Newcastle University
Faculty of Science, Agriculture and Engineering
School of Engineering

Efficient and Robust Numerical Modelling of Masonry

Nicko Kassotakis

Thesis submitted for the degree of
Doctor of Philosophy

September 2020

For my paternal grandparents Manolis and Eleni Kassotakis and maternal grandparents Alexander and Emma McKechnie

Abstract

Masonry structures constitute a sizeable portion of the built environment, yet the prediction of their structural behaviour remains an extremely complicated task. Despite the advances in structural engineering, safety assessment methods of masonry lag far behind those of modern materials such as steel and reinforced concrete. High-level numerical methods such as the Discrete Element Method (DEM) are the most advanced and effective tools available for modelling the complex structural behaviour of masonry, yet their robustness depends on the accuracy of the geometric and mechanical properties employed. Although abundant work exists for reliably obtaining and representing material properties, methodical strategies lack for the case of geometric properties, which renders such high-level numerical modelling either inefficient or inaccurate. This research develops a methodological framework for the geometrically-accurate and efficient high-level numerical modelling of masonry structures through the employment of non-contact sensing techniques and automation.

The framework is holistic, encompassing three stages of structural surveying, geometric model development and structural analysis. The first stage entails structural surveying of the masonry structure with non-contact sensing techniques such as terrestrial laser scanning (TLS) and Structure-from-Motion (SfM) photogrammetry. The second stage encompasses the utilisation of geometrical data (discrete points, orthoimages and point clouds) and computer vision geometric model development. The final stage of the framework consists of numerical model development and structural analysis with the DEM. With the specific numerical method, each block and joint are represented as a distinct entity, achieving a more faithful representation of the discontinuous nature of masonry than other state-of-the-art numerical methods, and thus permitting both the accurate and efficient prediction of the in-service and collapse behaviour the analysed structure.

Three main approaches stem from the framework which are implemented on both regular and rubble masonry structures. Firstly, the manual image-based approach is implemented on 25 arch specimens. This approach entails structural analysis of geometric models developed from an orthomosaic of SfM photogrammetry with manual CAD-based block segmentation. By comparing the manual CAD-based geometric models with those of traditional geospatial techniques (i.e. tape measurements), significant differences in: a) collapse load (-1 to 10%); b) stiffness

(-2 to 46%); and c) normal forces (-15 to 22%) were found, demonstrating the importance of employing accurate geometric models. Thereafter, the semi-automated image-based approach is implemented on the same 25 arch specimens. Conversely to the previous approach, structural analysis is now semi-automated, incorporating both SfM photogrammetry and image processing techniques (IPTs). By comparing IPT- and manual CAD-based geometric models, a relatively good agreement was found of collapse load, with differences of up to 7%. Stiffnesses, however, showed partial agreement, with differences of up to 7% for 10 specimens and 24% for 15 specimens). These findings, demonstrate both the potential efficiency and robustness of the framework. Finally, the cloud-based approach is implemented on Caerphilly Castle. This entails semi-automated structural analysis, highly irregular of rubble with structures through TLS and voxelization. For a course voxel size of 50 cm, an unprecedented DEM structural analysis of a full-scale masonry tower was achieved in an affordable time of 71 minutes. Therefore, this thesis ultimately paves the way for improving the efficiency and robustness of the structural analysis of masonry structures.

Acknowledgements

First and foremost, I want to express my gratitude to my supervisory team for providing me with the opportunity to conduct my doctoral studies at Newcastle University. Secondly, I would like to acknowledge the Engineering and Physical Sciences Research Council (EPSRC) for providing the Comparative Award in Science and Engineering (CASE/179/65/82), without which the research presented herein would not have been possible.

I am immensely grateful to Prof. Vasilis Sarhosis, for entrusting me with his invaluable ideas, providing me with endless support, and introducing me into the world of research. His impetus was what started this very research and brought me to Newcastle of which I shall never forget. Besides my initial primary supervisor, I would like to thank my current primary supervisor Prof. Stefano Utili for his guidance, dedication and endless encouragement. He was able to sense my most difficult moments and keep me focused on my research. I finally want to thank my secondary supervisor, Prof. Jon Mills who introduced me into the realm of geospatial engineering. After four years of working together, I have immense respect and admiration for each of my supervisors.

During my doctoral study, I have had the greatest fortune to partake in research collaborations with extremely talented people, all over the world. I am firstly indebted to Prof. Katalin Bagi for hosting me at the Technical University of Budapest and introducing me to the concepts of the structural analysis of masonry structures with *3DEC*. Thereafter, I am forever thankful to Prof. Belen Riveiro and Prof. Borja Conde for hosting me at the University of Vigo, and mentoring me in the processing of geometric data with *Matlab* for automated geometric model development. Finally, I am immensely grateful to Prof. Jason Ingham and Francisco Gonzalez who invited me to the University of Auckland for fruitful research collaborations.

Finally, I cannot omit to thank my friends, both inside and outside the university environment. Of my first and most dear academic friends, is Dr Antonino Iannuzzo whom I met at the technical university of Budapest, whilst embarking on this investigation. Then, within the University of Newcastle itself, I particularly thank my colleagues of both the *GEST* and *Neolab* research groups, especially my close friends Vasileios Angelidakis and Dr Maria Valassia Peppa who have offered me hours of help and support. Outside the university, I have my dearest friends to thank, particularly from the “Hygge Club”, Antonio, Nadege, Stef and Janna.

Contents

List of Figures

List of Tables

List of Abbreviations

ALS	Airborne Laser Scanning
BIM	Building Information Model
C2M	Cloud to Model distance
CAD	Computer-Aided Design
CH	Cultural heritage
CPL	Common plane
DDA	Discontinuous Deformation Analysis
DEM	Discrete Element Method/ Distinct Element Method
DigEM	Digital Elevation Model
DSLR	Digital Single Lens Reflex
EOP	Exterior Orientation Parameter
EFM	Equivalent Frame Method
FDEM	Finite-Discrete Element Method
FEM	Finite Element Method
FELA	Finite Element Limit Analysis
GCP	Ground Control Point
GPR	Ground-Penetrating Radar
HPC	High-Performance Computation
ICP	Iterative Closest Point

IOP	Interior Orientation Parameter
IPT	Image Processing Techniques
LIDAR	Light Detection and Ranging
MAB	Masonry Arch Bridge
MCP	Mixed Complementary Problem
MPEC	Mathematical Program with Equilibrium Constraints
NSCD	Non-Smooth Contact Method
NURBS	Non-Uniform-Rational-Base-Splines
RBSM	Rigid Block Spring Method
RMSE	Root Mean Square Error
RTK	Real Time Kinetics
RVE	Representative Volume Element
SfM	Structure-from-Motion
TLS	Terrestrial Laser Scanning
UAV	Unmanned Aerial Vehicle
URM	Un-Reinforced Masonry

List of Notations

f	Nominal focal length
(X_o, Y_o, Z_o)	Perspective centre coordinates
(x_n, y_n)	Image plane coordinates of n-th target
(X_n, Y_n, Z_n)	Object space coordinates of n-th target
$\mathit{\Lambda}$	Scale factor
M	Rotation matrix
β_v	Vertical viewing angle of the lens
β_h	Horizontal viewing angle of the lens
HSS	Horizontal sensor size
VSS	Vertical sensor size
HN	Horizontal pixel number
VN	Vertical pixel number
α_t	Camera tilt angle
WD	Working distance
GSD_{max}	Maximum ground sampling distance
FOV	Field-of-view
n_{face}	Mesh face number
$Res_{orth.}$	Orthomosaic resolution
AU	Absolute uncertainty
RU	Relative uncertainty
NU	Normalised uncertainty
x_{ortho}	Reference point of orthoimagery, x-axis
z_{ortho}	Reference point of orthoimagery, z-axis
$x_{geom.mod.}$	Reference point of geometric model, x-axis
$z_{geom.mod.}$	Reference point of geometric model, z-axis
x_{min}	Minimum x-axis spatial coordinates
x_{max}	Maximum x-axis spatial coordinates
y_{min}	Minimum y-axis spatial coordinates

y_{max}	Maximum y-axis spatial coordinates
z_{min}	Minimum z-axis spatial coordinates
z_{max}	Maximum z-axis spatial coordinates
$Grid$	Voxel size
N_x	Number of voxels, x-axis
N_y	Number of voxels, y-axis
N_z	Number of voxels, z-axis
$\mathit{\Delta}_x$	Voxel dimension, x-axis
$\mathit{\Delta}_y$	Voxel dimension, y-axis
$\mathit{\Delta}_z$	Voxel dimension, z-axis
P	Dense point cloud
P_x	Dense point cloud, x-axis
P_y	Dense point cloud, y-axis
P_z	Dense point cloud, z-axis
Dim_x	Rounded point cloud x-axis
Dim_y	Rounded point cloud y-axis
Dim_z	Rounded point cloud z-axis
Dim	Rounded point cloud
DVC	Dimensionless voxelized point cloud
EVC	Empty voxelized point cloud
VAC	Volume adjustment coefficient
$u_{acc.}$	Maximum accumulated displacement
C_{tol}	Contact tolerance
$\ddot{x}_i$	Block acceleration
$\dot{x}_i$	Block velocity
a	Mass proportionate damping coefficient
F_i	Total enacting forces on block (sub-contact and block)
g_i	Gravity
$\dot{\omega_\imath}$	Block angular acceleration
ω_i	Block angular velocity

M_i	Block torque
I	Block moment of inertia
e_{ijk}	Permutation tensor
$\varDelta t$	Timestep
t	Time variable
V_i	Contact velocity
$\varDelta \vartheta_i$	Rotational increment
$\varDelta x_i$	Translational increment
C_k	Common plane location
n_i	Contact normal
$\varDelta U_i$	Contact absolute displacement increment
U_i^n	Contact normal displacement increment
U_i^s	Contact shear displacement increment
$\varDelta F^n$	Contact normal force increment
F^n	Contact normal force
F_i^s	Contact shear force increment
F^s	Contact shear force
F_i	Total contact force
R	Adaptive damping constant
P	Absorbed power of damping
$\dot{E}_k$	Rate of change of kinetic energy
$\varDelta t_b$	Node-defined timestep
$\varDelta t_n$	System-defined timestep
m_{min}	Smallest block mass
k_{max}	Maximum contact stiffness of system
E_b	Young's modulus of block
E_m	Shear modulus of mortar
G_b	Young's modulus of block
G_m	Shear modulus of mortar
C	Joint cohesive strength

T	Joint tensile strength
φ	Joint friction angle
K_n	Joint normal stiffness
K_s	Joint shear stiffness
T_{max}	Maximum tensile force
F_{max}^s	Maximum shear force
A_c	Sub-contact area
θ_t	Theoretical inclination angle
g_h	Horizontal gravitational acceleration
g	Vertical gravitational acceleration
λ_h	Horizontal inclination angle multiplier
λ_v	Vertical inclination angle multiplier
g_{hx}	Gravitational acceleration of, x-axis
g_{hy}	Gravitational acceleration of, y-axis
g_{vz}	Gravitational acceleration of, z-axis
ψ	Azimuth of inclination
$\lambda_{h,max}$	Critical inclination angle multiplier
$U_{h,max}$	Critical horizontal displacement
$\mathrm{B_i}$	The *i*-th block
$\mathrm{J_i}$	The *i*-th joint
λ^{hi}	Load multiplier at the formation of the i-th hinge
$U_{x,Bi}$	Horizontal displacement of *i*-th block
F_i^n	Normal force of *i*-th joint
F_i^s	Shear force of *i*-th joint

Chapter 1. Introduction

1.1 Research background

1.1.1 Overview

Masonry is one of the oldest construction materials, still widely adopted in a similar manner to that of thousands of years ago (Lourenco, 1996). In the 19th century, the appearance of building materials such as steel and reinforced concrete have led to its reduction to either load-bearing walls of small-scale buildings or mere infill walls (Hendry, 2001) yet historic masonry structures encompass a considerable proportion of the built environment. While masonry is highly-effective for sustaining vertical compressive loads (Heyman, 1997; Como, 2013), it is particularly vulnerable to loads which would otherwise not affect modern construction materials. For instance, masonry is notably susceptible to out-of-plane horizontal loads and flexural moments (Doherty *et al.*, 2002; Griffith *et al.*, 2007), which is evident in countries of high seismicity, where earthquakes pose a serious threat to masonry dwellings (Godio, 2016). Indicatively, in the 2015 earthquakes of Gorkha, Nepal, damage to old historical buildings (stone/brick) masonry was severe, whereas the surrounding buildings of reinforced concrete were unaffected (Goda *et al.*, 2015). Even in countries without significant seismic actions, masonry structures are at risk. In the UK context, for instance, masonry arch bridges are found to be particularly prone threats which the reinforced concrete counterparts are unaffected. Notably, Olofsson *et al.* (2005) found a significant portion of masonry arch bridges at risk due to threats such as overloading, prolonged traffic exposure, vibrations, settlements, environmental conditions and natural events material degradation. From studies such as the above its evident that the safety assessment of masonry is therefore crucial.

Structural assessment methods can provide an understanding of a masonry structure's complex behaviour, and therefore determine their safety (Dejong, 2009). In actuality, vigorous research has provided an array of methods, the selection of which depends on the: a) type and scale of the structure; b) economic pores available; and c) required accuracy (Lourenco, 2002). Traditional methods include: semi-empirical methods such as the MEXE (The highways agency, 2001) to evaluate the structural capacity of existing masonry arch bridges; analytical methods, such as those formulated in (Housner, 1963); and equilibrium methods such as those proposed (Block, 2005; Dejong, 2009) for historic masonry structures, such as cathedrals and domes. Although all the aforementioned methods are still employed to this day, the information they provide is very limited whilst they lack in accuracy, detail and facility of implementation. Due to the necessity of more sophisticated methods of structural assessment, scientists have recently employed numerical methods, inspired by those employed for modern materials. Of the most advanced numerical methods to this day is the discrete element method (DEM), which is capable of providing a high-level structural assessment through capturing the in-service and collapse behaviour of masonry structures.

1.1.2 Masonry as a construction material

The art of construction by assembling units of stone, clay, brick, or concrete blocks defines masonry. Masonry is classified based on the spatial organisation of its units, the so-called masonry typology, which also significantly influences masonry's structural behaviour (Zhang *et al.*, 2018a). Brick masonry, for instance, can be distinguished into periodic (similar to stone ashlar, in which the patter is termed bond) and rubble. On the other hand, stone masonry has been classified (Vanin *et al.*, 2017; Zhang *et al.*, 2018a) by various levels of regularity, such as: a) irregular stone masonry, with pebbles, irregular stone units (Figure 1-1a); b) uncut stone masonry (Figure 1-1b); c) cut stone masonry with good bonds (Figure 1-1c); d) soft stone regular masonry (Figure 1-1d); e) ashlar masonry, built with sufficiently resistant blocks (Figure 1-1e); and f) ashlar masonry, built with sufficiently resistant blocks and the blocks are perfectly rectangular and all blocks of one row have the same height (Figure 1-1f).

