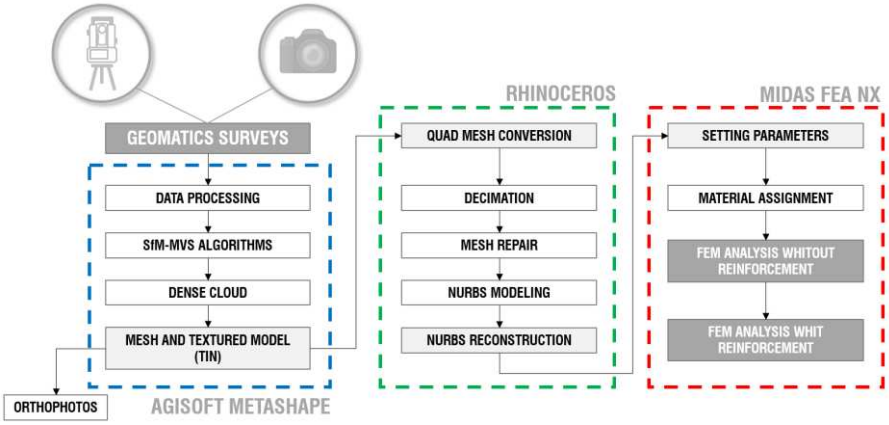


Fig. 6.2 - Illustration Pipeline of the developed and experimented methodological approach.

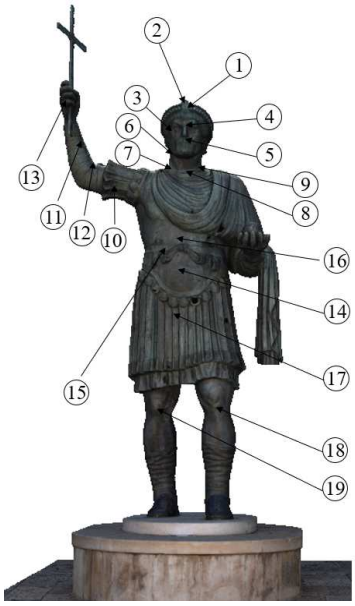
6.3. From point cloud to FEM model: the case study of "Colossus of Barletta"

The "Colossus of Barletta", located in the homonymous city of Barletta, Italy, is a gigantic bronze statue, about 5 metres high, dating from the 5th century (Figure 6.3). The emperor is depicted as "Imperator", one of the standard patterns for imperial representation (loricate statue). He wears a tunic, muscular cuirass with cingulum and paludamentum; the separately inserted fibula has been lost (Kiilerich et al., 2015). The work of Byzantine construction probably depicts Emperor Theodosius II. There are currently several versions of the statue's creation. The only certain information dates to the year 1309, when some religious officials in the area asked and obtained permission from Charles II of Anjou to remove and cast the limbs of the statue to make some bells to be placed in their church.

Fig. 6.3 - Geolocation of the statue called "Colossus of Barletta".

Recent studies have shown that the head and limbs are original, while the legs have a later date. Several studies have been conducted on the statue to characterise the casting materials used and their dating. In fact, non-destructive Energy Dispersive X-Ray Fluorescence (EDXRF) analysis carried out on the structure showed that the statue was made from different bronze alloys (Figure 6.4): a copper-rich alloy for the

face and hands, a second high-tin alloy for the crown, and a third high-lead alloy for the bust; as for the lower limbs, historians confirm that they are not coeval with the rest of the work (Buccolieri et al., 2015).



n.	Descrizione	Cu	Pb	Sn	Cl	Fe	Ca
1	Diadem	39.2	17.7	19.7			
2	Diadem	49.0	29.6	5.7			
3	Cheekbone dx	59.2	17.1		0.3		
4	Nose	63.8	20.6	2.1			
5	Nose	59.5	11.4	5.0	0.5		
6	Neck	57.7	22.5		3.6		
7	Neck	58.8	18.1		3.8		
8	Tunic	46.3	45.5		4.4		*
9	Neck - Tunic	44.8	36.3		1.2	4.1	**
10	Arm sleeve dx	46.2	36.5	1.3			
11	Forearm dx	43.3	37.7	3.4			
12	Plaster	15.0	10.0				*
13	Hand dx	63.4	15.9		6.7		
14	Abdomen	47.0	36.7		1.6		**
15	Belt	30.6	50.8		0.6		*
16	Belt	47.6	42.1		1.0		*
17	Kilt	50.5	40.0		1.6	1.0	*
18	Knee sx	57.8	24.3	1.7	5.1		*
19	Knee dx	53.5	31.4	4.2	0.9		***

Fig. 6.4 - Characterisation of the bronze statue's constituent materials obtained by EDXRF analysis.

Moreover, according to thermoluminescence tests of the casting earths of the statue conducted by the Laboratory of Archaeometry at the University of Milan-Bicocca in 2017, as well as a classification of the casting earths used, they conclude that the casting of the statue's torso most likely took place between the 4th and 6th century AD, excluding earlier and later dates. As for the limbs, however, the casting most likely occurred in the second half of the 14th century AD.

The monument underwent structural restoration in 1981, during which the Central Institute for Restoration in Rome carried out cleaning treatments and consolidation of the patinas. In addition, during the same restoration, the Colossus of Barletta was fixed to a steel structure (completely prefabricated with a steel structure on a wooden model) placed inside the statue.

6.3.1. Topographic and photogrammetric survey

In order to survey the GCPs, distributed along the statue, a topographic survey was carried out using a Leica Total Station mod. TS11, characterised by an angular accuracy of 1" (0.3 mgon) and a linear measurement accuracy with a prism of 1 mm + 1ppm and without a prism of 2 mm + 2 ppm.

For the determination of the GCPs, the polar coordinates of the targets were made without the use of prisms; using the dedicated software, the Cartesian coordinates of the identified points were obtained for the georeferencing and scaling of the model. In this way, the survey of the photogrammetric targets positioned along the base of the statue was carried out; in particular, 7 GCP (Ground Control Points) were surveyed. About the vertical development of the statue, 6 “natural” points on the statue were surveyed. In addition to the GCPs and natural points just described, several CPs (Check Points) and some metric bars were detected (Figure 6.5).

Fig. 6.5 - Geomatic survey: acquisition phase of GPCs and CPs.

After the topographic survey phase, the photogrammetric survey of the statue was carried out using a Nikon D3300 model reflex camera with a resolution of 24.2 Megapixels with a Nikkor 20 mm f/2.8D fixed focal length lens.

Considering the height of the statue, a tripod/pole had to be used during the surveying phase to reach the desired heights. For the digital images, the camera was

equipped with a remote controller that allowed the acquisition of shots from a distance, using a remote control.

In particular, a maximum height of 6 m was reached at which the first images were acquired following an orbital trajectory around the statue. Subsequently, the tripod/pole was gradually lowered by 0.50 m at a time until the height of the base was reached. In addition, further images were taken from the ground in order to obtain a more accurate definition of the details and carvings of the statue.

A total of 960 frames were acquired (Figure 6.6). The Ground Sample Distance (G.S.D.) of the image obtained was approximately 1 mm/pix.

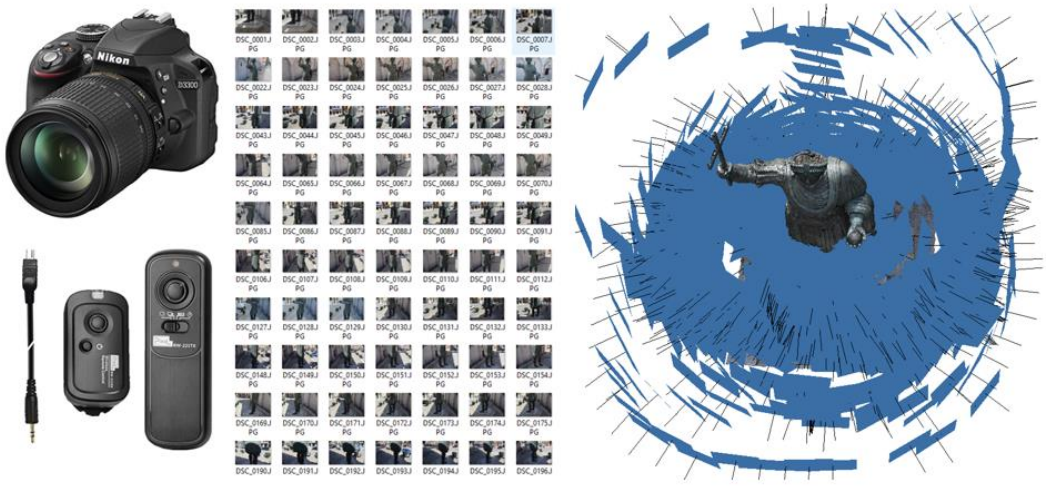


Fig. 6.6 - Photogrammetric survey: reflex camera with remote shutter device and geometry of captured images.

The dataset was processed in Agisoft Metashape, setting 'Highest' as the accuracy parameter in the image alignment phase. By using this parameter for alignment, the image is scaled (increased) by a factor of 4, allowing more accurate estimates of the camera position to be obtained; as constraint point positions are estimated based on the characteristic points found in the source images, it can be useful to zoom in on a source photo to precisely locate a constraint point. However, the highest precision

setting is only recommended for very sharp image data; in contrast, the 'High' or 'Medium' value causes the image to be resized (decreased) by 2 or 4 times respectively.

For a more accurate representation of the model, masks were placed on the photos to focus the algorithmic process on specific parts of the image. The creation of these masks, i.e. closed polygons drawn directly on each frame, makes it possible to isolate the portions of the scene acquired through the photographic capture, which are, however, not useful for the photogrammetric reconstruction process; this operation, although tedious, also makes it possible to reduce processing times and to considerably reduce the noise caused by elements external to the object under investigation.

After this phase, a dense point cloud consisting of 5,664,260 points was generated. Table 6.1 below summarises the parameters used, and the accuracies achieved in the photogrammetric processing phase, the errors, in metric and in pixel, for each control and verification point, and the total error.

Tab. 6.1 - Errors on GCPs and CPs and photogrammetric processing parameters.

ID	Type	Error [m]	Error [pix]	Parameter	Model
10	GCP	0.001	0.846		
12	GCP	0.001	0.898	**Alignment**	
19	GCP	0.000	0.831	Accuracy: Highest	
21	GCP	0.001	2.102	Point: 1,932,377	
1001	GCP	0.003	2.129		
1002	GCP	0.001	2.000	**Point Cloud**	
1003	GCP	0.004	1.636	Reconstruction:	
TOTAL ERROR		**0.002**	**1.607**	Medium	
C 1-2	MB	0.0001	-	Points: 5,664,260	
C 3-4	MB	-0.0003	-		
C 5-6	MB	0.0005	-	**Model**	
TOTAL ERROR		**0.0003**	-	Faces:10,775,553	
				Vertices: 5,388,628	

In addition to the dense point cloud, a textured three-dimensional triangular mesh (TIN) model and a series of high geometric resolution orthophotos were also generated, useful for a visual and interpretative analysis of the monument's state of conservation and, consequently, to support decision-making processes in targeted maintenance activities (Figure 6.7). In particular, the 3D mesh model elaborated, setting an accuracy value of 'Medium', which causes a resizing of the image by a factor of 16 (4 for each side), is formed by 10,775,553 triangles.

Fig. 6.7 - Orthophotos obtained from photogrammetric processing with a geometric resolution of 1mm/pix.

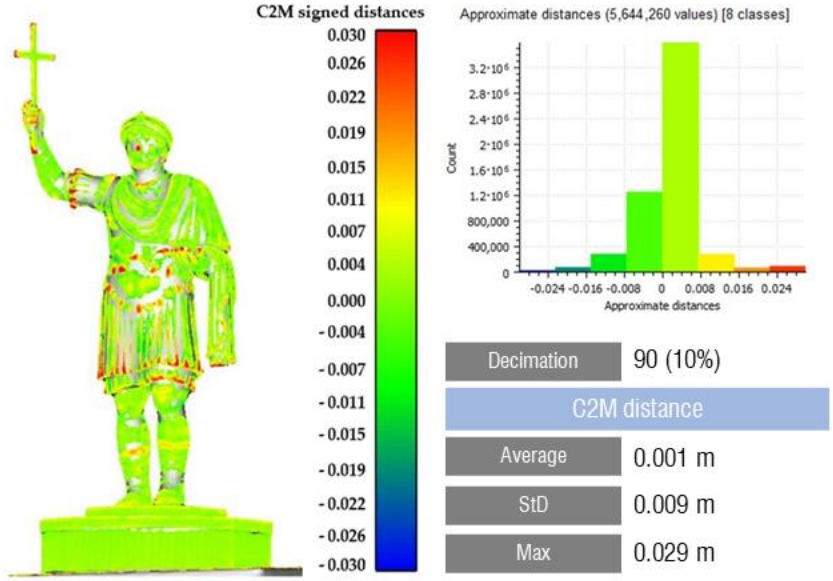
6.3.2. Polygonal parametric modelling and decimation

Through the process of photogrammetric processing, a high-resolution, triangular 3D mesh model was generated. For parametric modelling, this model had to be imported into Rhinoceros software. The import process, given the large number of triangles that made up the mesh, did not produce an acceptable result; this is due to the generation of a series of errors due to the structural elements of which the statue is composed not being perfectly closed. Moreover, the handling of a modelling composed of 10,775,533 faces is rather complex.