Of major importance to the structural behaviour of masonry is also the cross-section morphology (Binda *et al.*, 2009). For masonry walls, for instance, this may be: a) single leave; b) double leaves without connection; c) double leaves with connection; and treble leaf stone masonry walls. This aspect is important due to its influence on the monolithic behaviour of the masonry (Binda *et al.*, 2009). Finally, the type of structure masonry belongs may also significantly affect its structural behaviour. A classification of masonry structures can be into: a) dwellings; b) row house; c) palaces; d) bell-towers; e) arenas; f) churches and cathedrals (Binda *et al.*, 2009). In this investigation, to facilitate the organisation of the presented studies, masonry structures are classified as such: a) general unreinforced masonry, such as masonry panels and dwellings; b) masonry arches and bridges; c) cultural heritage structures, such as temples, domes, churches, castles and towers.

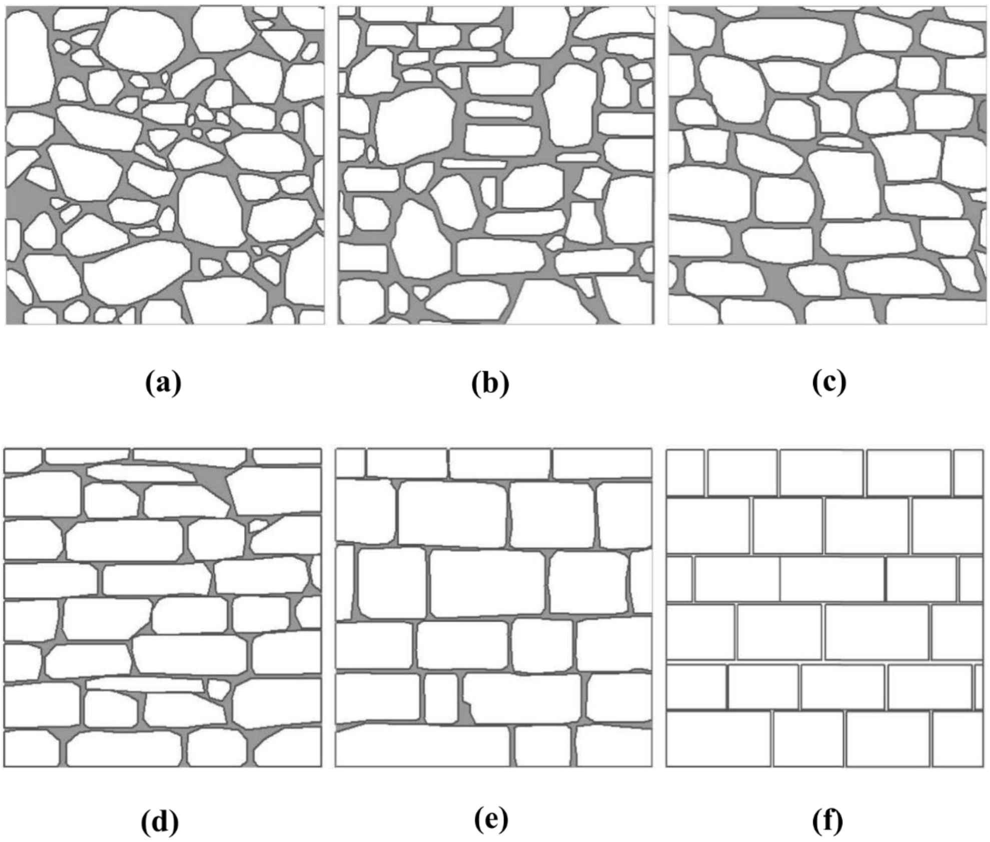


Figure 1-1: Typologies of stone masonry (Vanin et al., 2017; Zhang et al., 2018a): (a) class A; (b) class B; (c) class C; (d) class D; (e) Class E1; and (f) Class E2.

1.1.3 Numerical modelling

Numerical modelling consists of employing a given numerical method in order to carry out structural analysis. Approaches found in the state-of-the-art studies (Milani *et al.*, 2017; D'Altri *et al.*, 2018b; Sarhosis and Lemos, 2018; D'Altri *et al.*, 2019; D'Altri *et al.*, 2019; Forgács *et al.*, 2019; Malomo *et al.*, 2019; Napolitano *et al.*, 2019a; Pulatsu *et al.*, 2019a; Pulatsu *et al.*, 2019b; Sarhosis *et al.*, 2019) are of the most advanced tools of structural analysis available to scientists. This is owed to the understanding of the structural behaviour they provide which includes load-displacement type responses for arbitrary, three-dimensional, full-scale structures encompassing irregularities of mass, stiffness and geometry.

This potential of numerical modelling to provide a comprehensive understanding of the in-service and collapse behaviour of masonry structures renders it indispensable (Godio, 2016) for the safety assessment of masonry structures. However, it is an extremely complicated task to carry out (Asteris *et al.*, 2015), owing to the complex structural behaviour of masonry ranging from linear elastic at low stresses to highly non-linear, for high stresses (Sarhosis *et al.*, 2015). The non-linearity is typically exhibited by crack formation along the joints or within units, depending upon the strength of the materials. After the onset of cracking, further loading results in a reduction in the load-carrying capacity and eventually failure (Sarhosis and Lemos, 2018). Further from the aforementioned, various scenarios of cracking are also possible, as demonstrated for a two-block, brick masonry system in Figure 1-2.

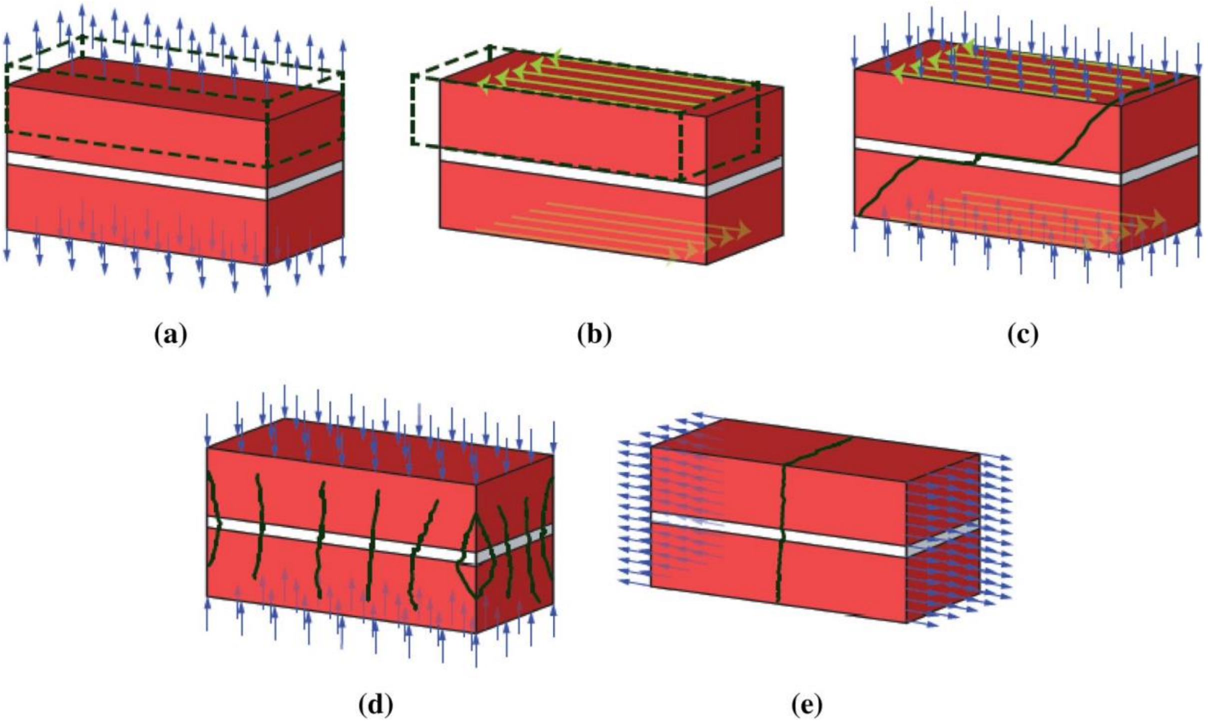


Figure 1-2: Characteristic failure modes of two-block assemblage according to D'Altri et al. (2019): (a) block-mortar tensile failure; (b) block-mortar shear failure; (c) diagonal block crushing failure; (d) masonry crushing failure; and (e) block and mortar tensile failure.

High-level numerical modelling consists of capturing the non-linear behaviour (such as shown in Figure 1-2) robustly and efficiently. However, this, unfortunately, has only been demonstrated in only a few state-of-the-art studies. One such group of numerical modelling approaches entails the employment of the DEM, which has demonstrated high efficacy in masonry, due to its ability in capturing its discontinuous nature. Despite its demonstrated effectivity, the DEM is only employed within the confines of research (Sarhosis *et al.*, 2016e) for the structural assessment of masonry, as opposed to other, more widespread numerical methods such as the Finite Element Method (FEM) or Limit Analysis (LA).

Of the major obstacles in DEM's employment is the fact that for its implementation, there is an inherent need for reliable mechanical and geometrical properties (D'Altri *et al.*, 2019). Whilst significant research has been devoted to reliably acquiring and representing the mechanical properties with the DEM (Sarhosis and Sheng, 2014; Sarhosis *et al.*, 2020), little work has been devoted to geometrical properties. This lack of research into geometry has led to numerical modelling of masonry with the DEM being either: a) limited to small-scale masonry structures (e.g. experimental masonry wall panels); or b) employed for full-scale structures, however with major simplifications of geometry.

Over the last two decades, advances in non-contact sensing techniques such as terrestrial laser scanning (TLS) and Structure-from-Motion (SfM) photogrammetry have started to drastically change the building industry, due to such techniques rapidly and remotely harvesting digital geometric records of objects and features in a point cloud format. SfM photogrammetry is a passive non-contact sensing technique in which, interest points (IPs) are detected in overlapping images of a structure and used to reconstruct a point cloud using common feature matching and triangulation (Westoby *et al.*, 2012). Compared to SfM photogrammetry, TLS is an active non-contact sensing technique. Time-of-flight scanners, which are more relevant to applications of masonry structures, measure distance by timing the emission of a pulse of laser energy to the detection of the reflected signal (Mills and Barber, 2004). Both SfM photogrammetry and TLS have a demonstrated suitability for accurately and rapidly obtaining the complex geometry of masonry structures (Altuntas *et al.*, 2017; Barrile *et al.*, 2017; Barrile *et al.*, 2019; Chen *et al.*, 2019).

From the aforementioned studies, the potential of employing non-contact sensing techniques is evident in the context of numerical modelling. Indeed, recent investigators have transitioned from numerical modelling employing simplified geometric models (i.e. ad-hoc or idealised) to accurate geometric models, developed from the data of non-contact sensing techniques. Although these approaches have significantly augmented the efficiency and robustness of numerical modelling, they have been limited to numerical methods such as the FEM and LA. Until now, very few approaches (Napolitano and Glisic, 2019; Napolitano *et al.*, 2019a; Napolitano *et al.*, 2019c) have regarded high-level numerical modelling approaches, such as the DEM.

1.2 Aims and objectives

This research aims *to develop a methodological framework to improve the efficiency and robustness of the structural analysis of masonry structures using discrete element modelling.* Focus is on the utilisation of the data of non-contact sensing techniques to automate the development of geometrically-accurate geometric models. The main objectives to achieve the research aim are:

1. To evaluate the suitability of the DEM for the structural analysis of masonry structures;

2. To both examine the suitability of SfM photogrammetry for rapidly providing accurate geometric data and approaches for automatically developing geometric models for the DEM from such data;

3. To propose and develop a methodological framework for the structural analysis of masonry structures from discrete points, orthoimages and point clouds with the discrete element method;

4. To quantify the effect of geometric uncertainty on the robustness of the structural analysis of regular masonry structures;

5. To demonstrate the framework's potential to perform semi-automated and robust discrete element modelling of regular masonry structures through the employment of image processing techniques (IPTs);

6. To demonstrate the framework's potential to perform semi-automated and robust discrete element modelling of rubble masonry structures through the employment of voxelization.

1.3 Thesis outline

Seven chapters constitute this thesis, as outlined below.

Chapter 1 sets the research context and provides relevant background information to masonry structures and the crucial role of numerical modelling. It highlights the research motivation and specifies the aims and objectives of the research.

Chapter 2 reviews both the traditional and state-of-the-art numerical modelling approaches in Section 2.1 and Section 2.2. Then, concerning structural surveying, various geospatial techniques are reviewed in Section 2.3. Finally, various approaches of geometric model development are reviewed, in Section 2.4.

Chapter 3 details the proposed three-stage methodological framework. The first stage entails structural surveying with various geospatial techniques, however with an emphasis on SfM photogrammetry. It commences with the numerical formulation of the SfM photogrammetry pipeline and concludes with two separate stepwise workflows for the employment of SfM photogrammetry and TLS for structural surveying. The second stage entails geometric model development which consists of converting geometric data directly in geometric models. For this, three workflows are detailed for developing geometric models from discrete points, orthoimages and point clouds respectively. The third stage entails structural analysis of masonry structures, parting from a given numerical model. It commences with explaining the numerical formulation of the Discrete Element Method and concludes with a stepwise workflow for its employment in the structural analysis of masonry. Three main approaches stem from the methodological framework, the so-called manual image-based, semi-automated image-based, and cloud-based approaches. A further point-based approach is also presented, however, is considered a secondary approach.

Chapter 4 implements the methodological framework on 25 arch specimens. Geometric models are developed from SfM photogrammetry and tape measurements, respectively. After the differences in geometry are calculated, the effect geometric uncertainty on the structural behaviour of masonry is quantified between the varying approaches (and consequent varying geospatial techniques). In this way, the importance of employing accurate geometric models is demonstrated to ensure the robustness of the structural analysis of regular masonry structures.

Chapter 5 implements the methodological framework on the same 25 experimentally tested arch specimens of Chapter 4. Geometric models are developed from SfM photogrammetry yet in this case, with the semi-automated image-based approach is implemented and manual image-based approach. By quantifying the geometric uncertainty on the structural behaviour of masonry when using varying block segmentation (manual CAD-based and IPT-based), good agreement is found and thus the potential of the methodological framework for semi-automated and robust discrete element modelling of regular masonry is demonstrated.

Chapter 6 implements the methodological framework on a full-scale rubble structure. Geometric models (termed voxel models) are developed with various block sizes and orientations. The effect of geometric uncertainty, block size, block orientation, joint strength parameters are investigated. Moreover, the influence of block size is also investigated on the simulation times and block properties since full-scale voxel models can become computationally unmanageable. In this way, the potential of the methodological framework for semi-automated and robust discrete element modelling of rubble masonry is demonstrated.

Chapter 7 draws the investigation's main conclusions by bringing together the findings from Chapter 4, Chapter 5 and Chapter 6. It commences with reviewing the research's contributions. Then, the investigation's course is examined by revisiting the aims and objectives. Finally, the limitations, future recommendations and avenues of investigation are recommended.

Chapter 2. Literature review

This Chapter presents the up to date literature on the: structural analysis, structural surveying and geometric model development of masonry structures. Chapter 2 commences by presenting the fundamental concepts of numerical modelling and then proceeds to review the state-of-the-art approaches and studies, concerning their potential to perform high-level structural analysis efficiently and robustly. Following, geospatial techniques approaches are reviewed for surveying masonry structures, concerning their accuracy and cost-effectivity. Finally, approaches of developing geometric models from the data of structural surveying are reviewed, concerning their facility in implementation.