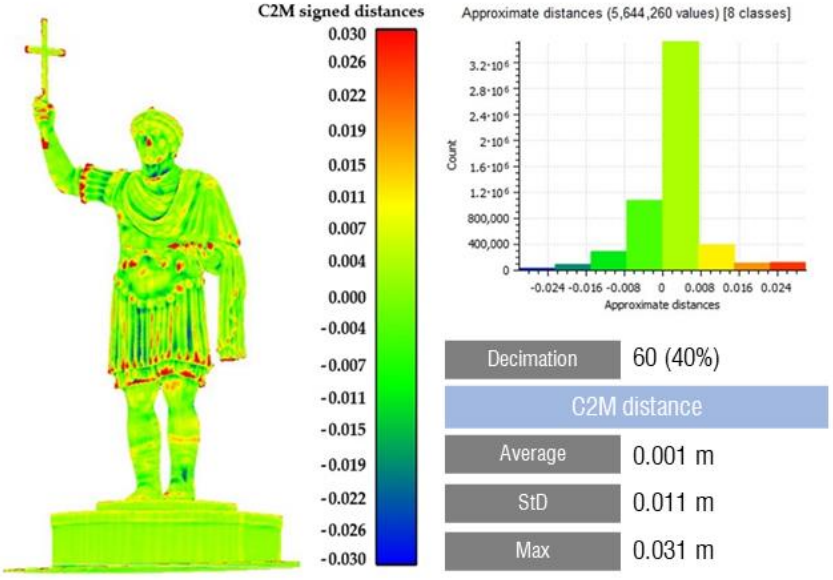
To overcome this problem, the mesh was decimated and transformed into a quad mesh, i.e. a polygonal square mesh. Using the "QuadRemesh" command, it was possible to quickly create a quad mesh with optimised topology from existing surfaces. Furthermore, with this tool it is also possible to define the number of faces for the re-meshing process.

By setting a minimum value, the result will be a mesh with a minimum number of polygons of uniform size. Conversely, by setting a high value, the mesh detail will be more accurate, and the polygon sizes will be smaller in areas of high curvature. The values of the "Adaptive Size" tool range from 0 to 100. A value of 0 means constructing a mesh that is as square as possible, with a uniform density of faces but with the risk of obtaining an irregular topology; conversely, a value of 100 means obtaining shapes of faces that vary in height and width and with a very high density of faces.

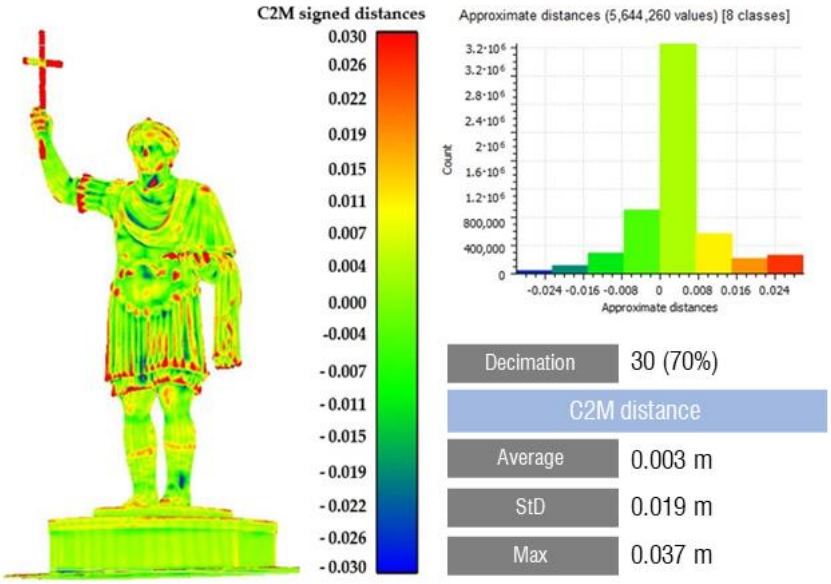
The result of this decimation and transformation of the mesh brings with it an intrinsic change in the 3D model. Indeed, with these processes, the risk of decreasing the resolution or accuracy of certain construction details is high. The morphological conformation of the structure to be analysed also makes it difficult to identify any portions or details of the statue that are lost as a result of the decimation of the mesh; the choice of the parameters to be used in this phase is important and must be made carefully. To identify the best degree of decimation and the best density of polygonal faces, several 3D models were produced, each with a different degree of decimation, and a Cloud to Mesh - C2M (cfr. 6.2.2) comparison was carried out in the Cloud Compare environment. In this way, it was possible to assess which elements could possibly be lost in the mesh simplification process and, depending on the set parameters, choose the best degree of decimation, obviously contextualised to the case study. As a result of the C2M comparison, the main mean, standard deviation, and maximum value values were calculated, which then made it possible to identify the best degree of decimation contextualised to the statue analysed. The following images summarise the C2M comparison performed and schematise the results obtained for a decimation degree of 90 (Figures 6.8a), 60 (Figures 6.8b) and 30 (Figures 6.8c).


C2M signed distances
0.030
0.026
0.022
0.019
0.015
0.011
0.007
0.004
0.000
-0.004
-0.007
-0.011
-0.015
-0.019
-0.022
-0.026
-0.030
Approximate distances (5,644,260 values) [8 classes]
Count
Approximate distances
Decimation
90 (10%)
C2M distance
Average
0.001 m
StD
0.009 m
Max
0.029 m

(a)


C2M signed distances
0.030
0.026
0.022
0.019
0.015
0.011
0.007
0.004
0.000
-0.004
-0.007
-0.011
-0.015
-0.019
-0.022
-0.026
-0.030
Approximate distances (5,644,260 values) [8 classes]
Count
Approximate distances
Decimation
60 (40%)
C2M distance
Average
0.001 m
StD
0.011 m
Max
0.031 m

(b)



(c)

Fig. 6.8 - Mean, standard deviation, and maximum value values, obtained from C2M comparison at different decimation rates: Decimation 90 (a), 60 (b), 70 (c).

In the case study of the statue, a value of 90 was applied, i.e. reducing and adjusting the quadruple mesh by 10%.

Analysing the results obtained, it can be seen that the model decimated to 10% shows an average deviation value, compared to the original model, of 0.001 m. In fact, in some places it can be seen that the value of the distance between the dense cloud and the mesh model is zero, i.e. it remains outside the search cylinder. By increasing the degree of decimation of the mesh model, the mean distance between the point cloud and the mesh increases; in particular, in the model reduced by 70%, a mean distance value of 0.003 m and a standard deviation value of approximately 0.02 are obtained. In the latter case, the qualitative analysis of the image shows that certain details and construction details are eliminated from the mesh. This additional degree of decimation, if chosen, could lead to misleading results as many details of the statue would not be part of the modelling process and would therefore negatively influence

the FEM analysis. Subsequently, the decimated mesh was optimised by means of commands capable of closing holes, repairing intersecting meshes and closing holes, repairing intersecting meshes and eliminating non-manifold elements in order to reduce possible errors in subsequent calculation phases.

Once these operations were completed, the entire decimated model was transformed into a NURBS surface, i.e. a representation of a mathematical surface capable of precisely defining any shape.

According (34) of a NURBS surface, recursive formulas for calculating the derivative of order k of B-spline basis functions of degree p are given as Hu et al., 2023:

$$N_{i,p}^{(k)}(u) = \frac{p}{u_{i+p} - u_i} N_{i,p-1}^{(k-1)}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}^{(k-1)}(u) \tag{45}$$

where the first and second derivative can be calculated as:

$$C^{(1)}(u) = \frac{A^{(1)}(u) - B^{(1)}(u)C(u)}{B(u)} \tag{46}$$

$$C^{(2)}(u) = \frac{A^{(2)}(u) - 2C^{(1)}(u)B^{(1)}(u) - C(u)B^{(2)}(u)}{B(u)} \tag{47}$$

where:

$$A^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p}^{(k)}(u) w_i P_i \tag{48}$$

$$B^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p}^{(k)}(u) w_i \quad \text{and} \quad k = 1,2 \tag{49}$$

It is also possible to define the radius of curvature (r(u)) and the function expressing the arc length of the NURBS curve (s(u)) as:

$$\rho(u) = \frac{\left\| C^{(1)}(u) \right\|^3}{\left\| C^{(1)}(u) \cdot C^{(2)}(u) \right\|} \tag{50}$$

$$s(u) = \int_b^a \left\| C^{(1)}(u) \right\| du \tag{51}$$

Since there is no analytical expression between $s(u)$ and u, the adaptive quadrature method based on Simpson's rule (Heng et al., 2010) is applied to calculate the arc length.

The first step is to approximate the length of the arc according to the following relationship:

$$l(a,b) = \frac{b-a}{6}\left(\left\| C^{(1)}(a) \right\| + \left\| C^{(1)}\left(\frac{a+b}{2}\right) \right\| + C^{(1)}(b) \right) \tag{52}$$

Thus, the parameter interval $[a,b]$ is divided into two intervals of equal size, $[a_1,b_1]$ e $[a_2,b_2]$, and Simpson's rule is applied again to obtain $l(a_1,b_1)$ e $l(a_2,b_2)$.

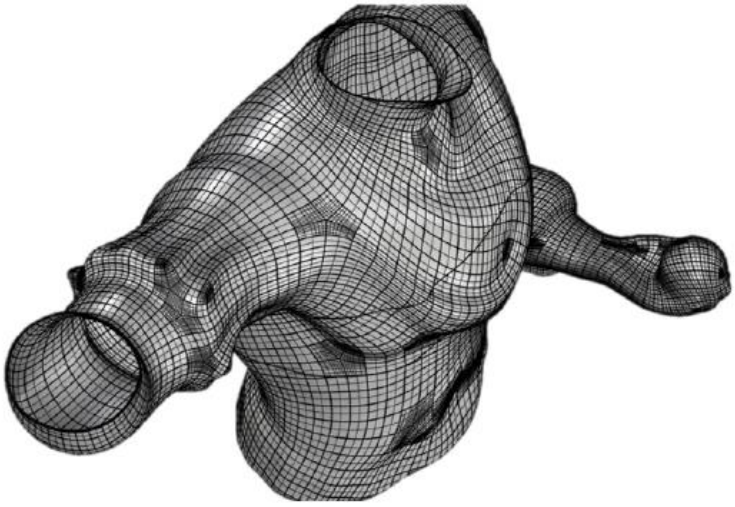
Given a specified tolerance εif the fowling condition is fulfilled, i.e:

$$|l(a,b) - l(a_1,b_1) - l(a_2,b_2)|/10 \leq \varepsilon \tag{53}$$

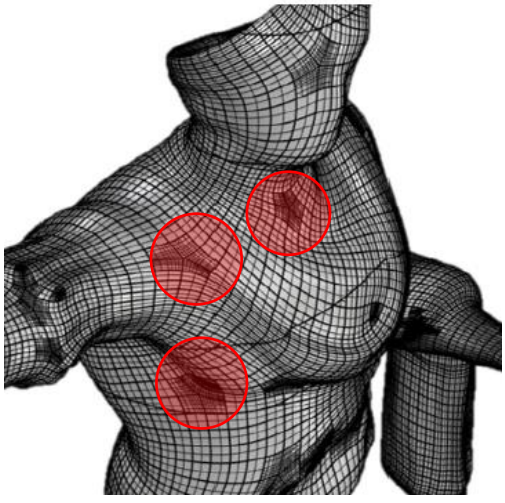
then $l(a,b)$, with the given tolerance, represents the true arc length. Otherwise, the subintervals are divided reiteratively into two equal intervals, with the tolerance value halved at each step, and Simpson's rule is applied to each subinterval. This process is repeated until all subintervals meet the tolerance, and the total length of the arc is obtained by summing the lengths of all subintervals.

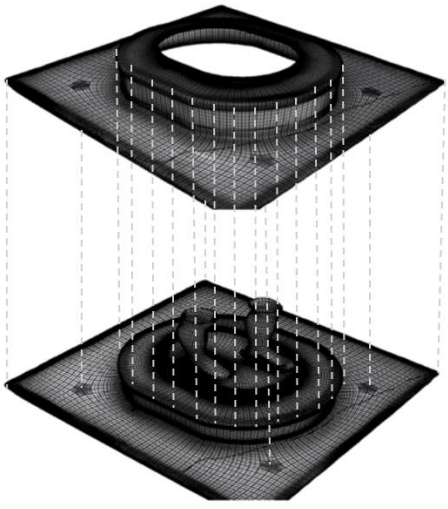
In parametric modelling, the objects of the statue (cloth, cross, sphere, etc.) were modelled as individual objects and then assembled.

For the FEM analysis, a series of parameters related to the material composition were assigned to each object. In particular, the different parameters that contribute to the structural analysis of the statue, such as Poisson's coefficient, modulus of elasticity, density, etc., were taken into account. In the following images, it is possible to visualise some details of the structure, such as the bust (Figure 6.9a), the bust and the insertion of the face and drape (Figure 6.9b), the plinth and the subsequent insertion of the boots (Figure 6.9c).



(a)





(b) (c)

Fig. 6.9 - Visualisation of some structure details in the NURBS modelling process. The elements highlighted in red, e.g. in figure 9b, represent the closing of holes or correction of the mesh in Rhino.

6.3.3. Repeatability of the decimation process in the field of CH

The proposed methodology made it possible to transform a 3D model (obtained by a photogrammetric process) into a parametric model optimised for FEM analysis. In particular, the photogrammetric survey made it possible to obtain a three-dimensional model quickly and accurately, reconstructing every detail of the structure under consideration. In fact, the 3D model showed very low RMSE values, of the order of millimetres. The steps that led to the definition of this process were:

1. creation of the TIN model;
2. reducing the number of faces of the TIN model;
3. construction of the quad mesh;
4. decimation of the quad mesh model;
5. repair of the model from any errors on the surface mesh;
6. transformation into NURBS surfaces.

With this methodological approach, it was possible to obtain a three-dimensional model as faithful as possible to the original structure, while retaining its accuracy and geometric detail. In fact, a comparison between the point cloud generated by the photogrammetric process (reference) and the square mesh reduced by 10% shows that 85% of the points differ by approximately 3 mm. Furthermore, as the level of decimation increases, it can be seen that some of the finer and more complex construction details, such as the cross or the folds of the cloth and cloak, deviate more from the original model. In order to verify the quality and repeatability of the method developed on different types of statues, additional datasets available on dedicated platforms were taken into consideration; in particular, 4 datasets were obtained from the website of the Geometric Computing Laboratory, École Polytechnique Fédérale de Lausanne, School of Computer and Communication Sciences, Switzerland (https://lgg.epfl.ch/). A further dataset, which concerns a point cloud obtained from a photogrammetric approach using images generated by a UAV platform, was generated by Furaysolutions Company and downloaded from Sketchfab (*https://sketchfab.com/3d-models/kossuth-statue-point-cloud-29d3f031a9ce4cd095b1ee7c3fbe9391*). Table 6.2 below describes and schematises the analysed datasets.

Tab. 6.2 - Datasets used for method repeatability verification.

Title	Dataset	Height [m]	Figure
Headmask	Image	2.00	
Aquarius	Image	1.50	
Kossuth	Image	6.00	
Paderwski	Point Cloud	3.00	
Taichi	Image	3.70	

As discussed in the previous paragraphs, the different datasets (number of images, dense cloud, number of meshes) were compared with their respective point clouds; in particular, Table 6.3 below summarises the characteristics of the compared datasets, in terms of TIN, original quad-mesh, quad-mesh decimated at 30 % and 70 % from the initial point cloud.

Tab. 6.3 - Point Cloud and polygon mesh dimensional characteristics of the compared datasets.

Dataset	Point Cloud	MESH			
		TIN	QUAD (100%)	QUAD (70%)	QUAD (30%)
AQUARIUS	835552	212142	119360	91624	33982
HEADMASK	150226	101958	100818	69437	33841
KOSSUTH	2369663	149816	107697	73197	34035
PADERWSKI	625404	206444	150185	105403	44484
TAICHI	864808	204635	141968	101955	46887

For each individual dataset, a comparison was performed using the C2M algorithm and the results were evaluated in terms of Average Distance (AVG Distance), StD (Standard Deviation) and Maximum Error (Max Error). The results of this comparison are summarised in the following tables (Tables 6.4-6.8).

Tab. 6.4 - Result of C2M algorithm "Aquarius" dataset.

C2M - AQUARIUS DATASET				
VALUE	**TIN**	**QUAD (100%)**	**QUAD (70%)**	**QUAD (30%)**
AVG DISTANCE	0.000	0.006	0.006	0.006
StD	0.002	0.013	0.013	0.013
MAX ERROR	0.016	0.015	0.016	0.016

Tab. 6.5 - Result of C2M algorithm "Headmask" dataset.

C2M - DATASET HEADMASK				
VALUE	**TIN**	**QUAD (100%)**	**QUAD (70%)**	**QUAD (30%)**
AVG DISTANCE	0.000	0.001	0.001	0.002
StD	0.009	0.017	0.019	0.020
MAX ERROR	0.024	0.024	0.025	0.025

Tab. 6.6 - Result of C2M algorithm "Kossuth" dataset.

C2M - KOSSUTH DATASET				
VALUE	**TIN**	**QUAD (100%)**	**QUAD (70%)**	**QUAD (30%)**
AVG DISTANCE	0.010	0.011	0.012	0.011
StD	0.019	0.020	0.020	0.020
MAX ERROR	0.022	0.021	0.021	0.022

Tab. 6.7 - Result of C2M algorithm "Paderwski" dataset.

C2M - PADERWSKI DATASET				
VALUE	**TIN**	**QUAD (100%)**	**QUAD (70%)**	**QUAD (30%)**
AVG DISTANCE	0.002	0.003	0.004	0.004
StD	0.012	0.014	0.014	0.015
MAX ERROR	0.029	0.029	0.029	0.029

Tab. 6.8 - Result of C2M algorithm "Taichi" dataset.

C2M - TAICHI DATASET				
VALUE	**TIN**	**QUAD (100%)**	**QUAD (70%)**	**QUAD (30%)**
AVG DISTANCE	0.002	0.002	0.002	0.016
StD	0.010	0.011	0.011	0.092
MAX ERROR	0.033	0.033	0.033	0.033

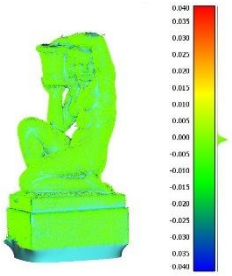
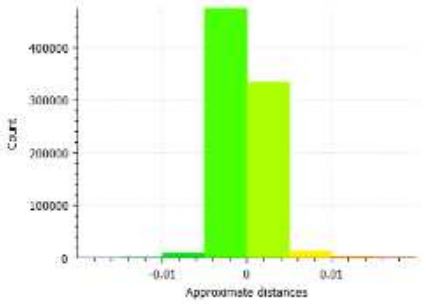
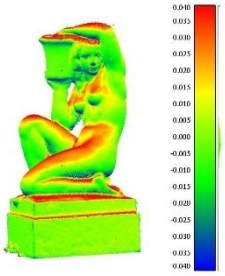
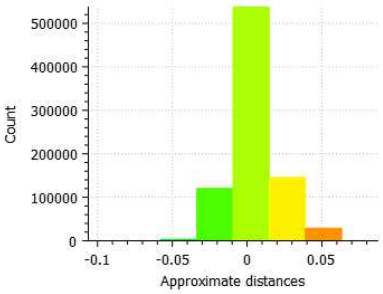
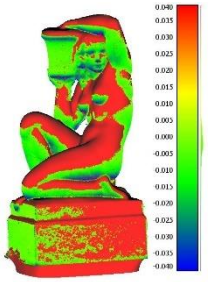
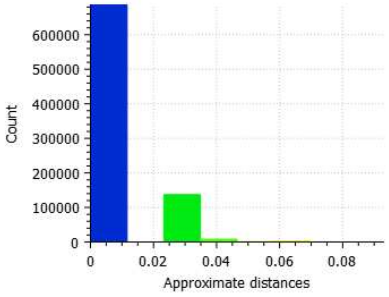
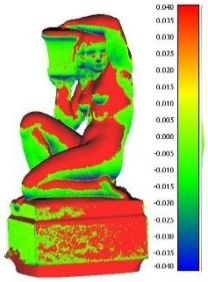
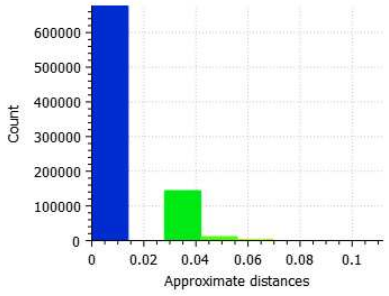
From the analysis of these, it is possible to note that the reduction from the TIN model to the polygonal quad mesh model is optimal as it produces a surface model with a smaller number of faces, thus more easily manageable, while maintaining accuracy. This process does not produce comparable results as it is dependent on the size, geometry and complexity of the model analysed.

Starting with the quad mesh model, a series of decimations of the model were progressively carried out with a reduction percentage of 30% and 70%. To decimate the model, the different parameters were then set according to the desired decimation percentage. In particular, the number of targets faces for the re-mesh operation (input to the algorithm used by the software) and a minimum number of quadrangles and uniform dimensions to be obtained were defined a priori. The value for the minimum number of quadrangles, which is controlled by the user, allows for the possibility of obtaining a smaller number of quadrangles; this value is important as it defines the degree of approximation in areas subject to curvature. The lower the value, the less control there will be over the quadrangles, which will, however, be of a similar size and with uniform density; conversely, higher values (max. 100%) allow a higher level of detail to be maintained.

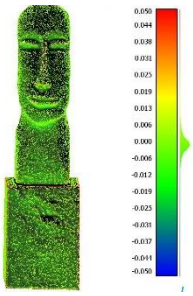
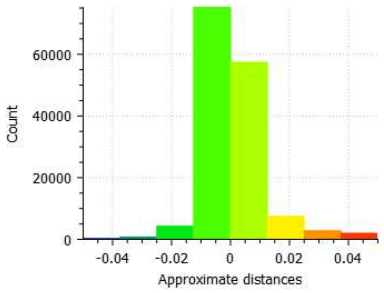
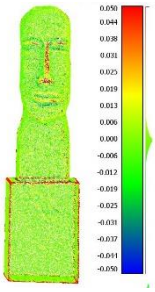
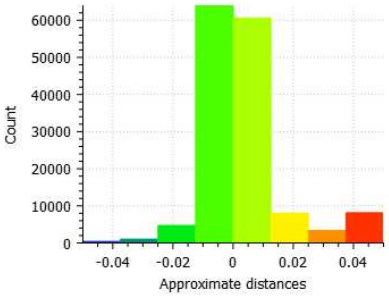
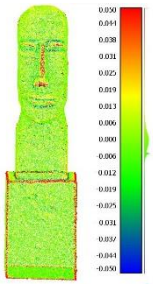
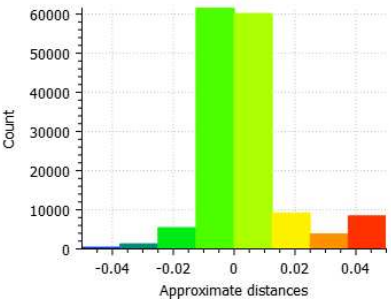
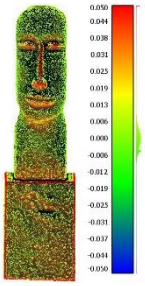
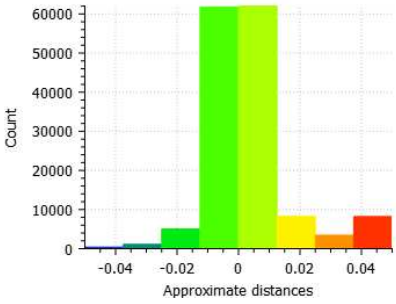
In the first case (low values below 30%) there will be a considerable reduction in the number of faces with the possibility of irregularities in the mesh type, while in the second case the accuracy and resolution in the reconstruction of details will be higher, resulting in the generation of a greater number of faces. In the decimation process tested, it was decided to decimate the number of quadrangles making up the mesh, while maintaining the last input value corresponding to 100%, in order to still maintain a high degree of detail in the resulting mesh.