2.1 Concepts of numerical modelling

Any given numerical modelling approach is implemented through an analysis approach, a modelling strategy, geometric model development and a numerical method. Thus, before the approaches and relative studies can be reported, these fundamental concepts of numerical modelling must be detailed.

2.1.1 Analysis approaches

There are two main analysis approaches for investigating the structural behaviour of masonry structures: the limit analysis and incremental iterative analysis (D'Altri *et al.*, 2019).

Limit analysis

With the LA, only the collapse load of the structure is investigated. Inspired by the limit theorems of plasticity, the LA was formulated in the pioneering work of Heyman (Heyman, 1969) under the basic assumptions:

a) Masonry has an infinite compressive strength;
b) Masonry has a tensile strength equal to zero;
c) Masonry blocks are not permitted to slide between each other (i.e. perfectly plastic shear behaviour of contacts between blocks);
d) Elastic strains are negligible (i.e. limit load occurs with small displacements).

Based on these assumptions, two formulations of the LA can be found; the so-called kinematic and static theorems (also termed upper and lower boundary theorems). According to the lower boundary theorem, a structure is stable, so long as a statically admissible field of internal forces exists for the exerted loads. The problem is solved through a series of linear equations to find the maximum collapse load multiplier (herein termed λ_c) that satisfies equilibrium. For the kinematic theorem, the structure is unstable, when the work of the external forces corresponding to mechanism formation is greater than or equal to zero. In this case, the problem is solved by finding the minimum collapse load multiplier, which corresponds to the virtual work of zero (Smoljanovic *et al.*, 2013b).

Of major importance when implementing the LA within a numerical model is the consideration of the so-called flow rule. Namely, traditional limit analyses require that an associative flow rule be employed. However, this specific flow rule requires that the angle of friction be equal to the angle of dilation, which according to experimental evidence is not well-standing (Van der Pluijm, 1999). As will be discussed in further paragraphs, to overcome this problem, the so-called non-standard limit analyses are also employed.

Incremental iterative analysis

With the incremental-iterative analysis, the structure is loaded in a step-by-step fashion with equilibrium sought after each cycle (or iteration). In this way, the evolution of damage is investigated on the structure. Due to the sequential nature of the analysis, the role of both mechanical and geometric nonlinearity is accounted for, which is a key feature of a high-level structural analysis. Two types of incremental iterative analysis can be found according to D’Altri *et al.* (2019), which are herein reported.

Regarding the non-linear static analysis, the analysed structure is subject to a load which is independent of time. Although time is included, it does not represent an actual physical quantity. Conversely, the monitored displacement or force dominates the progression of the analysis. Within a given numerical framework, equilibrium is reached by solving non-linear differential equations. Typical numerical frameworks employed for the iterative solution of the non-linear differential equations include the Picard, Newton-Raphson and Riks methods (Clough, 1993). A major limitation to the non-linear static analyses is that they are only suitable for quasi-static loading applications such as the so-called pushover analyses or the so-called sequential linear analysis (Dejong, 2009).

In regard to the non-linear dynamic analysis, the analysed structure is subject to a load which is dependent on time. To implement this in a given numerical method, time integration methods are employed to solve the equations of motion governing the structural system for each time step. Some typical time integration methods include the central differences method, the Euler Gauss and Newmark Beta (Clough, 1993). A basic classification of time integration methods is based on their type of solution which may be either explicit or implicit (Clough, 1993). In the former, the solution is solely dependent on the results of the previous step. In the latter, the solution is conditioned by both values of the previous timestep's solution as well as the current timestep's (Clough, 1993). Whilst explicit methods are associated with a lesser computational burden, implicit are considered advantageous since their solution is unconditionally stable, permitting smaller timestep values to be adopted and consequently a more rapid convergence (Sarhosis *et al.*, 2016c). The non-linear dynamic analysis is the most advanced analysis approach, which can be adopted for extreme dynamic loading cases, such as the seismic assessment of masonry structures. Furthermore, it can also be applied for simulating quasi-static loading tests, as will be the case of this investigation.

2.1.2 Modelling strategies

Depending on the required accuracy and detail required of the structural analysis, multiple modelling strategies have been proposed by various investigators (Lourenco, 2002; Asteris *et al.*, 2015; D'Altri *et al.*, 2019). Modelling strategies are distinguished according to the manner in which the heterogeneity of masonry is simulated in a given numerical model. Effectively, this is in terms of material properties (i.e. constitutive models); and geometrical representation (i.e. geometric models as will be further detailed in Section 2.1.3). According to a recent classification (D'Altri *et al.*, 2019), the following modelling strategies can be employed:

- Block-based models;
- Continuum models;
- Macroelement models;
- Geometry-based models.

Block-based models

In the block-based modelling strategy, Young’s modulus, Poisson’s ratio, and inelastic properties of both the masonry block and mortar are accounted for. Furthermore, the masonry blocks and joints are explicitly described within the geometric model as blocks and mortar elements. Due to this very detail, a simplified variation of the block-based modelling is usually employed in which the blocks are expanded to maintain the initial geometry and mortar is replaced with a zero-thickness element. This is particularly advantageous for regular masonry structures (and generally low bond strength masonry), where the anisotropy of the masonry is well-defined (D’Altri *et al.*, 2019) and failure predominantly occurs due to sliding and cracking between blocks. The main advantages of the block-based modelling strategy are: a) it enables a realistic description of a given masonry structure, with an explicit representation of the masonry’s anisotropy and structure’s geometric details; b) it facilitates the capturing of accurate structural behaviour and failure modes; and c) mechanical properties can be derived straightforwardly, directly from small-scale experiments. However, its limitations are also notable, such as that: a) it can be computationally burdensome, especially for the case of large-scale masonry structures; b) the development of geometric models is extremely cumbersome which can also severely delay the given numerical modelling approach (as will be further detailed in Section 2.1.3); and c) it is only employed in research except for few high-level practising engineers (D’Altri *et al.*, 2019).

Continuum models

In this strategy, the structure is considered as a homogenous anisotropic continuum which is especially effective for instances where the anisotropy of masonry is less significant (D’Altri *et al.*, 2019) such as the case of rubble masonry structures. The main advantages of continuum modelling, in comparison with block-based modelling, are: a) the anisotropy of the masonry is not explicitly represented with a block-by-block manner, which significantly simplifies geometric model development (as will be further detailed in Section 2.1.3); and b) since the geometric refinement is reduced, the computational burden is significantly reduced accordingly. However, a major limitation of this strategy is that the determination of material properties is an extremely complicated task; which is accomplished through either the employment of experimentally derived constitutive laws or homogenisation processes. Furthermore, another limitation is that only knowledge of global behaviour only is emphasised whilst the local behaviour (i.e. interaction between blocks) is neglected.

Macroelement models

In this strategy, the masonry structure is considered as an assembly of structural components, typically piers and spandrels (vertical and horizontal load-bearing elements respectively). Within the geometric model, spandrels and piers are assigned based on experimental knowledge of the modelled structure. Regarding material properties, a constitutive law is employed that governs the structural behaviour on a spandrel and pier level. Owing to this simplicity, the main advantages of the macroelement modelling strategy is its low computational burden which makes it of the most employed for the dynamic analysis of URM (D'Altri *et al.*, 2019). However, it has significant limitations, such as: a) the initial predefinition of the spandrels and piers appears to be oversimplified which necessitated judicious application; and b) the local behaviour associated with the anisotropy of masonry cannot be accounted for.

Geometry-based models

In this strategy, further, than the loading scenario, the only other variable is the geometric model which is described as a rigid medium. Such models typically employ LA approaches and lead to the estimation of the collapse or equilibrium load through the lower and upper boundary theorems respectively. The main advantage of this strategy is its simplicity in implementation and low computational burden (D'Altri *et al.*, 2019). However, the limitation of such models is associated with their simplicity (i.e. the only variables being geometry and loading), which does not permit an in-depth understanding of the structural behaviour of masonry. Furthermore, they share the inherent limitations of the LA models which is the fact that they only provide the collapse load and failure mechanism.

2.1.3 Geometric model development

Geometric model development (otherwise termed as solid model development), is the procedure of developing the geometry of the structure suitable for structural analysis with a given numerical method. According to various investigations (Brenner, 2005; Hinks *et al.*, 2012), the geometric models can be described by: a) a boundary representation, in which the geometric model represents the masonry structure explicitly; b) constructive solid geometry (CSG), in which the geometric model represents the masonry structure from Boolean operations of simpler geometric objects; and c) spatial enumeration, in which the geometric model of the masonry structure is represented as a composition of smaller geometric models occupying the domain of the masonry structure, e.g. voxels.

Of major importance to the geometric model development process is the numerical method employed which may be classified into continuum and discontinuum (Bobet *et al.*, 2009). Within continuum methods such as the FEM, the discontinuities may be implemented either implicitly or explicitly. Conversely, in discontinuum methods such as the DEM, discontinuities are explicitly incorporated.

2.2 Numerical modelling approaches

In this Section, the current state-of-the-art of numerical modelling approaches is reviewed. Due to the large volume of existent research on the numerical modelling of masonry, a comprehensive review of all the literature would be unrealistic. Thus, the current Section is limited to the two most advanced strategies which are the continuum and block-based, according to D'Altri *et al.* (2019). The interested reader is referred to (Roca *et al.*, 2010; Smoljanovic *et al.*, 2013b; Asteris *et al.*, 2015; Sarhosis *et al.*, 2016e; Baraldi *et al.*, 2017; Ademovic and Hadzima-Nyarko, 2019; D'Altri *et al.*, 2019).

2.2.1 Continuum models

LA continuum models

LA models are well-established for the structural analysis of masonry structures. The main advantage of such models is that they provide the masonry structure's collapse load and failure mode, rapidly with a relatively little computational demand in most cases. For this very reason, they are particularly attractive to practising engineers through their availability within various commercial software packages such as *Ring* (LimitState, 2019). However, a significant limitation is that only the collapse load is provided whilst failure displacements are unknown (i.e. load-displacement type responses are unattainable). Furthermore, the assumptions the LA employs, especially concerning material properties can be oversimplified.

A first class of LA models were developed within the continuum modelling strategy. In the so-called direct-continuum models, the macroscopic constitutive law ascribed to the numerical model is derived directly from experiments. For instance, Milani *et al.* (2012) developed direct continuum models of a full-scale historic masonry tower, employing a piecewise linear approximation with a Mohr-Coulomb failure criterion and tension cut-off and cap in compression for masonry interfaces. Concerning the solution, linear programming was employed. A second approach is based on the homogenisation theory; in which masonry is represented with a periodic regular texture while the macroscopic constitutive law is obtained from the solution of a boundary cell problem in at a cell level. Within this approach, Milani *et al.* (2006a) presented a pioneering investigation with homogenisation within the LA, by employing the polynomial expression of the stress field inside a representative volume element (RVE) whist the structural capacity of masonry was deduced by utilizing the strength domain. Many state-of-the-art studies have subsequently followed with advanced homogenisation approaches (Milani *et al.*, 2006b; Cecchi *et al.*, 2007; Cecchi and Milani, 2008; Milani, 2011; Cavalagli *et al.*, 2013; Godio *et al.*, 2017).

FEM continuum models

Whilst the LA is effective for an accurate prediction of the collapse load, it cannot provide a detailed structural analysis, which consists in describing the in-service and collapse behaviour. In the need for more sophisticated structural analysis than the LA, FEM continuum models have also been employed, which can provide load-displacement type responses. The main advantages of the FEM continuum models according to Sarhosis *et al.* (2016e), include:

- Straightforward implementation due to a multitude of commercial practice-oriented software packages;
- Facilitated geometric model development through user-friendly tools;
- Common application by both practising engineers and researchers.

The earliest class of FEM continuum models attempted to simulate masonry on a global scale through ascribing a constitutive law capable of reproducing the anisotropy of masonry. The so-called non-tension models, developed by Del Piero (1989), were built on the idealisation that masonry has a zero tensile strength. Whilst they were effective for an initiatory structural analysis, they could not be adopted for the tensile regime of masonry structures. Effectively, since actual masonry structures do possess a tensile strength (even if small), non-tension models cannot simulate the post-peak behaviour of masonry structures and lead to incorrect failure modes.

The necessity for capturing the non-linear behaviour of masonry led to the replacement of non-tension models with more advanced, non-linear models, inspired by the numerous smeared crack, orthotropic plasticity and orthotropic damage models of reinforced concrete (Hofstetter *et al.*, 2011; Jirásek, 2011). Lotfi and Shing (1991) pioneered the employment of non-linear models by evaluating the smeared crack model on masonry shear walls. Generally, smeared crack models are advantageous for historic masonry structures (such as rubble masonry structures) due to: a) the randomness of the masonry's geometry, the assumption of isotropy is well-standing; and b) their facility of implementation since are found within most commercial FEM codes (D'Altri *et al.*, 2019). However, for the case of regular masonry, especially of a low-bond strength, the assumption of orthotropy is not well-standing and smeared crack models cannot be employed.

In an attempt to overcome the limitations of smeared crack models, Lourenco and Rots (1997), formulated the first orthotropic plasticity models which effectively represented the tensile strength of the material in the principal directions. The specific models were validated in comparison with experimental masonry panels and proved extremely effective for capturing experimental behaviours. Recently, orthotropic damage models have been extensively employed, including many state-of-the-art studies. (Lopez *et al.*, 1999; Berto *et al.*, 2002; Reyes *et al.*, 2009; Pelà *et al.*, 2011; Pelà *et al.*, 2013; Pelà *et al.*, 2014).

As with the LA, homogenisation processes have also been applied to the FEM. Pietruszczak and Niu (1992) presented an early homogenisation approach which was performed on masonry wall panels with the FEM in a two-stage manner, by introducing the head joints and bed joints[1] as elastic inclusions and dispersed sets of weaknesses. Antoine (Anthoine, 1995) however, formalised the homogenisation procedure by carrying it out in one step only, introducing the actual pane thickness and the actual brick geometry. Further on, more advanced homogenisation approaches have been developed, capable of considering complex failure mechanisms such as in (Lopez *et al.*, 1999; Zucchini and Lourenço, 2002). State-of-the-art homogenisation studies have included multi-scale approaches which overcome mesh

[1] Head and bed joints are horizontal and vertical discontinuities of the masonry respectively.

dependency whilst representing localised failure (Leonetti *et al.*, 2018). This is carried out with a so-called first-order homogenisation until a threshold of damage is reached. After reaching the threshold, the damaged region of interest is replaced with a heterogeneous material (D'Altri *et al.*, 2019).