In table below (Tables 6.9-6.13), for each individual dataset analysed, using the point cloud as a reference, the results obtained from the C2M comparison are represented by means of a graduated scale bar and the corresponding histogram.

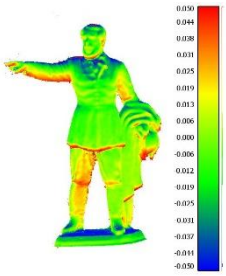
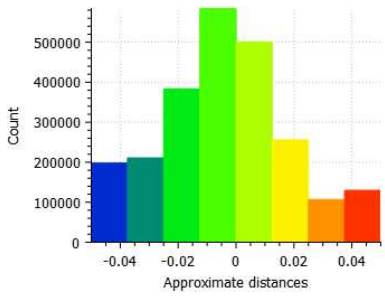
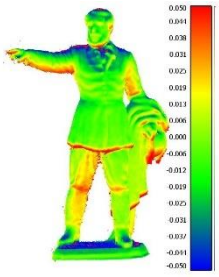
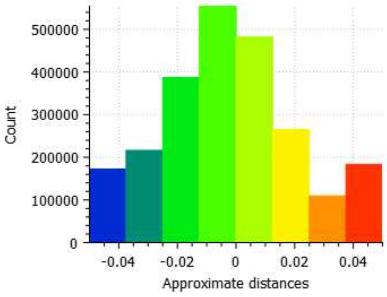
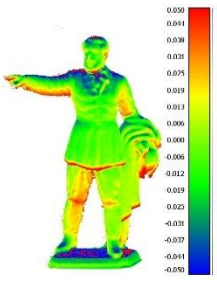
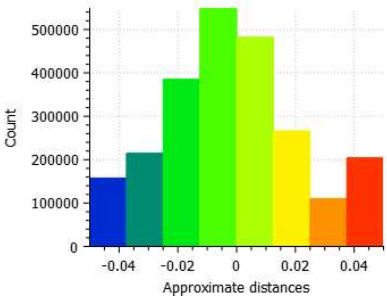
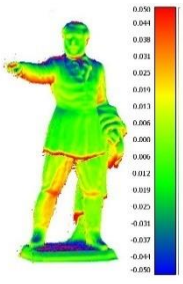
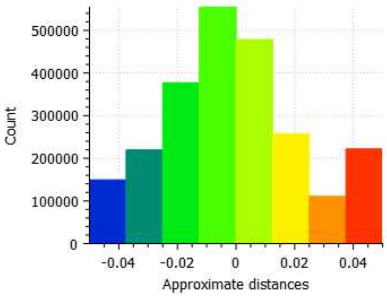
Tab. 6.9 - C2M algorithm results for the “Aquarius” dataset.

Mesh Type	C2M Model	Hystogram
TIN		
QuadMesh (No decimation)		
QuadMesh (30% decimation)		
QuadMesh (70% decimation)		

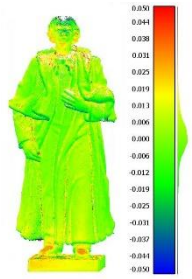
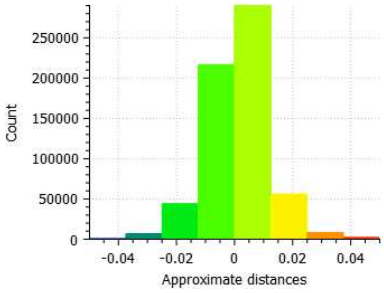
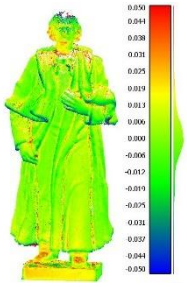
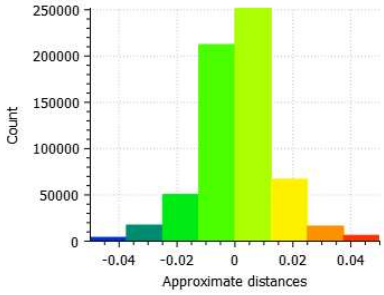
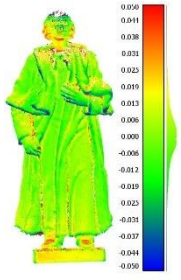
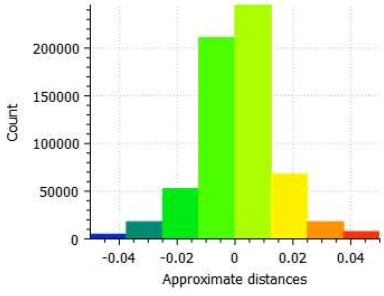
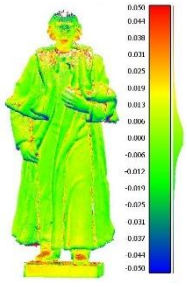
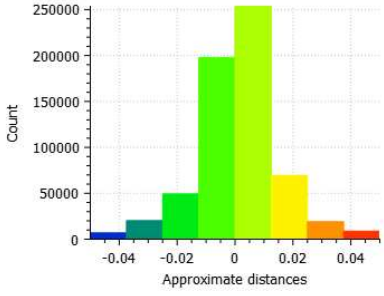
Tab. 6.9 - C2M algorithm results for the “Headmask” dataset.

Mesh Type	C2M Model	Hystogram
TIN		
QuadMesh (No decimation)		
QuadMesh (30% decimation)		
QuadMesh (70% decimation)		

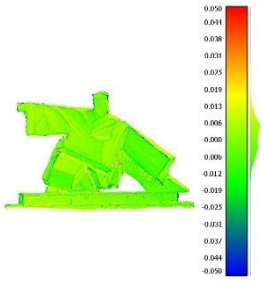
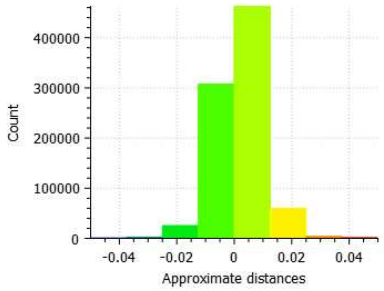
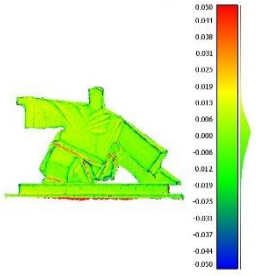
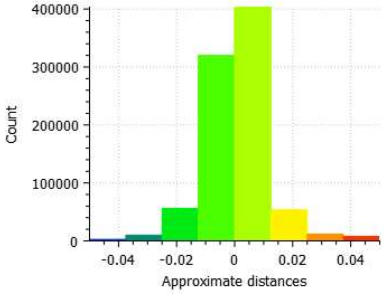
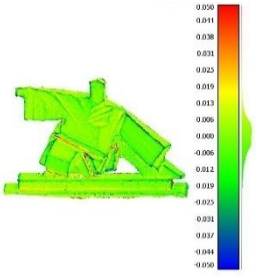
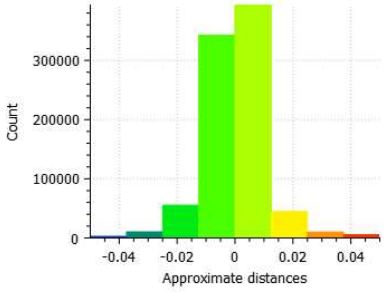
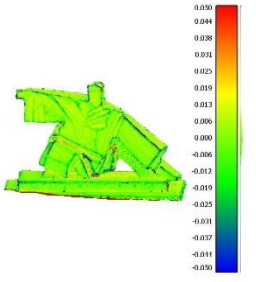
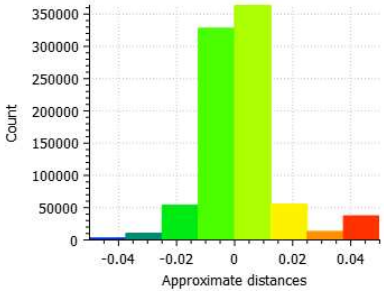
Tab. 6.10 - C2M algorithm results for the “Kossuth” dataset.

Mesh Type	C2M Model	Hystogram
TIN		
QuadMesh (No decimation)		
QuadMesh (30% decimation)		
QuadMesh (70% decimation)		

Tab. 6.11 - C2M algorithm results for the “Paderwski” dataset.

Mesh Type	C2M Model	Hystogram
TIN		
QuadMesh (No decimation)		
QuadMesh (30% decimation)		
QuadMesh (70% decimation)		

Tab. 6.12 - C2M algorithm results for the “Taichi” dataset.

Mesh Type	C2M Model	Hystogram
TIN		
QuadMesh (No decimation)		
QuadMesh (30% decimation)		
QuadMesh (70% decimation)		

A further evaluation conducted on the previous analysed datasets concerns the export format and, consequently, the correct import and management of meshes within the FEM solver.

In general, the choice of the most appropriate file export format must consider not only the characteristics and information exported, but also the size of the resulting file. In fact, large files are often not manageable by all programmes, as they require high computational capacities and, consequently, do not meet the requirements of optimal model management. The choice of the best available format for correctly exporting the generated model and thus realising an optimal process between the hardware components of a computer (processor, RAM, HDD) and the programmes selected for analysis is therefore obvious.

From a modelling point of view, the geometric modelling kernel is a software component that adds 3D modelling functions to CAD software. It enables the design of a virtual model that supports the simulation and production of real objects and the representation of these objects using various types of geometric representations (faceted models, meshes of triangular surfaces, and B-rep models, which use mathematically defined surfaces to precisely delineate a solid model).

In this case study, two different types of formats were analysed, namely the Parasolid format and the Step format.

The Parasolid format is a geometric kernel originally developed by Shape Data Limited and now owned by the Siemens PLM Software group (UGS Corp.). It is used by various computer graphics software platforms, particularly in Computer-aided design (CAD), Computer-aided manufacturing (CAM), Computer-aided engineering (CAE), Product visualisation, and CAD data exchange. This format includes Boolean functions, surface modelling, and other CAD features, as well as being highly performant in mechanical CAD software in particular. It also provides Convergent Modeling technology, which extends its robust B-rep modelling functions to models based on face representations. Files are saved in *.x_t.* or binary *.x_b*. format.

As for the STEP (Standard for the Exchange of Product Data) file format, this extension is associated with 3D models and objects and is compatible with various CAD design software.

A STEP file contains text strings in ASCII format and is able to describe objects, scenes and 3D models using the STEP format specification of the ISO 10303 file.

ISO 10303 is a standard containing a set of rules for data integration, presentation and exchange and is widely used for data transfer between systems: CAD, CAM, CAE, PDM/EDM. Using this standard provides an unambiguous description that can be adapted to all computer systems.

In general, a STEP file is typically divided into HEADER and DATA sections with commands and subsections delimited by semicolons and newlines. The HEADER section of a STEP file contains metadata, while the DATA section contains entity names, attribute and datatype mappings, references, etc. The files are saved in *.step* format.

There is no substantial difference in saving the model in one of the two formats but saving in Parasolid format may be the best solution if the FEM analysis software is capable of handling this format; in this case the risk of some geometries being lost in the saving process is reduced. However, this format has the disadvantage, in the presence of numerous meshes, of being very heavy in terms of storage capacity and file management, and there may be a need to update the final model to a more or less recent version of the Parasolid kernel.

The STEP export format is much more universal, with the risk, however, that some geometries are not translated and consequently not stored correctly or suitable for the final purpose.

The choice of one or the other format, therefore, must be based on several factors (file size, model geometry, export time, correct implementation in other software, etc.); for this purpose, an evaluation was carried out in terms of memory and storage time with respect to the possible export in the two formats (Tables 6.14-6.17).

Tab. 6.13 - Valuation of export time and storage memory - TIN model.

DATASET	POINT CLOUD	NURBS BY TIN	TIME		MEMORY	
			PARASOLID	STEP	PARASOLID	STEP
ACQUARIUS	835552	212058	>30	3 MIN	NO	250 MB
HEADMASK	150226	101958	10 MIN	1 MIN	245 MB	125 MB
KOSSUTH	2369663	149816	>30	2 MIN	NO	180 MB
PADEREVSKY	625416	206262	>30	4 MIN	NO	257 MB
TAICHI	864808	204541	>30	4 MIN	NO	254 MB

Tab. 6.14 -Valuation of export time and storage memory - QuadMesh model.