2.2.2 Block-based models

LA block-based models

A second class of LA models have been developed specifically for the block-based strategy. Livesley (1978) presented a pioneering lower bound LA solution for 2D masonry arches. However, due to the employed associated flow rule (as aforementioned in Section 2.1.1), two major shortcomings arose from this model: a) unreliable failure mode prediction; and b) overestimation of collapse load. Many investigators thus attempted to overcome these limitations, by implementing a non-associated flow rule. Notably, Fishwick (1996) developed a mixed lower and upper bound solution, to carry out non-associated LA of multiring arch bridges. This was through utilizing a mathematical program to solve an underlying mixed complementary problem (MCP) involving a system of orthogonal sign-constrained vectors. Despite its robustness for minimum collapse load calculation, it was effective for a small number of blocks only. Similarly, Baggio and Trovalusci (Baggio and Trovalusci, 1998; Baggio and Trovalusci, 2000) developed an MCP-like non-associated solution which attempted to find the minimum load factor by direct minimisation (with the so-called optimisation problem) under complementary constraints. This solution was also found unmanageable for structures of several blocks. Subsequently, Ferris and Tin-Loi (2001) proposed another approach for the collapse loads of discrete rigid block systems through a constrained optimisation problem known as a Mathematical Program with Equilibrium Constraints (MPEC). Orduña and Lourenço (2005) additionally employed a novel load path following procedure which yielded in the robust structural analysis of 3D masonry assemblies. Finally, in recent years, other investigators have employed more sophisticate techniques such as cone programming (Portioli *et al.*, 2014) to carry out the non-associated LA.

Despite the effectivity of non-associated LA models for the analysis of masonry, two disadvantages are made apparent here: a) they all assume infinitely resistant bricks which permits plastic dissipation at the interfaces; b) the combination of a non-tension and rigid block can lead miscalculation of the failure load (Milani, 2008). Thus, another group of investigators, have employed the so-called finite element limit analysis (FELA), without resorting to a non-associative flow rule. In a pioneering study, Sutcliffe *et al.* (2001) developed a novel lower bound finite element limit analysis (FELA) solution to calculate collapse loads of unreinforced masonry shear walls. The solution was derived from the impositions of equilibrium with appropriate yield and stress boundary conditions. Later on, another FELA approach was also presented, however with an upper bound solution (Milani, 2008). This included interfaces with a Mohr-Coulomb failure criterion, a tension cut-off and cap in compression for mortar joints in combination with a Mohr-Coulomb failure criterion for bricks, enabling complex failure modes (such as masonry crushing) to be captured. Other studies have also followed this approach (Milani, 2008; Milani *et al.*, 2009). Finally, the upper bound FELA has also been applied to the 2D static analysis of large-scale structures such as masonry arch bridges (Cavicchi and Gambarotta, 2006).

FEM block-based models

A significant scientific intent has also been devoted to the development of FEM models capable of block-based modelling. Page (1978) pioneered the so-called textured continuum approach, in which the discontinuities of masonry are represented implicitly by locally altering the texture of the mesh, corresponding to the mortar. Whilst many studies have the textured continuum approach (Ali and Page, 1988; Addessi and Sacco, 2016; Petracca *et al.*, 2017; Serpieri *et al.*, 2017), a significant limitation is the implicit representation of discontinuities (based on a continuum). This can primarily make it difficult to capture specific failure modes (e.g. sliding and separation of blocks) and secondarily computationally expensive to implement.

Another approach involves the explicit representation of masonry's discontinuities through zero-thickness interfaces. This so-called interface element approach was formulated by Lotfi and Shing (1994). Later on, the pioneering study of Lourenco and Rots (1997) greatly improved the interface approach with the so-called multi-surface models in which, the structure's damage was gathered at the interfaces only, permitting a notably increased efficiency of structural analysis. Recently, owing to the effectivity of the multi-surface models, they have been enhanced by other investigators, particularly for masonry wall panels (Gambarotta and Lagomarsino, 1997; Macorini and Izzuddin, 2011; Chisari *et al.*, 2018). Another confirmation of their effectivity is that they are of the few FEM block-based models which have been successfully employed for full-scale masonry structures such as bridges (Zhang *et al.*, 2016;

Zhang *et al.*, 2017; Zhang *et al.*, 2018b; Tubaldi *et al.*, 2019). However, here, a setback of the approach is also made apparent in the fact that, especially for the case of full-scale structures, they appear computationally burdensome, necessitating high power computational (HPC) facilities.

Whilst the interface approach is highly-effective, the representation of complex behaviours such as the crushing of masonry is still a challenge, whilst numerical properties cannot be obtained with ease. Attempting to overcome such difficulties, an innovative research group (D'Altri *et al.*, 2018a; D'Altri *et al.*, 2019) introduced a novel, so-called contact-based approach within a FEM framework (within the commercial software *Abaqus* (Simulia Inc., 2017)). Specifically, contact-based interfaces were coupled with 3D non-linear-damaging textured blocks to explicitly represent the mortar and masonry, representing one of a handful of detailed block-based models available in the literature. In the initial investigation (D'Altri *et al.*, 2018a), the approach was proposed and implemented on experimental panels and validated, for quasi-static loading. In the follow-up study (D'Altri *et al.*, 2019), the approach was implemented for cyclic loading on a full-scale experimental terraced house, yielding unprecedented results for a FEM model such as large displacements (i.e. 50 mm), crushing effects and manageable computational times. Additionally, material properties were derived directly from small scale experiments.

DEM models

Despite the apparent suitability of the FEM block-based models to simulate the heterogeneous nature of masonry, the state-of-the-art approaches such as the aforementioned interfaces approaches appear to still be generally computationally expensive, and in some case necessitating HPC resources. Furthermore, the more recent, innovating contact-based approaches, are also evidently suitable for the block-based modelling of masonry, and computationally manageable, however, have only found a small application which means their employability and efficiency is still questionable. In an attempt to overcome such difficulties, researches have been attracted to discontinuum numerical methods, which have been effectively employed for the block-based modelling of full-scale masonry structures, such as masonry arch bridges, temples and churches.

Of the most diffused discontinuum methods employed is the Distinct Element Method (DEM)[2] which was initially developed for problems of sliding and crashing rocks (Cundall, 1971). The main advantage of the DEM, as any such discontinuum method, is that discontinuities can be explicitly implemented into the numerical model and also handled efficiently during the simulation. Another advantage is that, like the FEM, it can capture both the in-service and collapse behaviour of the masonry. However, at the present moment, the DEM also has limitations (Sarhosis *et al.*, 2016e), including:

- A high-computational cost (which is, however, lower than that of a FEM block-based model);
- Its inherent need for block-based geometric models;
- Its limited employment to academia only at the present moment.

Also, of major interest to the numerical modelling of masonry are the conditions which define the DEM (Cundall and Hart, 1992), which are:

a) Finite (e.g. large) displacements and block rotation and detachment can be followed in an evolutive analysis;
b) Formation of new contact can be accounted for;
c) Block detachment is permissible.

The first condition ensures that the complex failure mechanisms of masonry can be captured whilst the second condition that arbitrary damage and post-peak behaviours can be efficiently simulated without the need of predefinition (Sarhosis *et al.*, 2016e).

Here a significant aspect of the discontinuum methods such as the DEM is made apparent, which is the contact type. This may be either the “soft contact” approach (also termed force-displacement formulation) or “hard contact” one. Essentially, the soft contact means that for two given deformable blocks, interpenetration is permitted by employing the assumption of elasticity to derive the normal stiffness. Conversely, hard contact implies that only shear

[2] The abbreviation DEM will herein be used interchangeably for both Discrete Element Method and Distinct Element Method.

movement and opening can occur (Cundall and Hart, 1992). According to Lemos (2007), the soft contact approach is preferable for the masonry where the shear and sliding forces significantly influence contacts forces in masonry structures.

DEM models are the most diffused discontinuum method employed in masonry, most likely owing to two commercial software packages, *UDEC* and *3DEC* (Itasca, 2019a; Itasca, 2019c). It is important to note that other software also implements the DEM, such as the commercial software package *PFC* (Itasca, 2019b), also developed by Cundall (Cundall and Strack, 1980) and the open-source code *YADE* (Šmilauer *et al.*, 2010). As opposed to *UDEC* and *3DEC*, both employ spherical elements and have seldomly been employed for masonry.

Of the earliest studies with the DEM on masonry was by Dialer (1992) to investigate the shear strength of wall panels within *UDEC*. In recent years, extensive research has followed on masonry wall panels. For instance, many studies have focused on the methodical definition of material properties employing both optimisation and stochastic methods (Sarhosis and Sheng, 2014; Sarhosis *et al.*, 2020). Another innovating contribution paved the way for the detailed block-based modelling within the DEM (Sarhosis and Lemos, 2018) by employing with Voronoi blocks to explicitly represent the mortar. It is of interest to note that this was also extended to 3D Voronoi blocks in *3DEC* (Pulatsu *et al.*, 2019b).

Given the DEM's effectivity and efficiency, it has also been widespread for masonry arch bridges. Lemos (1995) was of the earliest to carry out a 3D structural analysis of full-scale, masonry arch bridge in *3DEC*. In a later study (Jiang and Esaki, 2002), the influence of weakened material properties of an ancient bridge in Japan was assessed. Many more studies have followed on masonry arch bridges, however more recently, the research community has focused its concerns on addressing the complex soil-structure and spandrel wall behaviour of masonry arch bridges. For this aim, Sarhosis *et al.* (2019) employed Voronoi blocks to represent the soil, finding a good agreement with experimental studies. Finally, Forgács *et al.* (2019) also addressed the complex failure mechanisms of spandrel walls.

The DEM has also been extensively employed for the dynamic analysis behaviour of ancient temples and colonnades. Psycharis *et al.* (2003) carried out a 3D dynamic analysis of part of the ancient Acropolis in Athens with *3DEC*. Through this research, potential remediation strategies by reinforcing the temple with steel were considered. In the same spirit, many more investigations followed on ancient temples, incorporating more complex geometrical details and arbitrary loading scenarios (Psycharis *et al.*, 2013; Stefanou *et al.*, 2015; Bui *et al.*, 2017; Tavafi *et al.*, 2019). Finally, plenty of other studies have also focused solely on the dynamic behaviour colonnades such as in (Sarhosis *et al.*, 2016a; Sarhosis *et al.*, 2016b; Pulatsu *et al.*, 2017).

Non-smooth contact dynamics (NSCD) models

Whilst the DEM paved the way for employment of discontinuum numerical methods, other models have also been developed with time. One such class of models belong to the NSCD, developed by Moreau (1988) which employs hard contacts and implicit time integration. In comparison with the DEM, the NSCD is advantageous in that it does not employ fictitious numerical damping (Moreau, 1988) while its implicit time integration method permits unconditionally stable solutions with larger time steps. The NSCD has been employed for masonry, however in significantly fewer studies than the DEM which could be attributed to the fact that solely one code, *LMC90* implements it (Dubois and Jean, 2003). Chetouane *et al.* (2005) were of the first to employ the NSCD for masonry, whilst applying the *LMC90* for the 2D structural analysis of both masonry panels and arch bridges. Amongst others, the study demonstrated the performance of the NSCD for capturing the structural behaviour of masonry in a manageable time. In another publication (Rafiee *et al.*, 2008), a 3D dynamic analysis was carried out on the Arles aqueduct in the south-east of France with *LMC90*. Further than demonstrating both the potential and efficiency of the NSCD, this pioneering study is still of a handful of full-scale dynamic analyses of masonry arch bridges in literature. Recently, the NSCD has also been particularly attractive to investigators on historic churches and domes. Beatini *et al.* (2019) implemented the NCSD within a custom-built software to assess Brunelleschi’s dome in Florence, Italy. Since the actual geometry of the dome is hidden from view, various scenarios of geometric model were considered of, octagonal and circular domes with varying bond patterns developed by a parametric function. The NCSD was demonstrated as highly-effective for efficiently capturing the full-scale structures behaviour.

Discontinuous deformation analysis (DDA) models

Another class of models belong to the DDA, which was developed by Shi and Goodman (Shi and Goodman, 1985; Shi and Goodman, 1989), again adopting a hard contact, implicit time integration. In comparison with the DEM, the DDA is advantageous due to its implicit time integration method, and compatibility with the FEM. Thavalingam *et al.* (2001) pioneered the employment of the DDA (as well as employing *PFC*) for the 2D quasi-static analysis of an experimental arch. Within the study, the load-displacement responses of the DDA showed good agreement with the experimental, whilst outperforming a commercial FEM code, *DIANA* (TNO DIANA BV. Delft, 2020). In recent years, Perez-Aparicio *et al.* (2013) also analysed 2D arches within the DDA to examine the influence of load, voussoir number (i.e. block number) and arch embankment (i.e. the thickness of the fill above the arch crown's height). Finally, in another recent study, the DDA has also been applied to investigate the boulder impact on masonry structure in mountainous areas (Liu *et al.*, 2018). In the specific study, the DDA enabled the 3D dynamic analysis of various scenarios of boulder velocity and construction type (i.e. masonry bond) to propose remediation strategies for masonry structures at risk.

Finite discrete element method (FDEM) models

A final class discontinuum models belong to the FDEM, developed by Munjiza *et al.* (1995) employing a hard contact, implicit time integration method. Here the superiority of the FDEM is made apparent in that it is only discontinuum method that permits masonry crushing (i.e. blocks that can break and separated without the use of zero-thickness interfaces) whilst also sharing the advantages of the DDA and NSCD, in comparison to the DEM. Owen and co-workers (Owen *et al.*, 1998) were of the earliest to demonstrate the potential of the FDEM with the quasi-static 2D analysis of a full-scale masonry arch bridges whilst assessing reinforcement strategies (so-called *CINTEC* system). Amongst others, one of the model's novelties included the coupling spherical and polyhedral elements. Another FDEM investigation involving masonry arches regarded the modelling of arch reinforcements (Smoljanovic *et al.*, 2015). Further, than masonry arches, the FDEM has also found application in the context of cultural heritage masonry structures. Indicatively, Smoljanovic *et al.* (2013a) carried out a 2D dynamic analysis of the Prothyron in Split, Croatia. The models demonstrated the capabilities of the FDEM to capture extremely complex failure modes including cracking of the stone units, something unprecedented in DEM models. Finally, as for most numerical methods within the DEM, the FDEM has been particularly attractive for the research of out-of-plane seismic loading of URM, such as (Smoljanovic *et al.*, 2018).

2.2.3 The effect of geometric uncertainty of ad-hoc geometric models

Out of all the previously reported studies, the suitability of the DEM for carrying out a high-level structural analysis of masonry is apparent from the multitude of studies. Of particular interest to this study is the fact the DEM, as all the discontinuum methods, is extremely effective for the block-based modelling of large-scale masonry structures, something which only a handful of other models (even the latest FEM block-based) can achieve. However, a major shortcoming is also made evident here, which is inherent in block-based models. Specifically, all the previous investigations, which regarded state-of-the-art studies employed simplified geometric models (either ad-hoc or idealised). This is carried out to simplify the numerical modelling approach since block-based geometric model development is of labour-intensive manual procedures. Consequently of this, geometric uncertainty is introduced into the geometric model, since simplified geometric models do not accurately represent the masonry structure can be transferred into the structural analysis itself.

Indeed, early evidence (Heyman, 1969) has shown that geometric changes in masonry structures can greatly influence their mechanical response. Following this intuition of Heyman, numerous investigations, specifically within the DEM have also shown that variation in the geometry on a block-based level (i.e. joint inclination, block size and bond pattern) causes significant differences in the predicted structural behaviour, as summarised in Table 2-1. Notably, Szakály *et al.* (2016) demonstrated that the variance in masonry pattern yielded significant influence on both failure mode and collapse load of the masonry wall panels. In another study (Godio *et al.*, 2018), the arrangement and size of blocks were found to significantly influence the capacity for out-of-plane seismic loading of wall panels. Forgács *et al.* (2018), also found similar findings when the construction methods of masonry arches were found to significantly influence the collapse load and failure mode of arches. Finally, in a more recent study (Napolitano and Glisic, 2019), the influence of the block pattern was found to significantly influence the structural capacity.