DATASET	POINT CLOUD	NURBS BY QUAD (100%)	TIME		MEMORY	
			PARASOLID	STEP	PARASOLID	STEP
ACQUARIUS	835552	114999	>30	1 MIN	NO	187 MB
HEADMASK	150226	100818	5 MIN	< 1 MIN	337 MB	165 MB
KOSSUTH	2369663	107697	>30	1 MIN	NO	177 MB
PADEREVSKY	625416	143638	>30	2 MIN	NO	237 MB
YAICHI	864808	138415	>30	1.5 MIN	NO	227 MB

Tab. 6.15 - Valuation of export time and storage memory - QuadMesh model 30%.

DATASET	POINT CLOUD	NURBS BY QUAD (70%)	TIME		MEMORY	
			PARASOLID	STEP	PARASOLID	STEP
ACQUARIUS	835552	114999	12 MIN	<1 MIN	291 MB	142 MB
HEADMASK	150226	69437	8 MIN	<1 MIN	230 MB	112 MB
KOSSUTH	2369663	73197	10 MIN	<1 MIN	242 MB	119 MB
PADEREVSKY	625416	98496	16 MIN	<1 MIN	329 MB	162 MB
YAICHI	864808	98402	16 MIN	<1 MIN	328 MB	161 MB

Tab. 6.16 - Valuation of export time and storage memory - QuadMesh model 70%.

DATASET	POINT CLOUD	NURBS BY QUAD (30%)	TIME		MEMORY	
			PARASOLID	STEP	PARASOLID	STEP
ACQUARIUS	835552	29749	1 MIN	< 10 SEC	96 MB	47 MB
HEADMASK	150226	33841	2 MIN	< 10 SEC	110 MB	54 MB
KOSSUTH	2369663	34035	2 MIN	< 10 SEC	110 MB	55 MB
PADEREVSKY	625416	38043	2.5 MIN	< 10 SEC	125 MB	61 MB
YAICHI	864808	43334	3 MIN	< 10 SEC	142 MB	69 MB

From the analysis of the tables, it is possible to see how the process of decimation and simplification of the initial mesh model also influences the final export process to external software. In fact, it can be seen how, by simplifying the model, not only does the file size decrease, but also the file saving times become progressively shorter. In the case of the TIN model and the NURBS model obtained from QuadMesh, for all datasets (except one), it was not possible to proceed with the saving of the model in Parasolid format: the saving times exceeded 30 minutes, resulting in a software crash and a total lack of data. On the contrary, with however long times, it was possible to export the same model in STEP format, showing in the only case that could be analysed an occupancy rate in terms of file size reduced by around 50%. This value is confirmed when analysing the results obtained with the subsequent reduced models (reduction to 30% and 70%), which also show that by decimating and simplifying the model, it is also possible to export the file in Parasolid format (average time of 12 min. in the case of a 30% mesh reduction and 2 min. in the case of a 70% mesh reduction).

6.4. Implementation in FEM software

In order to characterise the structural behaviour of the statuary element, a FEM analysis simulation was conducted using Midas FEA NX software (MIDASoft, Inc., New York, USA, v. 1.1).

The choice of this software is based on the fact that it has advanced modelling tools and instruments and that, thanks to its compatibility with other formats, high performance can be achieved in a short time. In addition, the possibility of managing and solving numerous meshes allows the finite element model to represent and solve the previously created solid geometry more accurately.

The following pipeline (Figure 6.10) shows a classic operational workflow contextualised to the case study under consideration which, starting from the NURBS model obtained, allows the study and structural analysis of the 3D model.

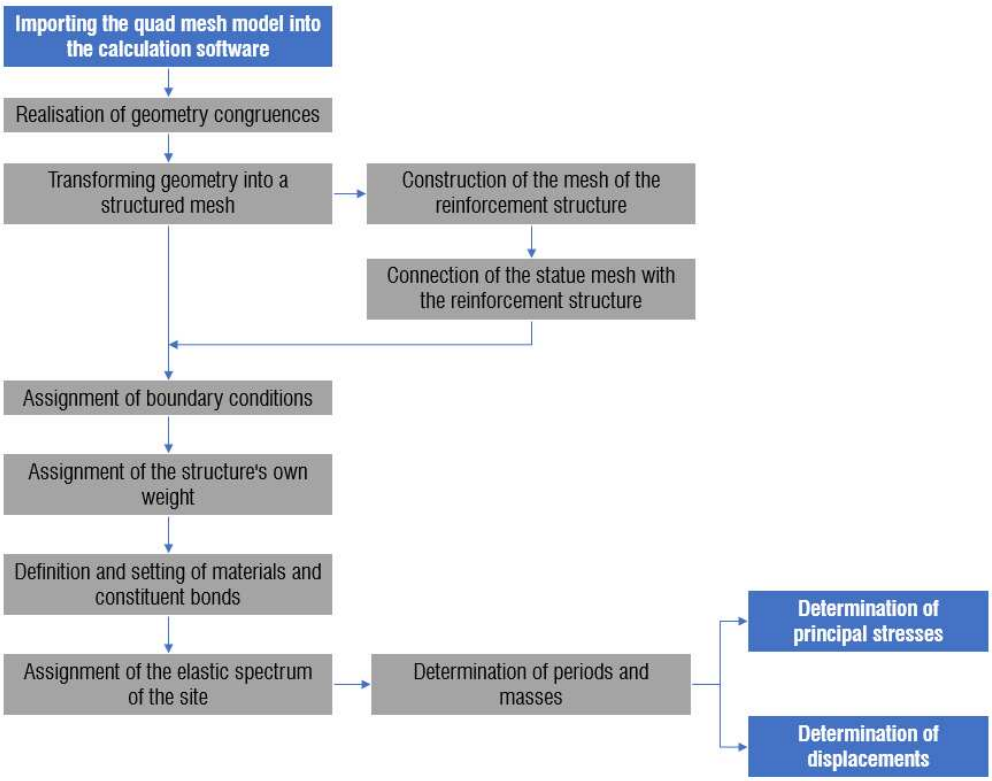


Fig. 6.10 - Workflow used for FEM analysis in the two different scenarios.

6.4.1. FEM modelling of the unreinforced model

Once the geometry constructed using NURBS surfaces had been processed (Figure 6.11a), the model was imported into the Midas FEA NX software; at this stage, some problems were encountered due to the failure to recognise all the surfaces making up the geometry. To remedy this problem, the parametric model was sectioned and subdivided into different portions and corrected. In this way, the software was able to correctly recognise all the geometries of the statue; by means of Boolean commands, the geometries were reconstructed and made congruent in order to obtain a single model. From the reconstructed geometry, the structural mesh was then created (Figure 6.11b), where each colour represents an individual portion of the model. The model

was perfectly reassembled, and boundary conditions were defined and set, such as the creation of an interlocking constraint at the base of the structure (Figure 6.11c).

Fig. 6.11 - Parametric 3D model for FEM analysis: NURBS processed in Rhino software (a), structural mesh (b), base constraint condition (c).

6.4.2. Modelling of the reinforced present in the statue

About the internal support of the statue, it was necessary to model the steel structures that could not be detected with a photogrammetric survey. In fact, to increase the stability of the statue, steel reinforcements were inserted into the interior during the last phase of the restoration. This steel reticular structure is connected to the statue through bolted joints. The 2D drawings and construction details of this reinforcement structure were available in non-vector format. For this reason, it was necessary to reconstruct the geometry of the internal reinforcement structures by means of 2D drawings in CAD format and realised during one of the restoration phases. The 3D reconstruction of the internal structure was modelled with Rhino software; in addition, all

structures were assembled within the same software and subsequently imported into the FEM environment.

For this purpose, a further analysis was conducted on the same model, incorporating the steel structure inside the statue (Figure 6.12).

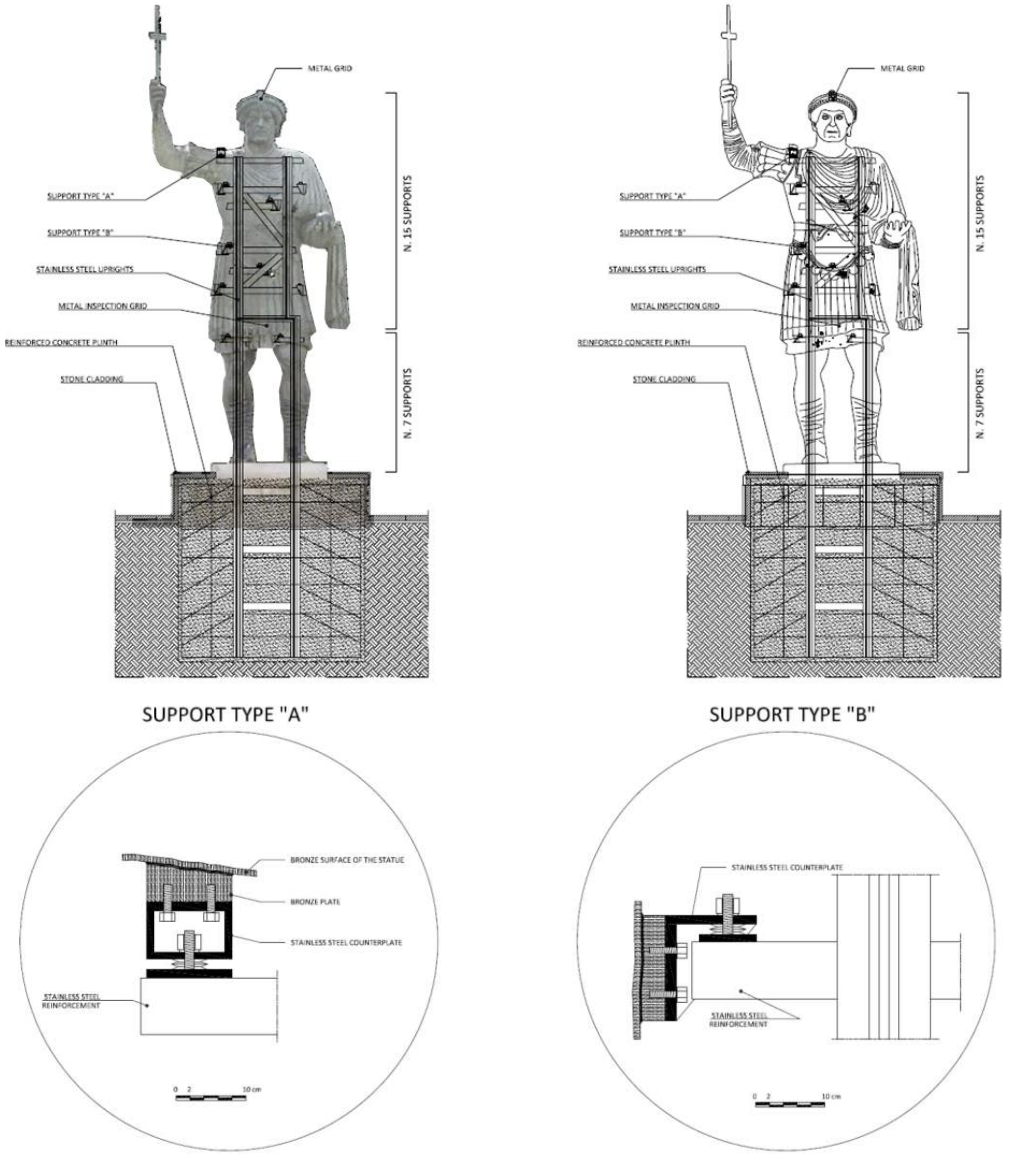


Fig. 6.12 - Construction details of the internal steel reinforcement structure and structural nodes.

For the elaboration of the FEM analysis with reinforcement, it was necessary to first model the entire steel structure in Rhino software. For this modelling phase, the design drawings used in one of the last restoration phases were useful for the three-dimensional reconstruction of the structure. The realisation of the mesh of the reinforcement structure and its exact positioning inside the statue positioning inside the statue was not an easy task as the steel structure had several areas of overlapping steel plates, i.e. the structural nodes.

To solve this problem, the mesh was carefully calibrated and scaled at the nodes. In addition, a further connection-related difficulty was addressed: in order to create a connection between the steel structure and the statue, the structural parts to be connected were extruded with Boolean commands and intersected the surface of the statue by 0.004 m. Subsequently, the parts exceeding the statue were cut off. The congruence between the structure and the statue was achieved through sophisticated contact modelling.

Once the mesh of the reinforcement structure had been created, it was correctly oriented and positioned within the statue to be subsequently imported into the calculation software; through Boolean commands, all the different parts of the reinforcement structure were connected as geometries (Figure 6.13).