Table 2-1: The effect of geometric uncertainty of ad-hoc geometric models on structural behaviour.

Study	Geometrical properties investigated	Structural behaviour indices investigated	Main findings
(Szakály *et al.*, 2016)	Masonry wall patterns	Shear resistance of masonry wall due to horizontal point load	Vertical bricks affect shear resistance for low confining vertical loads
(Godio *et al.*, 2018)	Block size, bed joint orientation	Collapse load due to horizontal gravitational load	The larger the blocks, the higher the structural capacity. The bed joint angle also influences structural capacity
(Forgacs *et al.*, 2017)	Block size, block length-to-width ratio (L/H)	Stability due to self-load	The larger the blocks (the larger the L/W for constant W), the more stable the arch is
(Forgács *et al.*, 2018)	Masonry arch construction methods	Collapse load due to vertical point load	The method of construction (false skew, helicoidal or logarithmic) influenced the structural capacity of masonry arches
(Pulatsu *et al.*, 2018)	Masonry arch block bond pattern and layer number (ring number)	Collapse load due to vertical point load	Double-layer arch had a lower structural capacity of the respective single. Bond pattern of arch voussoirs was not influential on structural capacity
(Napolitano and Glisic, 2019)	Bonding course of masonry walls	Maximum vertical displacement and principal stress due to settlement	The consideration of bonding courses increased structural capacity for settlement

From all the studies of Table 2-1, it is evident that the employment of simplified (i.e. ad-hoc or idealised) geometric models, can significantly compromise the robustness of the structural analysis. The reasons why the majority of state-of-the-art DEM studies employ simplified geometric models are given to following difficulties: a) geometric data acquisition, i.e. obtaining the exact location of each block and joint (geometry on a block-based level) manually is an extremely laborious procedure, and potentially prohibitive for the case of a large-scale structure; b) accurate geometric model development is extremely complex (i.e. the block-by-block procedure (Sarhosis *et al.*, 2016e) in comparison with the user-friendly continuum-based FEM models), while methodical and automated frameworks for such task lack; and c) comprehensive investigation still lacks on the effect of geometric uncertainty to justify the employment of accurate geometric models.

2.3 Structural surveying of masonry structures for numerical modelling

As found in Section 2.2.3, one reason why state-of-the-art DEM studies employ simplified geometric models is the difficulty of geometric data acquisition, owing to the employment of manual measurements (i.e. traditional geospatial techniques such as direct measurement with a tape). Nowadays, however, geospatial techniques such as non-contact sensing have revolutionised many applications of civil engineering, including numerical modelling (Tang *et al.*, 2007; Chen, 2012; Olsen and Kayen, 2012; Vosselman and Maas, 2014; Ye *et al.*, 2018). Of particular interest is the SfM photogrammetry pipeline which further than practical, is a considerably accurate and low-cost non-contact sensing technique (Dai and Lu, 2010). Following this intuition, this Section examines the suitability of SfM photogrammetry in comparison to other geospatial techniques in its suitability for providing geometric data rapidly and reliably. It is noteworthy that the presented geospatial techniques are classified by the author into: point-based techniques, which provide discrete points only; and cloud-based, which provide points clouds and orthoimages.

2.3.1 Point-based techniques

Total station

A total station consists of an electronic theodolite combined with an electronic distance measurement. Through the recording of angles and points, the accurate 3D positions of discrete points are obtained. In the context of numerical modelling of masonry structures, total stations are employed to directly measure the structure (e.g. the blocks and joints positions) and develop a geometric model according to a given modelling strategy (e.g. with the point-based approaches of Section 2.4) as well as to provide control information for geospatial techniques. The main advantages of the total station are its simplicity of use, and sub-cm accuracy in each direction (Morer *et al.*, 2013) which indeed makes it particularly attractive for providing control information (i.e. georeferencing ground control points), as will be demonstrated in this investigation. However, the main limitation of the total station is owed its relatively high cost and to the nature of its observations (i.e. discrete points). This makes its employment for the numerical modelling of large-scale structures costly and time-prohibitive due to the impractical and laborious task of measuring an unmanageable number of blocks.

Laser tracker

The laser tracker is a recent technology used in large scale precision manufacturing such as aerospace and the automotive industry (Estler *et al.*, 2002). It is similar to the total station in that it is placed upon a tripod, is pointed at targets in sequence, and measures the distance to each, as well as the angles between each pair. From this raw data, full 3D coordinates of each target can be calculated. Like a robotic total station, the tracker can move itself to find the centre of the target. The tracker measures the position of a retro-reflective prism, which rather than a traditional target, is mounted in nests, permanently fixed to the structure. The prism is set in a stainless-steel sphere, such that the measurement point is at the centre of the sphere with extremely high accuracy. The nests are designed as such so that the sphere sits on three points and is held in place by a magnet, ensuring repeatability of the measurements. As is surveying with a total station, it is not possible to see all the measurement points from one instrument position which requires measurements from several positions are combined into a complete survey.

While the application of laser trackers is still limited in masonry structures, some pioneering investigations do exist. In one recent study, the employment of laser trackers was carried out (Barazzetti *et al.*, 2015b) to detect the static movement of the column the Cathedral of Milan. The achieved precision which was 0.1 mm, clearly demonstrated the performance of the laser tracker in such applications. In another study (Yang and Xu, 2019), laser tracking was employed for providing control information to a TLS survey of a concrete bridge. Specifically, parameters of a B-spline model developed from the TLS point cloud were calibrated and validated surface accuracy of about 0.1 mm. These studies both demonstrate the main advantage of the laser tracker which is its accuracy, which is invaluable for providing control information. However, as the total station, it does not appear advantageous for geometric model development due to providing discrete points.

2.3.2 Cloud-based techniques

Airbourne and terrestrial laser scanning

At the present moment, laser scanning is one of the most important non-contact sensing techniques for the structural surveying of masonry structures (Tobiasz *et al.*, 2019). For the so-called time-of-flight type laser scanners which are more pertinent to this investigation, pulses of light are emitted from the scanner's position to the masonry structure's surface whilst distance measurement results from recording the time interval between light emission and return (Baltsavias, 1999). Typical laser scanner components are a rotary mirror, a laser source, and a data storage module (Tobiasz *et al.*, 2019). Whilst laser scanning provides a number of returns, the main product concerning numerical modelling is a dense point cloud. Apart from 3D positions, this also includes a fourth parameter; the intensity of the returning signal, which is particularly useful for characterizing the scanned material (Tobiasz *et al.*, 2019).

Laser scanning can be distinguished based upon the platform in which it is employed. When the laser scanning is carried out from the ground, it is TLS. Whilst, when airborne platforms are employed (such as an unmanned aerial vehicle), it is airbourne laser scanning (ALS). In the context of this investigation, TLS is more pertinent due to the scale and required accuracy of the problem. Additionally, the errors of TLS according to Tobiasz *et al.* (2019) are summarised as: a) internal, such as instrumental errors, laser beam errors (propagation, reflection, and refraction); and b) external errors, such as the case of the material colour affecting the intensity, and material translucency.

The main advantage of laser scanning, in comparison with all the aforementioned point-based techniques, is the rapid geometric data acquisition (e.g. M Pts level) and high-accuracy (e.g. sub-cm level), comparable to a total station (Vosselman and Maas, 2014). For this reason, TLS is the benchmark method of structural surveying of masonry structures, as will be demonstrated further on. However, the disadvantages of TLS are found in the high cost of equipment, the necessity of multiple scan stations when oblique incidence angles occur, and in the lack of textural information provided (Peppa, 2018).

UAV-based and terrestrial photogrammetry

Photogrammetry is another commonly employed non-contact sensing technique (Tobiasz *et al.*, 2019) that deals with extracting geometric data from imagery (Wolf *et al.*, 2014). As with TLS, photogrammetry may by terrestrial or airbourne according to the platform of the employed sensors. In the past, high-quality analogues metric cameras were employed in conventional photogrammetry such as in Mills and Barber (2004). Now, photogrammetry is also employed with digital cameras in combination with low-cost SfM platforms (the formulation of which will be detailed in the forthcoming paragraphs).

In stereo photogrammetry, which is more pertinent to this investigation, two optical rays, representing conjugate image points, ideally, intersect at an object point through a so-called spatial intersection. The establishment of the camera's internal geometry is termed interior orientation carried out by defining the interior orientation parameters (IOPs). These geometric parameters, also reported in the literature as the inner, intrinsic orientation or camera intrinsics (Luhmann et al., 2006) are: a) the focal length, which is the distance between the lens centre and the lens focus point; b) the principal point (the intersection of fiducial lines); c) symmetrical radial lens; and d) the decentring distortion parameters. Relative orientation consists of the determination of the position and orientation between two images, relative to each other, resulting in the generation of a stereo model. Absolute orientation consists of defining the 3D position of control points of a stereo model in a desired coordinate system, via a 3D conformal coordinate transformation using at least two horizontal and three vertical control points. After absolute orientation, the camera's exterior orientation parameters (EOPs) are defined which are three translations and three rotations. Simultaneous multiple image orientation is determined by aerial triangulation whilst the establishment of the position and orientation of each bundle of the optical ray is termed bundle block adjustment. In the case of self-calibrating bundle adjustment such as in the software *Metashape*, re-optimisation of IOPs and EOPs are simultaneously in a least-squares sense. The interested reader is referred to Wolf *et al.* (2014).

The theoretical basis of photogrammetry is the so-called collinearity condition, as shown in equation (2.1), according to Dai and Lu (2010). According to this condition, any given optical ray (Figure 2-1) can be defined by three points: a) the image point; b) the camera perspective centre; and c) the object point. Moreover, any point of an image captured by a camera is the representation of the convergence of many optical rays (Historic England, 2017). Where f is the nominal focal length; (x_o, y_o) and (X_o, Y_o, Z_o) are the coordinates of the perspective centre in the image plane and object space, respectively; (x_n, y_n) and (X_n, Y_n, Z_n) are the coordinates of the n-th target in the image plane and object space, respectively; Λ is the scale factor and M is the rotation matrix.

$$\begin{bmatrix} x_n - x_o \\ y_n - y_o \\ -f \end{bmatrix} = \Lambda \times M \times \begin{bmatrix} X_n - X_o \\ Y_n - Y_o \\ Z_n - Z_o \end{bmatrix} \tag{2.1}$$

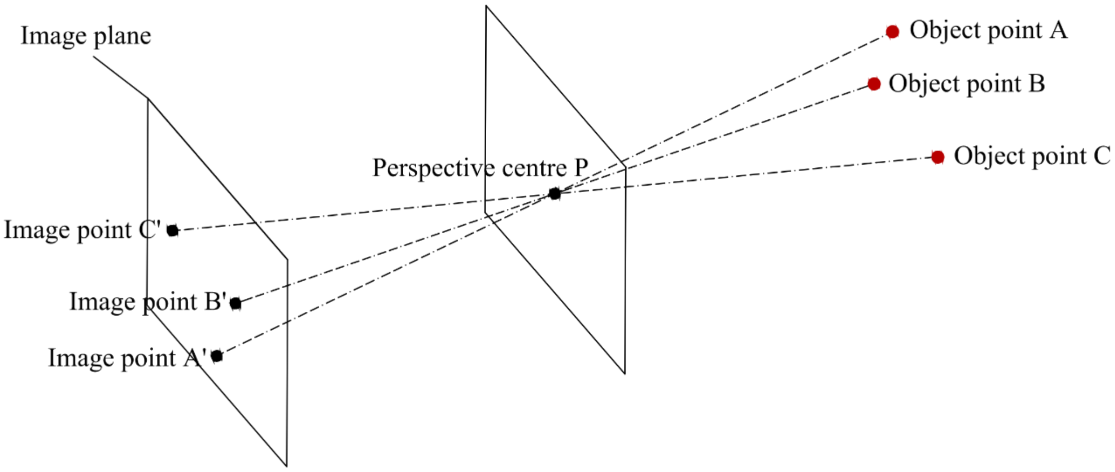


Figure 2-1: The optical rays for object points A, B and C.

SfM photogrammetry, which is of major interest to this investigation, is a recent addition to photogrammetry which has been widely employed for the structural surveying of masonry in the latest years. In comparison with TLS, it is advantageous due to its low-cost, facility of employment and the high quality of its returns (e.g. high-quality RGB orhtoimagary and point clouds). Additionally, SfM photogrammetry consists of three main phases which are: a) sparse point cloud reconstruction; b) georeferencing; and c) dense point cloud reconstruction. Sparse point cloud reconstruction regards the process of aligning acquired images with a process of automated feature detection and correspondence until all the photogrammetric block is oriented (Golparvar-Fard *et al.*, 2011). In specific, feature matching firstly is carried out, which effectively finds distinct features on each image, allowing for the automated matching across a subset of images. For example, a welluknown method of carrying this out is with the so-called Scale Invariant Feature Transform (SIFT) developed by Lowe (2004). Then, once feature detection has been carried out throughout the dataset, quantification of the detected features match in each image pair is carried out. The result of this process is a sparse point cloud which refers to a point cloud of tie points. Georeferencing regards the providing of control information for the scaling and orientation of sparse point cloud. This is commonly carried out with the use of ground control points (GCPs) in two ways: a) indirect georeferencing, in which the points are the result of surveying; and b) direct georeferencing, in which the obtained points are the actual camera positions (e.g. provided by GPS or RTK). The process of georeferencing consists of recalculation of both the camera's IOPs, EOPs resulting in a recalculated sparse point cloud coordinates which are in accordance with the control information provided. This is carried out in a least-squares bundle adjustment using the information as weighted in conjunction with the tie points. Finally, once the sparse point cloud is georeferenced, the dense cloud reconstruction follows by employing a pixel disparity calculation with area-based image matching. Thereafter, pixel back-projection and triangulation (i.e. via spatial intersection) follows, in which a 3D surface is formed via gradient-based and energy minimisation algorithms to avoid irregularities.

As a result of the aforementioned pipeline, the main product of SfM photogrammetry is a dense point cloud, which is RGB-coloured, being a significant advantage in comparison to TLS, (which normally doesn't have RGB). However, it is to be noted that SfM photogrammetry, as opposed to TLS, does not provide the intensity data of the surveyed structure. Further than a dense point cloud, of major interest to masonry structures is orthoimagery, which may be digital elevation models (DigEMs), orthophotos or orthomosaics. Specifically, a DigEM is a mathematical description of a 3D surface in which, each grid point represents a single elevation value (Aguilar *et al.*, 2005; Wolf *et al.*, 2014) whilst the data type of DigEM is a double array of square pixels with a uniform size (Wolf *et al.*, 2014). Moreover, an orthophoto is the result of the orthorectification of DigEM and represents a continuous image grid of a uniform scale (Wolf *et al.*, 2014). The so-called orthorectification regards describing an object in its true orthographic position through the collinearity condition. Finally, orthomosaics result from joining multiple orthophotos together.

Also, of major importance are the errors of SfM photogrammetry, which may be due to: a) image overlap; b) GCPs; and c) external factors. The following paragraphs detail each error type. Indeed, an important aspect of acquiring images of SfM photogrammetry is the relative overlap between consecutive images. A lack of overlap has been found to cause erroneous initial image alignments and consequent erroneous sparse point cloud reconstruction (e.g. discontinuities) according to (Harwin *et al.*, 2015; Dietrich, 2016). Whilst the obvious solution is though high overlap this increases the point determination redundancy (Haala and Rothermel, 2012), the computational burden can become unmanageable and thus requires consideration. Furthermore, GCP's can both decrease the systematic errors of the bundle adjustment and increase the photogrammetric abundancy (Wolf *et al.*, 2014; James *et al.*, 2017). The two main factors associated with effecting the accuracy of the end-product are the GPC layout and the geometrical accuracy of the measurement GPC itself. Concerning their layout, the importance of the existence of GCPs on the border of the surveyed object has been stated on many occasions (James and Robson, 2012; Eltner *et al.*, 2016). Concerning the metric accuracy of the GCPs, it has been stated that they should be measured with an accuracy three times higher than that of the expected result (Remondino *et al.*, 2014). Finally, errors can also be associated with external factors such as image surface texture, lighting, weather conditions and instability of the camera. These such factors have been attributed to affecting the SfM photogrammetry image matching algorithms (James and Robson, 2012; Remondino *et al.*, 2014) and thus causing errors.

Both terrestrial and UAV-based SfM photogrammetry are well-established techniques of structural surveying of masonry structures, as summarised in Table 2-2. Notably Bosché *et al.* (2015), surveyed walls of a historic masonry castle with the terrestrial SfM photogrammetry, yielding comparable results to TLS in terms of accuracy and point cloud density. Another study (Barrile *et al.*, 2015) investigated the performance of various SfM photogrammetry pipeline software types against a reference TLS point cloud. Of all the point clouds were generated from 219 images and three software types, *Metashape* presented the best agreement with the TLS data. In the context of the deformation analysis of historic masonry tower (Teza *et al.*, 2016), terrestrial SfM photogrammetry was also found to yield comparable results with the TLS, with errors in the range of 5-20 mm. Finally, the deformation analysis of masonry arch bridges (Soni *et al.*, 2015) also demonstrated the effectivity and accuracy of SfM photogrammetry.

As for many other applications of civil engineering, the use of unmanned aerial vehicles (UAVs) has also grown immensely in the past decade, especially for the structural surveying of cultural heritage masonry structures. For instance, Bosché *et al.* (2015) compared the outputs of UAV-based SfM photogrammetry against those of TLS and terrestrial SfM photogrammetry for the survey of a historic castle. In this case, the UAV-based SfM photogrammetry was found to be disadvantageous, possibly due to the anteriority of the approach which lacked a methodically pre-defined flight path design. In recent years, Barrile *et al.* (2017) assessed the

UAV-based SfM photogrammetry against TLS finding that the data acquisition faster, more flexible, and cost-effective but dependant on uncontrollable conditions such as weather and lighting.

Due to the lack of accessibility and highly irregular geometries, UAV-based SfM photogrammetry is increasingly favoured for the surveying of masonry arch bridges. For instance, one study (Bruno *et al.*, 2019), combined UAV-based SfM photogrammetry and TLS for the 3D documentation of a historic bridge in Italy, leading to highly-detailed and accurate surveying, which also included textural information due to inclusion of SfM photogrammetry. In another innovating study, Pepe *et al.* (2019) captured nadir images of the intrados of a masonry arch bridge by mounting a camera (in specific a smartphone) on top of the UAV. This low-cost approach also resulted in an accurate and detailed structural surveying. Finally, in a recent study (Chen *et al.*, 2019), structural surveying of a historic aqueduct was carried out with UAV-based SfM photogrammetry to assess the performance of consumer-grade UAVs for bridge inspection. The study demonstrated that the UAV-based SfM photogrammetry was easier to apply and more cost-effective than TLS. However, problems arose, regarding non-covered areas (e.g. railings), high noise levels and low geometrical accuracy persisted (cm-level compared to the mm-level of the TLS).

Table 2-2: Structural surveying of masonry structures with the terrestrial and UAV-based SfM photogrammetry.

	Study	Application	Structure	Software	Image #	Dense point cloud # (M pts)	Reported GCP error (cm)	Camera
Terrestrial SfM photogrammetry	(Soni *et al.*, 2015)	Deformation monitoring	MAB	*Visual SfM*	-	-	0.1	Nikon D3200
	(Bosché *et al.*, 2015)	3D documentation	CH	*Metashape*	260	79	3	Nikon D810
	(Barrile *et al.*, 2015)	3D documentation	CH	*Metashape*	219	28.9	2	Samsung model PL20
	(Teza *et al.*, 2016)	Deformation monitoring	CH	*Metashape*	156	14.9	0.5-2	Nikon D300S
UAV-based SfM photogrammetry	(Bosché *et al.*, 2015)	3D documentation	CH	*Metashape*	460	34	3	LC Sony Alpha-7R
	(Bruno *et al.*, 2019)	3D documentation	MAB	*Metashape*	610	-	-	DJI Phantom 4 (20 Megapixel)
	(Pepe *et al.*, 2019)	3D documentation	MAB	*Metashape*	768	10	0.7	Xiaomi Mi Drone 4K UHD WiFi FPV
	(Chen *et al.*, 2019)	3D documentation	MAB	*Metashape*	295	-	-	DJI Phantom 4 (12 Megapixel)

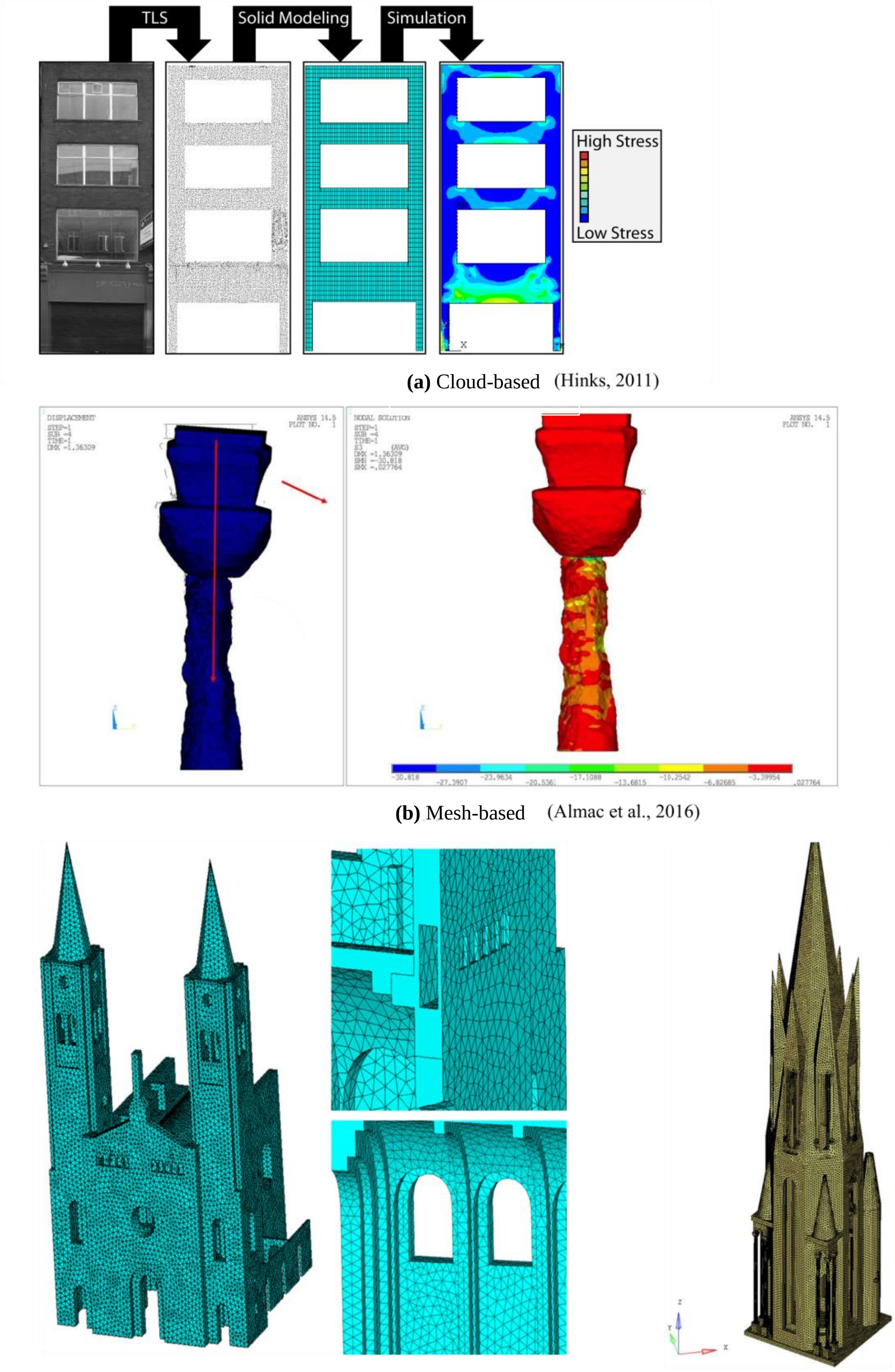
From the summarised studies in Table 2-2, it is evident that SfM photogrammetry can provide accurate geometric data (cm-level) and rapidly (M-Pts) which is comparable to a benchmark geospatial technique such as a total station or laser scanner. Furthermore, SfM photogrammetry can be cost-effective due to the potential of employing of consumer-grade digital cameras. Finally, it has been demonstrated as straightforward, which can be employed by on-site engineers with available cameras (i.e. such as a smartphone) replacing the necessity to purposely carry survey-grade equipment (Kim *et al.*, 2019).

2.4 Geometric model development approaches

As found in Section 2.2.3, another reason for which the majority of state-of-the-art DEM studies employ simplified geometric models is the complex block-by-block geometric model development, inherent to the DEM. Indicatively, it has been found that for FEM continuum models, (which are significantly simpler than block-based), geometric model developed consume up to 80% of the total modelling time (Zhang, 2013). To overcome this difficulty, various approaches have been adopted to automatically convert the data of various geospatial techniques into geometric models, however mainly for numerical methods such as the FEM and LA. In the following paragraphs, such approaches are detailed. It is noteworthy that since few studies (Zhang, 2013; Riveiro *et al.*, 2020) exists on this relatively novel subject, the classification is proposed by the author specifically for masonry, distinguished according to the continuum and block-based modelling strategies of Section 2.1.2.

2.4.1 Geometric model development for continuum modelling

Within the continuum modelling strategy, the following approaches have been adopted for accurate geometric model development: a) cloud-based; b) mesh-based; c) NURBS-based; and d) BIM-based. The following paragraphs introduce both the approaches and relevant studies, as demonstrated in Figure 2-2.



(a) Cloud-based (Hinks, 2011)

(b) Mesh-based (Almac et al., 2016)

(c) NURBS-based (Sánchez-Aparicio et al., 2014)

(d) BIM-based (Rolin et al., 2019)

Figure 2-2: Geometric model development for continuum modelling.

Cloud-based approaches

With the cloud-based approach, a point cloud is directly converted into a geometric model, usually through spatial enumeration (Section 2.1.3), such as voxelization. The term voxelization describes the conversion of a masonry structure's geometric domain into an equivalent volumetric representation in form of cuboids (voxels). The main advantage of this approach is that the structure can be of any geometric form (i.e. non-watertight or non-convex), without the necessity of mesh generation, whilst the volumetric modelling is achieved directly with the voxels themselves.

As part of his doctoral investigation, Hinks (2011) presented a pioneering voxelization approach (shown in Figure 2-2a) for developing of geometric models of URM building facades. This was a novel point-based (i.e. employing point clouds) voxelization method based on volumetric subdivision rather than the previously applied methods of surface reconstruction (i.e. using meshes). Due to the anteriority of this work, the geometric models were only two-dimensional. Based on this pioneering study, various studies employing point cloud segmentation techniques improved the 2D geometric models of the façades (Linh *et al.*, 2012; Linh and Laefer, 2013; Linh and Laefer, 2014; Truong-Hong and Laefer, 2014; Iman Zolanvari and Laefer, 2016) which were all incorporated in FEM software. Castellazzi *et al.* (2015) further advanced the cloud-based approaches by developing the first three-dimensional geometric models. This was another instance of a point-based voxelization workflow which led to the full-scale FEM structural analysis of a historic masonry castle. The same particular been further employed with the structural analysis of other historic masonry structures (Bitelli *et al.*, 2016; Castellazzi *et al.*, 2017). Finally, more recently, Selvaggi *et al.* (2018) added a simplified process of geometrical assessment of the geometric models developed to the workflow.

Mesh-based approaches

The mesh-based approaches refer to converting a mesh[3] into a geometric model. Often, other processes precede a mesh-based approach such as watertight conversion and mesh simplification, for the structural analysis software to be able to handle a manageable amount of faces and vertices from the mesh (Riveiro *et al.*, 2020). After the surface of the structures is represented through the final mesh, it is volumetrically subdivided into either pyramidal or tetrahedral finite elements within either the structural analysis or a third-party software itself. The main advantage of the mesh-based approach is its simplicity in implication.

Due to its very simplicity, the mesh-based approach has been extensively applied within the context of masonry arch bridges. In an early study, Vatan and Arun (2005) developed a geometric model of an aqueduct with the mesh-based approach for structural analysis within the FEM. Later on, Arias *et al.* (2007) employed a mesh-based approach for geometric model development of a historic bridge for structural analysis within a FEM framework. This study innovatively combined data of ground-penetrating radar (GPR) to additionally determine the interior (fill) material of the bridge. Similarly, Lubowiecka *et al.* (2011) later on employed the mesh-based approach for a FEM structural analysis with a textured mesh from the SfM photogrammetry, which included the damaged areas of the bridge detected and marked, whilst GPR was again used to determine the fill of the bridge. A further study (Stavroulaki *et al.*, 2016) successfully added damage to the FEM mesh corresponding to cracks from the textured mesh of SfM photogrammetry. Finally, while all the previous studies regarded single-span masonry arch bridges, Conde *et al.* (2017) developed a pioneering 3D geometric model of a full-scale, multi-span bridge for FEM structural analysis.

The mesh-based approach has also been applied within the context of cultural heritage masonry structures. Notably, Pieraccini *et al.* (2014) developed FEM geometric models with the use of CAD of a historic tower. In another study (Meschini *et al.*, 2015), FEM analysis of a fortress was also carried out using a mesh from developing a geometric model from a TLS mesh. Barrile *et al.* (2016) also followed a similar approach, however employing terrestrial SfM

[3] Mesh herein refers to a surface mesh such as triangulated irregular network constructed from nodes of a dense point cloud and facets by Delaunay triangulation.

photogrammetry. Furthermore, Hacıefendioğlu and Maraş (2016) were of the first to employed UAV-based SfM photogrammetry to develop geometric models (FEM) of a mosque. D'Altri *et al.* (2018b) recently employed the mesh-based approach which enabled semi-automated structural analysis of a leaning tower with both FELA geometric models. Finally, more recently, Bassier *et al.* (2018) presented a mesh-based approach of which the contribution was the additional crack introduction tool, capable of adding cracks to the geometric model with manual intervention.