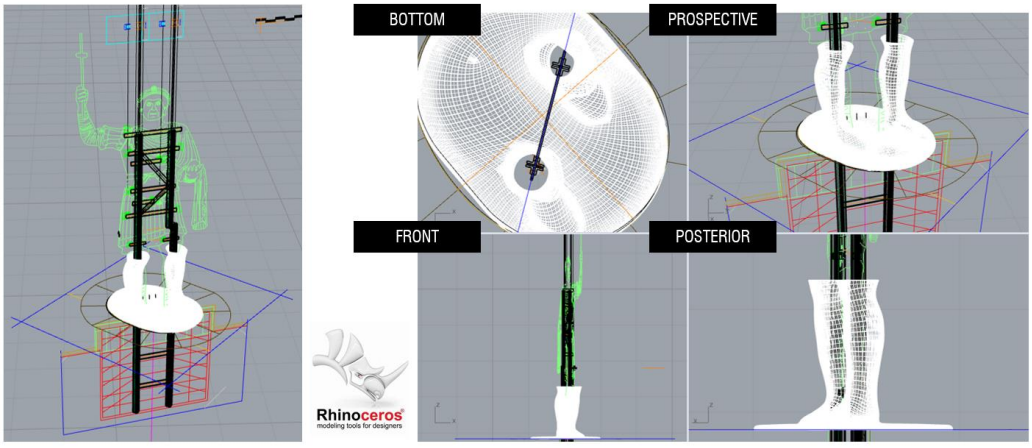


Fig. 6.13 - Modelling and positioning of steel reinforcement in Rhinoceros software.

After having correctly positioned the mesh of the steel reinforcement within the mesh of the previously realised statue, it was possible to import the parametric model thus elaborated into the calculation software for the subsequent transformation into a structural mesh. In particular, the modelling of the mesh of the steel parts constituting the internal structure is shown in Figure 6.14. Each colour refers to the creation of a mesh of a distinct geometric element. In addition, the mesh can be detailed differently with respect to the type of element in order to clearly visualise the stresses and displacements on the structure. Figure 6.14 also shows the results obtained in the realisation of the connections between the bronze structure, the statue body and the steel structure. For the realisation of these connections with the statue mesh, both classical Boolean operations and contact theory were used.

This last strategy is frequently used in mechanical engineering for the realisation of precise mechanical unions, bolts, steel knots, welds, etc. Once the entire structure had been realised, the boundary conditions and forces in play were recalibrated and, using the 'size control' command, mesh infills were created at the most delicate elements or areas.

Fig. 6.14 - Modelling the connections between the bronze structure, the body and the steel structure.

CONCLUSIONS

The aim of this study was to identify a methodology for processing the point cloud obtained from photogrammetric or TLS surveys, semi-automatically and with controlled metric rigour, in order to generate high-performance digital 3D models in the BIM/FEM framework.

Starting from a triangular surface, the objective was to transform the model into a quadrangular polygonal model (quad-mesh) in order to optimise and correct all topological and geometrical errors in the resulting mesh. In this step, a very important aspect was the metric rigour with which the entire methodological process was developed and the control of the accuracies, the latter depending on the final purpose of the modelling.

The transformation of this model into a NURBS has the advantage of defining, through a precise mathematical representation, the 3D geometry of the object represented; therefore, thanks also to the accuracy and reliability of NURBS geometries, the methodological approach analysed can be used in a variety of processes and requires far less information than is necessary to represent the same geometry through mesh approximations.

Furthermore, the mathematical representation of each geometry also provides the opportunity to carry out an eventual multi-temporal comparison or monitoring of the analysed structure. In this way, it is possible over time to implement the model only for the portions that are necessary for monitoring, thus also reducing the time necessary for the subsequent new survey and data processing phases.

Through this method of mesh generation, processing and management, the resulted three-dimensional model will be ready and performant for implementation in BIM software and platforms and will be totally parameterised; in fact, for each object all the semantic, material, and structural information can be assigned for its complete classification. In the same way, if correctly implemented with the required information, it will be able to be imported and analysed using appropriate FEM performance solvers.

Thanks to this innovative modelling approach and the tests carried out on two different architectural typologies, a specific and shared operating procedure has been identified; in this way the users can be able to realise high-performance 3D models especially in the field of Cultural Heritage.

The described workflow is, moreover, adaptable in any context and offers an adequate control tool on the final three-dimensional processing.

A further advantage is the simplification and decimation process of the model in order to make the final model easier to manage in terms of storage and computational capacities, interoperable in different software through appropriate data saving formats, and, very important, without the risk of some geometries being lost in the saving and exporting process.

The simplification and decimation of the model is a research to be developed and improved through new algorithms and solutions based on AI (Artificial Intelligence); in this context, the development of new geometric and topological indicators linked to mesh surfaces can make this approach fully automatic. In fact, by creating new automatic feature and geometry recognition algorithms, it will be possible to automate the mesh simplification and decimation process. This will lead in a very automatic way to classify the point cloud and identify and categorise all the complex geometries of the investigated model. In this way, all the quadrangular surfaces that will be generated in the 3D modelling process will be appropriately simplified and/or decimated according to the element to be represented. This research has already been performed, testing a semantic classification and segmentation methodology of the point cloud, through Deep Learning techniques; using this approach, it was possible to automatically recognise objects and architectural elements with a high degree of detail.

Furthermore, through statistical performance indices, it is possible to assess the accuracy of the method and apply this approach also on complex geometries. The advantage of this innovative framework will be to reduce the time and optimise the overall Scan to BIM or Scan to FEM process (in particular in the field of Cultural Heritage).

The final result will consist in a 3D digital model where the simplest geometries will be represented by standard geometric objects, and finer details and construction details will not lose resolution but will be reconstructed and represented with high metric and geometric rigour, modelled through polygonal surfaces and fully parameterised, as well as converted to NURBS and optimised for the subsequent processing steps.

ACKNOWLEDGEMENTS

At the end of this experience, my first thanks go to Prof. Domenica Costantino who has followed, assisted and shaped my academic progress since my Bachelor's degree, providing me with extremely patient and helpful advice and suggestions. To her goes my sincerest gratitude.

I would like to thank Prof. Massimiliano Pepe, my guide, support but above all my fraternal friend over these years; thank you for always motivating and inspiring me.

I would like to express my sincere appreciation and thanks to Prof. Vito Iacobellis, PhD Course Coordinator, always available and ready to help me.

I would like to thank Prof. Sorin Herban of the Politehnica of Timisoara, his immense availability and hospitality, as well as his expertise are skills to be envied.

I would like to thank Gabriele and Cristian, colleagues from the Geomatics Laboratory in Taranto, with whom I shared part of this experience.

I dedicate this achievement to Alessandra and my whole family because they are the people who have never stopped believing in me and who have always encouraged me to overcome all barriers. They are the ones who, over these three years and throughout my entire academic and professional career, have supported and endured all the moments of joy and difficulty. To them and to those who guide me from up there, I dedicate this other piece of my life.

Finally, thanks to all those with whom, in some way, I have shared part of this experience.

BIBLIOGRAPHY

Adamopoulos, E., & Rinaudo, F. (2021). Documenting the state of preservation of historical stone sculptures in three dimensions with digital tools. In Pattern Recognition. *ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021*, Proceedings, Part III (pp. 666-673). Springer International Publishing.

Alessandri, L., Baiocchi, V., Del Pizzo, S., Rolfo, M. F., & Troisi, S. (2019). Photogrammetric survey with fisheye lens for the characterization of the La Sassa cave. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 25-32.

Alfio, V. S., Costantino, D., & Pepe, M. (2020). Influence of Image TIFF Format and JPEG Compression Level in the Accuracy of the 3D Model and Quality of the Orthophoto in UAV Photogrammetry. *Journal of Imaging*, 6(5), 30.

Alliez, P., Cohen-Steiner, D., Yvinec, M., & Desbrun, M. (2005). Variational tetrahedral meshing. *In ACM SIGGRAPH 2005 Papers* (pp. 617-625).

Artese, S. (2019). The Survey of the San Francesco bridge by Santiago Calatrava in Cosenza, Italy. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 33-37.

Aulinas, J., Petillot, Y., Salvi, J., & Lladó, X. (2008). The SLAM problem: a survey. *Artificial Intelligence Research and Development*, 363-371.

Baik, A. (2021). The use of interactive virtual BIM to boost virtual tourism in heritage sites, historic Jeddah. *ISPRS International Journal of Geo-Information*, 10(9), 577.

Barazzetti, L., Banfi, F., Brumana, R., Gusmeroli, G., Oreni, D., Previtali, M., ... & Schiantarelli, G. (2015). BIM from laser clouds and finite element analysis: combining structural analysis and geometric complexity. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40, 345-350.

Barone, S., Paoli, A., & Razionale, A. V. (2012). 3D reconstruction and restoration monitoring of sculptural artworks by a multi-sensor framework. *Sensors*, 12(12), 16785-16801.

Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., & Taubin, G. (1999). The ball-pivoting algorithm for surface reconstruction. *IEEE transactions on visualization and computer graphics*, 5(4), 349-359.

Bertolini-Cestari, C., Chiabrando, F., Invernizzi, S., Marzi, T., & Spanò, A. (2013). Terrestrial laser scanning and settled techniques: A support to detect pathologies and safety conditions of timber structures. *Advanced materials research*, 778, 350-357.

Bianco, S., Ciocca, G., & Marelli, D. (2018). Evaluating the performance of structure from motion pipelines. *Journal of Imaging*, 4(8), 98.

Bici, M., Campana, F., Colacicchi, O., & D'Ercoli, G. (2018, June). CAD-CAE methods to support restoration and museum exhibition of bronze statues: the "Principe Ellenistico". *In IOP Conference Series: Materials Science and Engineering* (Vol. 364, No. 1, p. 012014). IOP Publishing.

Bitelli, G., Girelli, V. A., & Sammarini, G. (2018). 4-dimensional recording and visualization of urban archeological excavations. *Applied Geomatics*, 10(4), 415-426.

Bommes, D., Lévy, B., Pietroni, N., Puppo, E., Silva, C., Tarini, M., & Zorin, D. (2013, September). Quad- mesh generation and processing: A survey. *In Computer graphics forum* (Vol. 32, No. 6, pp. 51-76).

Borri, A., & Grazini, A. (2006). Diagnostic analysis of the lesions and stability of Michelangelo's David. *Journal of Cultural Heritage*, 7(4), 273-285.

Botsch, M., Pauly, M., Kobbelt, L., Alliez, P., Lévy, B., Bischoff, S., & Röossl, C. (2007). *Geometric modeling based on polygonal meshes*.

Brown, D.C.: Close-Range Camera Calibration, Photogrammetric Engineering, 37(8), 1971, 855-866

Brumana, R., Della Torre, S., Previtali, M., Barazzetti, L., Cantini, L., Oreni, D., & Banfi, F. (2018). Generative HBIM modelling to embody complexity (LOD, LOG, LOA, LOI): Surveying, preservation, site intervention—The Basilica di Collemaggio (L'Aquila). *Applied Geomatics*, 10, 545-567.

Buccolieri, G., Nassisi, V., Delle Side, D., Giuffreda, E., Marabelli, M., Buccolieri, A., ... & Castellano, A. (2015, November). The restoration of the Colosso di Barletta: EDXRF analysis. *In 4th Workshop-Plasmi, Sorgenti, Biofisica ed Applicazioni* (Vol. 2014, pp. 7-10).

Čekada, M. T., Crosilla, F., & Fras, M. K. (2010). Theoretical LiDAR point density for topographic mapping in the largest scales. *Geodetski vestnik*, 54(3).

Chen, C., & Tang, L. (2019). Development of BIM-based innovative workflow for architecture, engineering and construction projects in China. *International Journal of Engineering, Science and Technology*.

Chen, L. (2004, September). Mesh Smoothing Schemes Based on Optimal Delaunay Triangulations. *In IMR* (pp. 109-120)

Coli, M., Ciuffreda, A. L., & Micheloni, M. (2019). An informative content 3D model for the hall holding the resurrection of Christ by Piero della Francesca mural painting at Sansepolcro, Italy. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 435-442.

Costantino, D., & Angelini, M. G. (2013). Production of DTM quality by TLS data. *European Journal of Remote Sensing*, 46(1), 80-103.

Costantino, D., Pepe, M., & Alfio, V. S. (2020). Point Cloud accuracy of Smartphone Images: Applications in Cultural Heritage Environment. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(4).

Costantino, D., Pepe, M., & Restuccia, A. G. (2021). Scan-to-HBIM for conservation and preservation of Cultural Heritage building: The case study of San Nicola in Montedoro church (Italy). *Applied Geomatics*, 1-15.

Costantino, D., Vozza, G., Pepe, M., & Alfio, V. S. (2022). Smartphone LiDAR Technologies for Surveying and Reality Modelling in Urban Scenarios: Evaluation Methods, Performance and Challenges. *Applied System Innovation*, 5(4), 63.

Daniels, J., Silva, C. T., & Cohen, E. (2009, July). Localized quadrilateral coarsening. *In Computer Graphics Forum* (Vol. 28, No. 5, pp. 1437-1444). Oxford, UK: Blackwell Publishing Ltd.

Daniels, J., Silva, C. T., Shepherd, J., & Cohen, E. (2008). Quadrilateral mesh simplification. *ACM transactions on graphics* (TOG), 27(5), 1-9.

De Loera, J., Rambau, J., & Santos, F. (2010). Triangulations: structures for algorithms and applications (Vol. 25). *Springer Science & Business Media*.

Dhond, U. R., & Aggarwal, J. K. (1989). Structure from stereo-a review. *IEEE transactions on systems, man, and cybernetics*, 19(6), 1489-1510.

Diara, F., & Rinaudo, F. (2018). Open source HBIM for cultural heritage: A project proposal. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 303-309

Dore, C., & Murphy, M. (2012, September). Integration of Historic Building Information Modeling (HBIM) and 3D GIS for recording and managing Cultural Heritage sites. *In 2012 18th International conference on virtual systems and multimedia* (pp. 369-376). IEEE.

Dore, C., Murphy, M., McCarthy, S., Brechin, F., Casidy, C., & Dirix, E. (2015). Structural simulations and conservation analysis-historic building information model (HBIM). *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40, 351-357.

Du, Q., & Gunzburger, M. (2002). Grid generation and optimization based on centroidal Voronoi tessellations. *Applied mathematics and computation*, 133(2-3), 591-607.

Erten, H., Üngör, A., & Zhao, C. (2009). Mesh smoothing algorithms for complex geometric domains. *In Proceedings of the 18th international meshing roundtable* (pp. 175-193). Springer Berlin Heidelberg.

Faugeras, O., & Faugeras, O. A. (1993). Three-dimensional computer vision: a geometric viewpoint. *MIT press*.

Field, D. A. (1988). Laplacian smoothing and Delaunay triangulations. *Communications in applied numerical methods*, 4(6), 709-712

Fischler, M; Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Readings in Computer Vision: Issues, Problems, Principles, and Paradigms, 1987, 726–740. https://doi.org/10.1016/B978-0-08-051581-6.50070-2

Fraser, C. Camera calibration considerations for UAV photogrammetry. *In Proceedings of the ISPRS TC II Symposium: Towards Photogrammetry, Riva del Garda, Italy, 3–7 June 2018*.

Fraser, C. S.: Digital camera self-calibration, *ISPRS Journal of Photogrammetry and Remote sensing* 52(4), 1997, 149-159.

Fraser, C.S. Network Design in Close-range Photogrammetry and Machine Vision. *In Proceedings of the 26th International CIPA Symposium 2017, Ottawa, ON, Canada, 28 August–1 September 2017*; pp. 256–282.

Freitag, L.A.: On combining Laplacian and optimization-based mesh smoothing techniques. *In Trends in Unstructured Mesh Generation*, pp. 37–43 (1997).

Funari, M. F., Hajjat, A. E., Masciotta, M. G., Oliveira, D. V., & Lourenço, P. B. (2021). A parametric scan-to-FEM framework for the digital twin generation of historic masonry structures. *Sustainability*, 13(19), 11088.

Furukawa, Y., & Hernández, C. (2015). Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2), 1-148.

Ganovelli, F., Corsini, M., Pattanaik, S., & Di Benedetto, M. (2014). Introduction to computer graphics: A practical learning approach. *CRC Press*.

Gu, X., Gortler, S. J., & Hoppe, H. (2002, July). Geometry images. *In Proceedings of the 29th annual conference on Computer graphics and interactive techniques* (pp. 355-361).

Guarnieri, A., Fissore, F., Masiero, A., Di Donna, A., Coppa, U., & Vettore, A. (2017). From survey to FEM analysis for documentation of built heritage: The case study of Villa Revedin-Bolasco. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(5W1), 527-533.

Girardeau-Montaut, D. *CloudCompare*; EDF R&D Telecom ParisTech: Paris, France, 2016.

Guidi, G., Russo, M., & Beraldin, J. A. (2010). *Acquisizione 3D e modellazione poligonale*, ed Mc-Graw-Hill.

Hartley, R.; Zisserman, A. Multiple View Geometry in Computer Vision; *Cambridge University Press*:Cambridge, UK, 2003..

Heng, M., & Erkorkmaz, K. (2010). Design of a NURBS interpolator with minimal feed fluctuation and continuous feed modulation capability. *International Journal of Machine Tools and Manufacture*, 50(3), 281-293.

Herban, S., Costantino, D., Alfio, V. S., & Pepe, M. (2022). Use of low-cost spherical cameras for the digitisation of cultural heritage structures into 3D point clouds. *Journal of Imaging*, 8(1), 13.

Hoxhaj, D., Mazzucato, N., Montis, M., & Sommacal, M. (2008). *Localizzazione e SLAM con Wireless Sensors Network* (WSN).

Hu, Y., Jiang, X., Huo, G., Su, C., Li, H., & Zheng, Z. (2023). A novel method for calculating interpolation points of NURBS curves based on chord length-parameter ratio.

Ji, S., Lei, L., Zhao, J., Lu, X., & Gao, H. (2021). An adaptive real-time NURBS curve interpolation for 4-axis polishing machine tool. *Robotics and Computer-Integrated Manufacturing*, 67, 102025.

Kabadayi, A., Yunus, K., & Yiğit, A. Y. (2020). Comparison of documentation cultural artifacts using the 3D model in different software. *Mersin Photogrammetry Journal*, 2(2), 51-58.

Kangas, A., Gobakken, T., Puliti, S., Hauglin, M., & Naesset, E. (2018). Value of airborne laser scanning and digital aerial photogrammetry data in forest decision making. *Silva Fennica*, 52(1).

Kazhdan, M., Bolitho, M., & Hoppe, H. (2006, June). Poisson surface reconstruction. *In Proceedings of the fourth Eurographics symposium on Geometry processing* (Vol. 7, p. 0).

Kazhdan, M., Klein, A., Dalal, K., & Hoppe, H. (2007, July). Unconstrained isosurface extraction on arbitrary octrees. *In Symposium on Geometry Processing* (Vol. 7, No. 2).

Kiilerich, B. (2015). The Barletta Colossos revisited. *Acta ad archaeologiam et artium historiam pertinentia*, 28(14 NS), 55-72.

Kovacs, D., Myles, A., & Zorin, D. (2010, September). Anisotropic quadrangulation. *In Proceedings of the 14th ACM Symposium on solid and physical modeling* (pp. 137-146).

Krishna, R. (2017). Computer vision: *Foundations and applications*. Reference Book, 213.

Łącka, M. (2021). Analysis of algorithms for automatic classification of photogrammetric point clouds to build a digital terrain model in a shoreline area. *Zeszyty Naukowe Akademii Morskiej w Szczecinie*.

Li, T., Jiang, C., Bian, Z., Wang, M., & Niu, X. (2020, March). A review of true orthophoto rectification algorithms. *In IOP Conference Series: Materials Science and Engineering* (Vol. 780, No. 2, p. 022035). IOP Publishing.

Lorensen, W. E. (1987). Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics (Proceedings of SIGGRAPH'87)*, 21(4), 163-170.

Marton, Z. C., Rusu, R. B., & Beetz, M. (2009, May). On fast surface reconstruction methods for large and noisy point clouds. *In 2009 IEEE international conference on robotics and automation* (pp. 3218-3223). IEEE

Nguyen, V. P., Kerfriden, P., & Bordas, S. (2013). Isogeometric cohesive elements for two and three dimensional composite delamination analysis. *arXiv preprint* arXiv:1305.2738.

Nieto, J. E., Moyano, J. J., Rico Delgado, F., & Antón García, D. (2016). Management of built heritage via HBIM Project: A case of study of flooring and tiling. *Virtual Archaeology Review*, 7(14), 1-12.

Paris, L., & Wahbeh, W. (2016). Survey and representation of the parametric geometries in HBIM. *Disegnarecon*, 9(16), 12-1

Pepe M. & Costantino, D. (2020). Techniques, tools, platforms and algorithms in close range photogrammetry in building 3D model and 2D representation of objects and complex architectures. *Comput. Aided Des. Appl*, 18, 42-65.

Pepe, M. (2017). A survey by Airborne Laser Scanner of open large structure: A case study of Pompeii Amphitheatre. *ARPN J. Eng. Appl. Sci*, 12, 1-11.

Pepe, M., & Costantino, D. (2020). Techniques, tools, platforms and algorithms in close range photogrammetry in building 3D model and 2D representation of objects and complex architectures. *Comput. Aided Des. Appl*, 18, 42-65.

Pepe, M., Costantino, D., & Restuccia Garofalo, A. (2020). An efficient pipeline to obtain 3D model for HBIM and structural analysis purposes from 3D point clouds. *Applied Sciences*, 10(4), 1235.

Petitjean, S., & Boyer, E. (2001). Regular and non-regular point sets: Properties and reconstruction. *Computational Geometry*, 19(2-3), 101-126

Pfeuffer, C., Rahrig, M., Snethlage, R., & Drewello, R. (2018). 3D mapping as a tool for the planning of preservation measures on sculptures made of natural stone. *Environmental Earth Sciences*, 77, 1-12.

Remondino, F.: Worth a thousand words – photogrammetry for archaeological 3D surveying, in: Opitz, R.S. 2006.

Riccardelli, C., Morris, M., Wheeler, G., Soultanian, J., Becker, L., & Street, R. (2014). The treatment of Tullio Lombardo's Adam: a new approach to the conservation of monumental marble sculpture. *Metropolitan Museum Journal*, 49(1), 48-116.

Russo, M., Remondino, F., & Guidi, G. (2011). Principali tecniche e strumenti per il rilievo tridimensionale in ambito archeologico. *Archeologia e calcolatori*, 22, 169-198.

Santini, S., Baggio, C., Sabbatini, V., & Sebastiani, C. (2021). Setup Optimization of Experimental Measures on a Historical Building: The Octagonal Hall of the Diocletian's Bath. *Heritage*, 4(3), 2205-2223.

Schonberger, J. L., & Frahm, J. M. (2016). Structure-from-motion revisited. *In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4104-4113).

Schwalbe, E. (2005, February). Geometric modelling and calibration of fisheye lens camera systems. *In Proc. 2nd Panoramic Photogrammetry Workshop, Int. Archives of Photogrammetry and Remote Sensing* (Vol. 36, No. Part 5, p. W8).

Si, H.: TetGen, a Delaunay-based quality tetrahedral mesh generator, ACM Transactions on Mathematical Software, 41(2), 2015, 11.

Solem, J.E. Programming Computer Vision with Python: Tools and Algorithms for Analyzing Images; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2012.

Spangher, A., Visintini, D., Tucci, G., & Bonora, V. (2017). Geomatic 3D modeling of a statue (also) for structural analysis and risk evaluation: the example of San Giovannino Martelli in Florence. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 61-68.

Szeliski, R. (2022). Computer vision: algorithms and applications. *Springer Nature.*

Sztwiertnia, D., Ochałek, A., Tama, A., & Lewińska, P. (2021). HBIM (heritage building information modell) of the Wang stave church in Karpacz–case study. *International Journal of Architectural Heritage*, 15(5), 713-727.

Tarini, M., Pietroni, N., Cignoni, P., Panozzo, D., & Puppo, E. (2010, May). Practical quad mesh simplification. *In Computer Graphics Forum* (Vol. 29, No. 2, pp. 407-418). Oxford, UK: Blackwell Publishing Ltd.

Tsai, V. J.: Delaunay triangulations in TIN creation: an overview and a linear-time algorithm, *International Journal of Geographical Information Science*, 7(6), 1993, 501-524.

Velho, L., & Zorin, D. (2001). 4–8 Subdivision. *Computer Aided Geometric Design*, 18(5), 397-427.

Visintini, D., & Spangher, A. (2014, October). Il contributo della Geomatica per l'analisi strutturale dei beni culturali: l'esempio di una statua romana lesionata. *In Proceedings of the Atti 18a Conferenza Nazionale ASITA, Firenze, Italy* (pp. 14-16).

Visintini, D., Marcon, E., Pantò, G., Canevese, E. P., De Gottardo, T., & Bertani, I. (2019). Advanced 3d modeling versus building information modeling: The case study of palazzo ettoreo in sacile (Italy). *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 1137-1143.