NURBS-based approaches

The non-uniform rational basis spline (NURBS) refers to an approach of approximating a complex geometry, to facilitate the handling of the geometric model within the structural analysis software while retaining a high degree of geometric accuracy (Riveiro *et al.*, 2020). The basis of the NURBS is the mathematical spline, a curve defined by multiple nodes (named control nodes) and polynomial functions. The simplest form of a spline is a line joining two control points. For n control points, the general rule is for polynomial function with a degree of n-1. Base Splines (B-Splines), are the subcategory of spline curves with the mathematical property of minimal support. Minimal support means that a linear combination of B-spline can be employed to express any spline function of the same degree. NURBS curve are common to B-Splines except to that, each control point has a weight; if weights were equal to 1, then the NURBS would be a B-spline. The result of a tensor product of two NURBS curves which originates from a quadrangular patch is a NURBS surface (patch). In this way, data of geospatial techniques can be used as control points for retopology in a NURBS approach. It's noteworthy that the NURBS-based approach belongs to the boundary representation method of geometric model development. The NURBS approach is particularly advantageous due to providing an accurate geometrical representation of the masonry structure while requiring less manual intervention (Riveiro *et al.*, 2020). However, as with the mesh-based approach, since it represents the masonries surface only, it must be volumetrically subdivided into either pyramidal or tetrahedral finite elements within the structural analysis software.

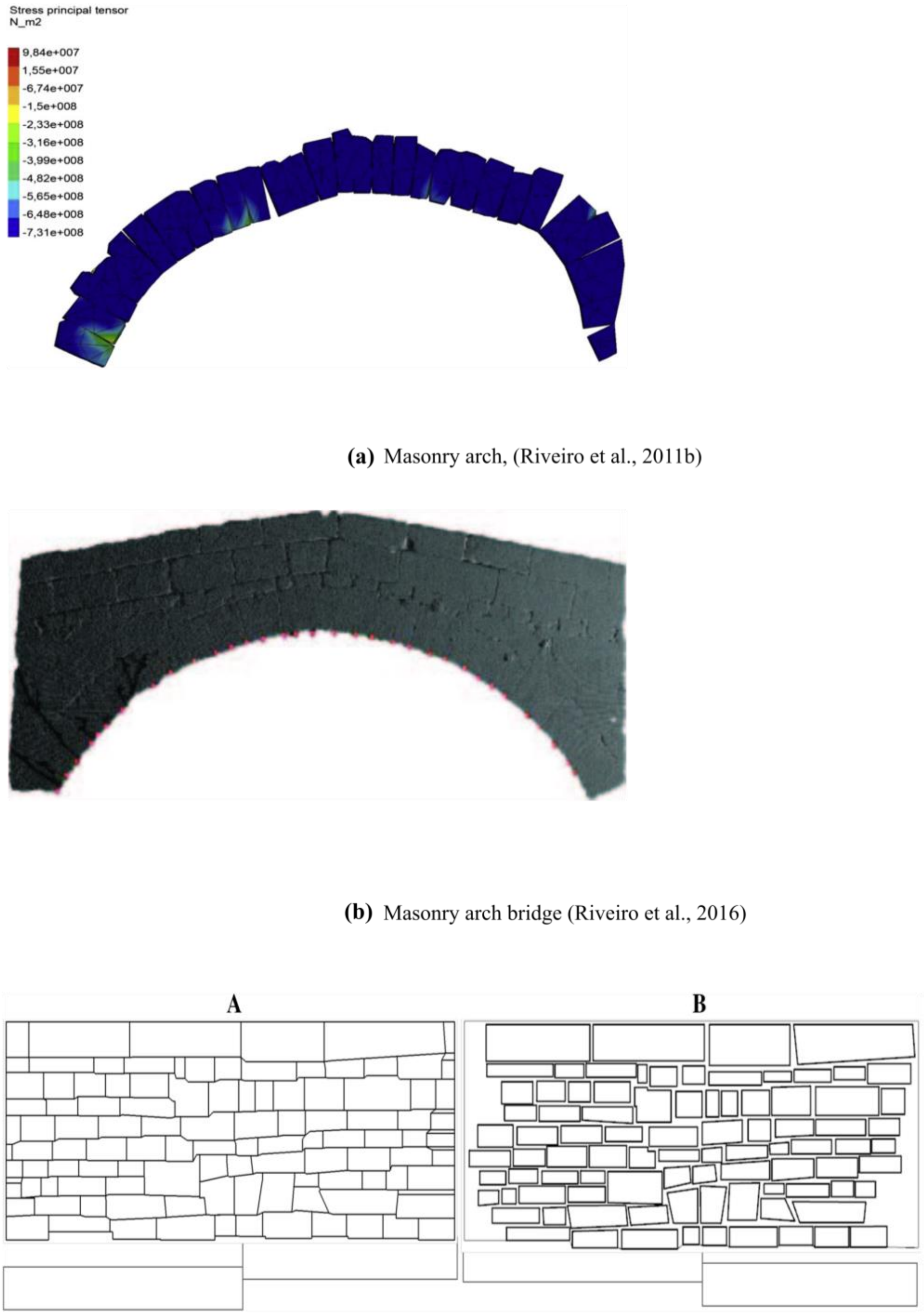
Tucci and Guardini (2014) proposed a procedure of developing geometric models from meshes using a NURBS-based approach. This was carried out by applying retopology within third-party software, in which the mesh was made compliant to NURBS generation. In another pioneering study (Sánchez-Aparicio *et al.*, 2014) the NURBS-based approach was used to develop a FEM geometric model (shown in Figure 2-2c) of a historic masonry church from the data of UAV-based SfM photogrammetry. The same research group (Sánchez-Aparicio *et al.*, 2016) carried out another structural analysis of a historic masonry structure with geometric model development carried out with the NURBS-based approach. Korumaz *et al.* (2017) carried out a structural analysis of a leaning minaret with the FEM with the NURBS-based approach based on TLS data. Vincenzi *et al.* (2019) used a NURBS approach to develop the geometric model of a historic tower employing combined UAV-based SfM photogrammetry and TLS data.

BIM-based approaches

Another way of developing a geometric model is by employing a building information model (BIM). In the context of construction, the BIM is defined according to Volk *et al.* (2014) as a shared digital representation of physical and functional characteristics of a given built object which forms a reliable basis for decisions. Though BIMs are most often employed solely for documentation, in some cases, BIM models have been directly converted into geometric models for subsequent structural analysis, though on few occasions as the other approaches. In the context of cultural heritage masonry structures for instances, a pioneering investigation involving the conversion of points clouds to BIMs and BIMs to geometric models for FEM structural analysis was presented (Barazzetti *et al.*, 2015a). It must be noted that to develop the complex geometry of the church within the BIM, generative NURBS profiles were used to obtain a rigorous geometric representation of the vault, while the simple shapes were used for regular sections of the building. These procedures were carried out in a manual CAD-based environment. In a more recent study (Rolin *et al.*, 2019), a slicing method of developing BIMs from point cloud was used (shown in Figure 2-2d) and then the BIMs were automatically converted into FEM geometric models. Whilst the BIM approach is easy to implement, it is disadvantageous due to the lack of automation since it requires the manual development of a BIM. Furthermore, as with the mesh-based approach, since it represents the masonries surface only, it must be volumetrically subdivided into either pyramidal or tetrahedral finite elements within the structural analysis software.

2.4.2 Geometric model development for block-based modelling

Within the block-based modelling strategy, the following approaches have been adopted for geometric model development: a) point-based; and b) image-based approaches. The following paragraphs introduce the aforementioned approaches, including relevant studies.



(a) Masonry arch, (Riveiro et al., 2011b)

(b) Masonry arch bridge (Riveiro et al., 2016)

(c) Masonry wall (Napolitano and Glisic, 2019)

Figure 2-3: Geometric model development for block-based modelling.

Point-based approaches

A point-based approach of geometric model development implies the employment of discrete points to develop a geometric model, in a block-by-block manner. This is carried out by use of manual CAD-based software to represent the structure using the measurements of point-based geospatial techniques such as a total station, or even direct measurement with a tape or a gauge. In a notable study, Morer *et al.* (2013) employed a total station to carry out numerical modelling of a masonry arch bridge, employing the block-based modelling strategy. The total station was placed in a suitable position to be able to scan all the desired target points levelled, and measurement commenced. In the specific study, the vertices of the masonry arch's voussoirs (i.e. blocks) were measured. Whilst point-based approaches can be adequate for small-scale structures of relatively few blocks (e.g. less than one hundred), a major disadvantage is laborious nature of both measuring discrete points and developing a geometric model from them within manual CAD-based design.

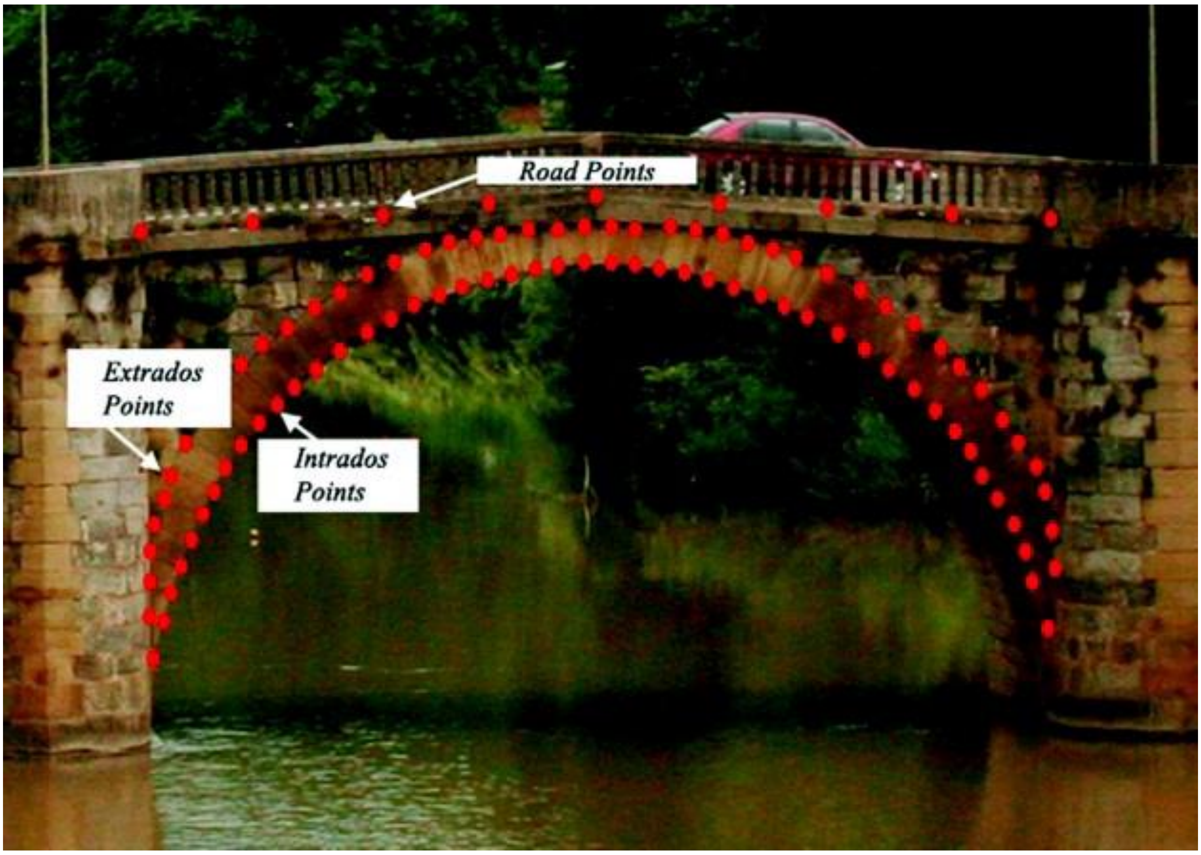


Figure 2-4: Application of the point-approach on a masonry arch bridge (Morer et al., 2013).

Image-based approaches

The image-based approach implies introducing an orthoimage of a masonry structure within a manual CAD-based framework and manually tracing the blocks and joints of the structure. Acary *et al.* (1999) pioneered the image-based approach with the structural analysis of a historic masonry structure from an orthorectified image with the NCSD. The pioneering study showed that the accurate geometric model led to a realistic failure mode of a full-scale building façade.

The approach has since then primarily found widespread application for masonry arch bridges. For instance, Morer *et al.* (2011) carried out the structural analysis of masonry arches with various LA approaches. Geometric models of various arches of a multi-span arch bridge were developed by manually extracting the contours of the arches from the orthoimagery of TLS. Here a limitation to the study is made apparent in that the voussoirs were not extracted from the orthoimagery, however, obtained roughly by dividing the arch into n segments. Later on, Riveiro *et al.* (2011) extended the approach, by accurately representing the arch blocks. Both FEM and LA models resulted from this study. Subsequently, Solla *et al.* (2012) also accurately represented arches with voussoirs from both SfM photogrammetry and GPR. However, in this innovating study, the authors defined the internal profiles of the arches from GPR data and compared various scenarios. A later study also employed GPR for together with orthoimagery to develop accurate geometric models of arches within the LA to investigate the influence of geometric uncertainty. In a final study, an array of geometric models was developed, among which with the image-based approach. Through the specific study, it was shown that whilst 2D block-based models and 3D continuum-based models showed good agreement for vertical loading, the 3D models (from mesh-based approach) were advantageous for complex loading cases which include transverse loading effects.

Furthermore, apart from masonry bridges, the image-based approach has recently found application within cultural heritage masonry structures, albeit on a smaller scale. For instance, Napolitano *et al.* (2019c), developed accurate DEM geometric models of a baptistery from SfM photogrammetry. In a follow-up study (Napolitano *et al.*, 2019b) the same research team also investigated the importance of accurate geometry through comparison of simplified and accurate geometric models respectively. The study demonstrated that accurate geometric models were indeed advantageous for capturing structural behaviour.

As can be observed from the aforementioned studies, the main advantage of the image-based approach is the nature of the geometric data (i.e. orthoimages). Especially in the case such as SfM photogrammetry, orthoimagery (especially orthomosaic) is more straightforwardly and rapidly attainable (as demonstrated in Section 2.3.2) in comparison to discrete points whilst it also contains textural information. However, commonly with the point-based approach, a disadvantage is the dependency on manual CAD-based block segmentation. It is notable that in an attempt to overcome this difficulty, various computer vision techniques have been applied for automated-block segmentation, though not yet explicitly for numerical modelling as will be demonstrated in Chapter 4. Another inherent limitation of image-based approaches, in general, is also made apparent here which is that they are limited to describing the structure in two-dimensions since they employ two-dimensional metric information (i.e. of the orthoimagery) and have a constant, user-assigned thickness in the transverse direction. Although evidence suggests (of Section 2.4.2) for the structural analysis of regular masonry structure such as bridges, 2D and 3D models agree in the absence of transverse loading, two consequences are associated with this limitation: a) only one layer of masonry is described in the geometric model (i.e. only the spandrel walls and arch of a masonry arch bridge); and b) the geometric models have planar faces, due to constant transverse coordinates.