Vosselman, G., & Maas, H. G. (2010). Airborne and terrestrial laser scanning. *CRC press*.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AHRS - Attitude and Heading Reference System
AI - Artificial Intelligence
ALS - Aerial Laser Scanner
API - Application Programming Interface
BA - Bundle adjustment
BIM - Building Information Modeling
BoF - Bag of Features
BoVW - Bag of Visual Words
C2M - Cloud to Mesh
CAM - Computer-Aided Modelling
CCD - Charged-Couple Device
CML - Concurrent Mapping and Localisation
CMOS - Complementary metal-oxide semiconductor
CMP - Continuous MultiPulse
CNN - Convolutional Neural Network
CP - Check Point
CV - Computer Vision
CV - Control Vertices
DL - Deep Learning
dSLR - digital Single-Lens-Reflex
DSM - Digital Surface Models
DTM - Digital Terrain Models
DTM - Delaunay Triangulation
EP - End Points
FEA - Finite Element Analysis
FEM - Finite Element Method
FOV - Field Of View
GCP - Ground Control Point
GIS - Geographic Information System
GNSS - Global Navigation Satellite System

GSD - Ground Sample Distance
H-BIM - Heritage-BIM
H-FEM - Heritage-FEM
ICP - Iterative Closest Point
IFC - Industry Foundation Classes
IMU - Inertial Measurement Unit
INS - Inertial Navigation Systems
LiDAR - Light Detection And Ranging
LOD - Level of Detail
MEMS - Micro-Electro-Mechanical Systems
ML - Machine Learning
MMS - Mobile Mapping System
MTA - Multiple Time Around
MTF - Modulation Transfer Function
MVS - Multi View Stereo
NIR - Near InfraRed
NURBS - Non-Uniform Rational B-Splines
OCR - Optical Character Recognition
ODT - Optimal Delaunay Triangulation
PnP - Perspective-n-Point
RANSAC - RANdom SAmple Consensus
RMSE - Root Mean Square Error
RNN - Recurrent Neural Network
SfM - Structure from Motion
SIFT - Scale-Invariant Feature Transform
SLAM - Simultaneous Localization and Mapping
TIN - Triangulated Irregular Network
TLS - Terrestrial Laser Scanner
TOF - Time Of Flight
UAV - Unmanned Aerial Vehicle
vSLAM - Visual SLAM

CURRICULUM

First Name: **VINCENZO SAVERIO**
Surname: **ALFIO**
Place and date of birth: **GROTTAGLIE, 21/09/1987**
Nationality: **Italy**

CONTACTS

vincenzosaverio.alfio@poliba.it
https://www.researchgate.net/profile/Vincenzo-Alfio
https://orcid.org/0000-0003-4816-8913

EDUCATION AND TRAINING

- Since August 2022 he was a **research fellow** in the S.S.D. ICAR/06 - Topography and Cartography at the Polytechnic University of Bari (DICATECh Department).
- In 2022, he was a **PhD visiting** at Polytechnic University of Timisoara (Romania).
- From 2022 to 2025 he is a Subject Expert in "GIS and Cultural Heritage" (ICAR/06) at the Department of Pure and Applied Sciences University of Urbino "Carlo Bo".
- Since October 2020 he has been a **PhD student in in Risk and Environmental, Territorial and Building Development** at the Polytechnic University of Bari (DICATECh Department), Tutor: Proff. Domenica COSTANTINO, Massimiliano PEPE.
- From October 2019 to October 2021, he was a **research fellow** at the Polytechnic University of Bari (DICATECh Department) in the S.S.D. ICAR/06 - Topography and Cartography.
- In 2017, he obtained a **Master's degree in Environmental Engineering** with a final grade of 110 cum laude, with a thesis in GIS and Remote Sensing entitled: *Implementazione di dati metrici nel WebGIS per la tutela del patrimonio rurale: caso studio "Comparto G.A.L."*, Proff. Domenica COSTANTINO, Maria Giuseppa ANGELINI, Polytechnic University of Bari - Taranto.
- In 2014, he obtained a **Bachelor's degree in Civil and Environmental Engineering** with a thesis in Topography and Geomatics entitled: *Creazione di un database relazionale (DBMS) come supporto allo sviluppo del territorio rurale delle "Cento Masserie di Crispiano", caso studio: masseria Lupoli*, Prof. Domenica COSTANTINO, Polytechnic University of Bari - Taranto.

SCIENTIFIC PUBLICATIONS

1. Pepe, M., Alfio, V.S., Costantino D., (2023). **Assessment of 3D Model for Photogrammetric Purposes Using AI Tools Based on NeRF Algorithm.** Heritage, 6, 5719-5732.
2. Alfio, V.S., Costantino D., Pepe, M., Rossi, G., Leserri, M. (2023). **Production of high geometric resolution orthophotos by photogrammetric approach for the Royal Racconigi Castel park documentation.** Geographia Technica, 18, 2, pp. 40-50.

3. Pepe, M., Costantino, D., & Alfio, V. S. (2023). **A GIS Procedure to Assess Shoreline Changes over Time Using Multi-temporal Maps: An Analysis of a Sandy Shoreline in Southern Italy over the Last 100 Years**. Geomatics and Environmental Engineering, 17(3), 107-134.
4. Vozza, G., Costantino, D., Pepe, M., & Alfio, V. S. (2023). **Smart Sensors System Based on Smartphones and Methodology for 3D Modelling in Shallow Water Scenarios**. Applied System Innovation, 6(1), 28.
5. Pepe, M., Alfio, V. S., & Costantino, D. (2022). **UAV Platforms and the SfM-MVS Approach in the 3D Surveys and Modelling: A Review in the Cultural Heritage Field**. Applied Sciences, 12(24), 12886.
6. Pepe, M., Alfio, V. S., Costantino, D., Herban, S. **Rapid and Accurate Production of 3D Point Cloud via Latest-Generation Sensors in the Field of Cultural Heritage: A Comparison between SLAM and Spherical Videogrammetry**. Heritage, 5(3), 1910-1928, 2022.
7. Costantino, D., Vozza, G., Pepe, M., Alfio, V. S. **Smartphone LiDAR Technologies for Surveying and Reality Modelling in Urban Scenarios: Evaluation Methods, Performance and Challenges**. Applied System Innovation, 5(4), 63, 2022.
8. Pepe M., Alfio, V.S., Costantino, D., Scaringi, D. **Data for 3D reconstruction and point cloud classification using machine learning in cultural heritage environment**, Data in Brief, Volume 42, 2022.
9. Alfio, V.S., Costantino, D., Pepe, M., Garofalo, A.R. **A Geomatics Approach in Scan to FEM Process Applied to Cultural Heritage Structure: The Case Study of the "Colossus of Barletta"** (2022) Remote Sensing, 14.
10. Costantino, D., Pepe, M., Alfio, V. S., Vozza, G., Occhinegro, M. (2022). **Building 3D CityGML Model and Digital Management of a Vernacular Architecture.** International Review of Civil Engineering, 2022, 13(3), pp. 156–162.
11. Herban, S., Costantino, D., Alfio, V.S., Pepe, M. **Use of Low-Cost Spherical Cameras for the Digitisation of Cultural Heritage Structures into 3D Point Clouds** (2022) Journal of Imaging, 8 (1).
12. Pepe, M., Costantino, D., Alfio, V.S., Zannotti, N. **4D geomatics monitoring of a quarry for the calculation of extracted volumes by tin and grid model: Contribute of UAV photogrammetry** (2021) Geographia Technica, 16 (Special Issue), pp. 1-14.
13. Costantino, D., Vozza, G., Alfio, V.S., Pepe, M. **Strategies for 3d modelling of buildings from airborne laser scanner and photogrammetric data based on free-form and model-driven methods: The case study of the old town centre of Bordeaux (France)** (2021) Applied Sciences (Switzerland), 11 (22).
14. Pepe, M., Costantino, D., Alfio, V.S., Vozza, G., Cartellino, E. **A novel method based on deep learning, gis and geomatics software for building a 3D city model from VHR satellite stereo imagery** (2021) ISPRS International Journal of Geo-Information, 10 (10).

15. Pepe, M., Costantino, D., Alfio, V.S., Restuccia, A.G., Papalino, N.M. **Scan to BIM for the digital management and representation in 3D GIS environment of Cultural Heritage site** (2021) Journal of Cultural Heritage, 50, pp. 115-125.
16. Pepe, M., Costantino, D., Vozza, G., Alfio, V.S. **Comparison of two approaches to GNSS positioning using code pseudoranges generated by smartphone device** (2021) Applied Sciences (Switzerland), 11 (11).
17. Costantino, D., Settembrini, F., Pepe, M., Alfio, V.S. **Develop of new tools for 4D monitoring: Case study of cliff in Apulia Region (Italy)** (2021) Remote Sensing, 13 (9).
18. Pepe, M., Costantino, D., Alfio, V.S., Angelini, M.G., Garofalo, A.R. **A CityGML multiscale approach for the conservation and management of cultural heritage: The case study of the old town of Taranto** (Italy) (2020) ISPRS International Journal of Geo-Information, 9 (7).
19. Costantino, D., Pepe, M., Alfio, V. S. (2020). **Point Cloud accuracy of Smartphone Images: Applications in Cultural Heritage Environment**. International Journal, 9(4).
20. Alfio, V.S., Costantino, D., Pepe, M. **Influence of image tiff format and jpeg compression level in the accuracy of the 3D model and quality of the orthophoto in UAV photogrammetry** (2020) Journal of Imaging, 6 (5).
21. Costantino, D., Angelini, M.G., Alfio, V.S., Claveri, M., Settembrini, F. **Implementation of a system WebGIS open-source for the protection and sustainable management of rural heritage** (2020) Applied Geomatics, 12 (1), pp. 41-54.
22. Costantino, D., Pepe, M., Alfio, V., Carrieri, M. **Geomatic techniques for monitoring and verifying of the wear condition of the runways of the bridge cranes** (2019) International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives, 42 (2/W18), pp. 23-30.
23. Costantino, D., Angelini, M.G., Claveri, M., Alfio, V.S. **GIS and DBMS implementation for the development of rural areas of the "one hundred masserie" of Crispiano** (2016) Geographia Technica, 11 (1), pp. 23-32.

CONFERENCE PROCEEDINGS

1. Pepe, M., Costantino, D., Barnaba, G., Alfio, V. S., & Vozza, G. (2022). **Building 3D City model from painting: the case study of Old Town of Taranto, Italy.** 2022 IMEKO TC-4 International Conference on Metrology for Archaeology and Cultural Heritage University of Calabria, Italy, October 19-21, 2022.
2. Alfio, V.S., Costantino, D., Pepe, M. **Monitoring of a cultural heritage site by Sentinel-2 images and change-detection analysis** (2020) Proceedings of SPIE - The International Society for Optical Engineering, 11534.
3. Sonnessa, A., Saponaro, M., Alfio, V.S., Capolupo, A., Turso, A., Tarantino, E. **Indoor Positioning Methods – A Short Review and First Tests Using a Robotic Platform for Tunnel Monitoring** (2020) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12252 LNCS, pp. 664-679.

Abstract
3D parametric modelling is a current and important research topic for the digitisation of cultural heritage in order to preserve the evolution and historical identity of an area over time; for this reason, cultural heritage assets must be subjected to continuous maintenance and monitoring, as well as valorisation and promotion. In addition, a considerable amount of these assets are often exposed to a high risk of damage caused by natural disasters and a poor state of care and neglect that leads to inevitable loss. In this context, in order to adopt all safeguarding and preservation measures in the field of CH, new sensors, digital acquisition technologies and suitable methodological approaches make it possible to create high-performance three-dimensional models, capable of increasing the detail and representation of particulars and enriching these models with heterogeneous information. However, such 3D models are often difficult to manage as they require very high processing and storage capacities. Therefore, it is necessary to define a suitable methodology that is capable of simplifying and optimising the three-dimensional models obtained from a photogrammetric or TLS approach, preserving their accuracy and metric rigour and, at the same time, performing efficiently in BIM (Building In-formation Modeling) and/or FEM (Finite Element Analysis) applications. The aim of this work is to identify a suitable process that is able, starting from a point cloud obtained by means of geomatic techniques, to generate three-dimensional surface models that are subsequently decimated and optimised according to the complexity of the case and the desired output. To achieve these objectives, a semi-automatic process is described that is capable of transforming a TIN (Triangulated Irregular Network) surface model into a quad-mesh polygonal model and then converting it into NURBS (Non-Uniform Rational Basis-Splines) for subsequent import into a BIM software or FEM solver for finite element calculations.

On the cover: Statue of Heraclius