The potential of computer vision for automating image-based approaches

Over the last decade, various investigations have demonstrated that computer vision techniques can be employed for automating the procedure of block segmentation. For instance, Sithole (Sithole, 2008) presented the first development of a deliberate methodology for brick segmentation with point cloud processing techniques. Later on, Willis *et al.* (2010) employed IPTs (ITPs) for estimating the shape of masonry elements present in the facade of a Gothic building from a single image based on automatically detected radiometric variations to separate individual stones the façade of a historic masonry church. For masonry/mortar detection, the theoretical background of this method was a watershed-based binary with the segmentation of the façade image into stones (black) and mortar (white) using a merge criterion based on colour similarity. Later on, Oses *et al.* (2014) also presented an IPT-based block segmentation method, based on the detection of mortar lines independent of conventional edge detection methods (e.g. Canny, Prewitt etc.). Specifically, to delineate the mortar lines, a framework was developed using fine-grained visual categorisation within the open-source computer vision library, *OpenCV* by extracting a set of straight-line segments. With a specific focus on numerical modelling however, without structural analysis, Riveiro *et al.* (2016) employed IPTs for block segmentation. Block segmentation was based on the intersection of the maximum intensity lines. On the other hand, Shen *et al.* (2016) also employed point cloud processing for block segmentation, through K-means clustering. Of the first investigations to successfully segment rubble masonry (Valero *et al.*, 2018) was based on the 2D continuous wavelet transform with an IPT framework. The same research team (Forster *et al.*, 2019) later extended this approach by using machine learning techniques, which make it of the most advanced and robust block segmentation methods reviewed (Figure 2-5), able to extract regular masonry or arbitrary shape without a high dependency on block-joint colour contrast such as in the case of IPTs. Finally, another recent study (Shen *et al.*, 2019) recently employed IPTs for block segmentation, entailing principal component analysis in combination with rectangle fitting.

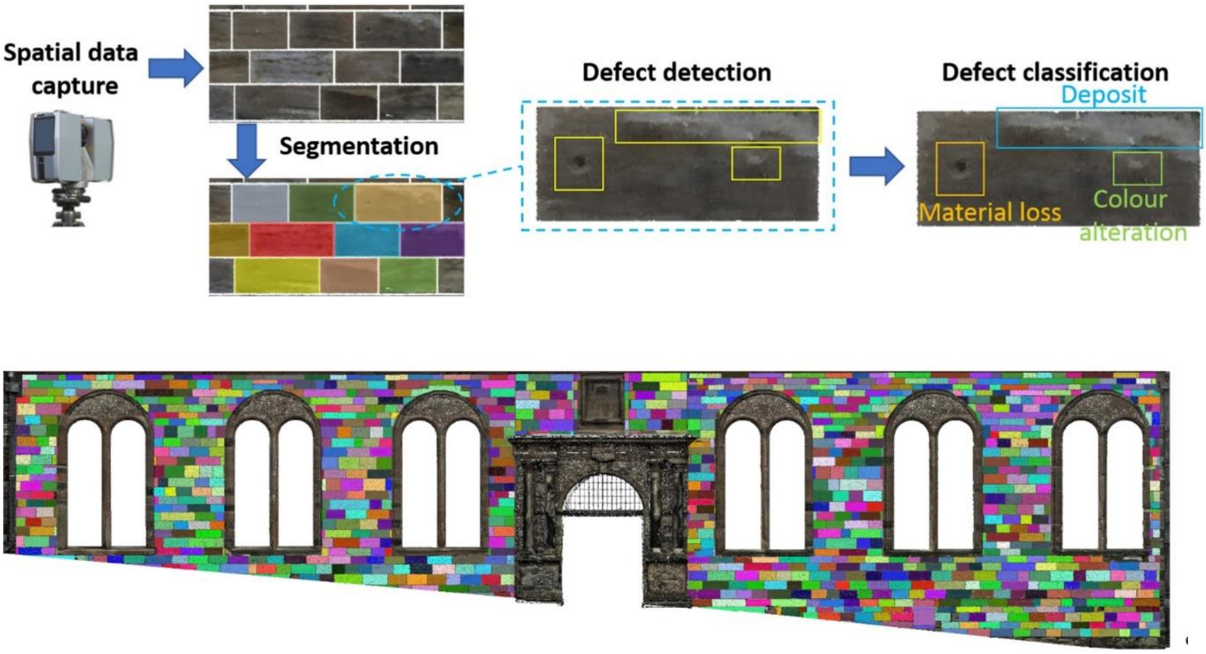


Figure 2-5: Automated block segmentation point clouds with machine learning techniques (Forster et al., 2019).

Despite the numerous investigations such as the previous that demonstrate the potential of employing computer vision for automated block segmentation, no study has yet evaluated the implementation of automated block segmentation specifically for structural analysis of regular masonry. Therefore, computer vision techniques have remained unexploited for the high-level numerical modelling of masonry such as the DEM.

2.5 Summary

This Chapter firstly evaluated the suitability of DEM in its potential for the high-level structural analysis of masonry structures. The DEM indeed was found to be of the most powerful tools of structural analysis of masonry available. However, it was also found that the majority of state-of-the-art DEM studies employ simplified geometric models (i.e. ad-hoc or idealised simplified). Consequently, the efficiency of such numerical modelling approaches is compromised due to the limitations of laborious manual measurements and procedures related to developing a geometric model. Furthermore, the fact that the reliability of the geometrical properties is neglected means that uncertainty is transferred into the structural analysis itself. The reasons for which state-of-the-art DEM studies neglect the employment of accurate geometric models were given to: a) difficulties in geometric data acquisition; b) the complex geometric model development of the DEM (i.e. block-by-block); and c) the lack of comprehensive investigation on the effect of geometric uncertainty to justify the employment of accurate geometric models in the first place. To overcome these difficulties, firstly, various geospatial techniques were reviewed, in their suitability for providing geometric data for numerical modelling rapidly and reliably. Non-contact sensing techniques were found to be attractive for this, particularly the SfM photogrammetry due to its low operational costs and straightforwardly approach. Then, various approaches of geometric model development were reviewed, of which the image-based and cloud-based appear particularly suitable for the DEM. Based on these findings, the proposed methodological framework is proposed in Chapter 3.

Chapter 3. Methodological framework

In light of the findings of Chapter 2, this Chapter proposes a three-stage methodological framework for the semi-automated and geometrically-accurate discrete element modelling of masonry structures. The first stage of the framework concerns structural surveying with various geospatial techniques. The second stage of the framework entails geometric model development from discrete points, orthoimage and point clouds. Specifically, three workflows are presented, the: point-based, image-based, and cloud-based respectively. The final stage entails numerical model development and structural analysis with the DEM. Stemming from the various geometric model development workflows, the methodological framework provides various approaches, as shown in Figure 3-1. These are the so-called, point-based, manual image-based, semi-automated image-based and cloud-based approaches, entitled *"Point2DEM", "manual Image2DEM", "semi-automated Image2DEM", and "Cloud2DEM"* respectively.

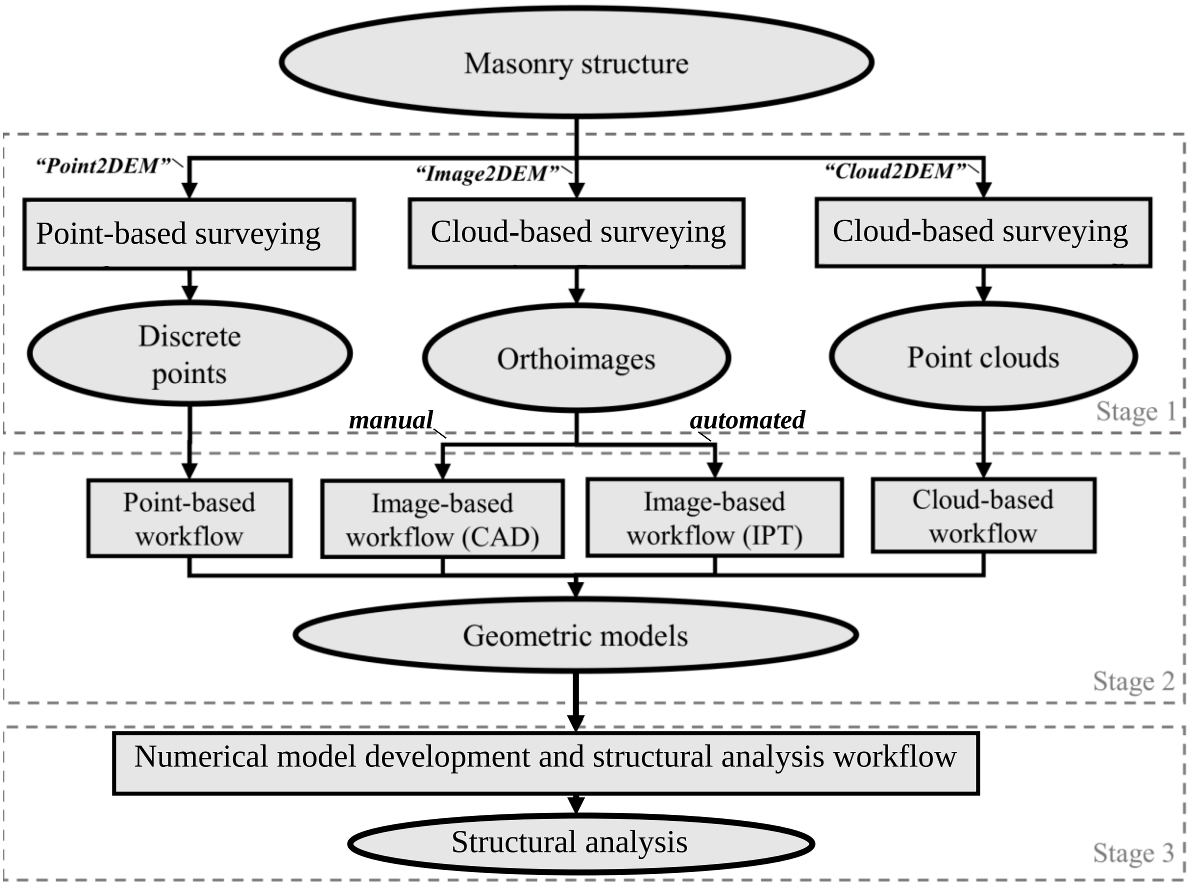


Figure 3-1: The methodological framework.

For the implementation of the methodological framework, a classification of masonry will be proposed, regardless of the material and according to the geometrical randomness solely, as such: a) regular masonry, in which masonry units are either orthogonal or regular-shaped; i.e. ashlars, bricks or voussoirs (Figure 3-2a); or b) irregular, in which masonry units are highly-irregular as in (Figure 3-2b). According to this classification, the point-based and image-based approaches regard block-based numerical modelling of regular masonry, in the spirit of the micro-modelling strategy of Lourenco (1996). Moreover, the cloud-based approach regards block-based numerical modelling of rubble masonry, however in the spirit of the macro-modelling strategy of Lourenco (1996).

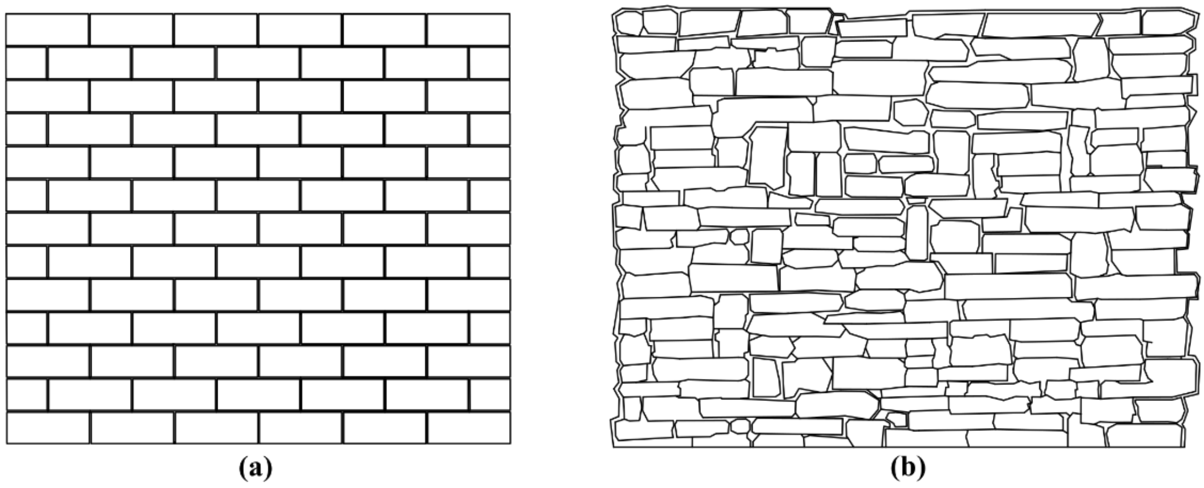


Figure 3-2: Classification of masonry into:(a) regular; and (b) rubble masonry (irrespective of material).

Moreover, it is important to note that the point-based approach is recommended for structures of regular masonry of a small scale (e.g. less than 500 blocks). On the contrary, the manual image-based workflow is recommended for large-scale structures of regular masonry, where the blocks are non-distinguishable with image processing techniques (a definition of distinguishable is provided in Section 3.2.2). Moreover, the semi-automated image-based workflow is recommended for large-scale structures of regular masonry, however of distinguishable blocks. Finally, the cloud-based is recommended for rubble since the anisotropy of the rubble cannot be determined reliable and thus it must be fictitiously assigned.

3.1 Stage 1: Structural surveying

The first stage of the three-stage methodological framework concerns the structural surveying of the masonry structure subject to structural analysis with various geospatial techniques. In this investigation, emphasis is given to the structural surveying with the SfM photogrammetry. As such, apart from the workflow of structural surveying, a Section is also included, explicitly detailing the reasoning for the selection of the employed software and its implementation of the SfM photogrammetry pipeline. Then, given that the methodological framework is employable with the geometric data of TLS, a workflow of structural surveying with TLS is also included. Finally, with regards to the point-based techniques, a workflow of structural surveying is not provided however the interested reader is referred to Morer *et al.* (2013) of Section 2.3.1, where the process of obtaining discrete points of masonry structure specifically for numerical modelling is detailed (with a total station).

3.1.1 The SfM photogrammetry pipeline in Metashape

Various low-cost commercial and open-source software that adopts the SfM photogrammetry were considered for this investigation. Table 3-1 reports some commonly found software within the context of the 3D documentation of masonry structures. The reasoning for employing *Metashape* in this investigation was: a) its user-friendly, quasi "black-box" workflow, which facilitates implementation; and b) it has a demonstrated effectivity due to its employment in vast the majority of investigations on masonry, as was demonstrated in Section 2.3.2.

Table 3-1: SfM photogrammetry software.

Software	Source	Type
Metashape	(Agisoft, 2019)	Commercial
ContextCapture	(Bentley, 2020)	Commercial
MicMac	(ENSG, 2020)	Open-source
Colmap	(Schonberger, 2020)	Open-source

PhotoModeler	(PhotoModelerTechnologies, 2019)	Commercial
PMVS2	(Furukawa and Ponce, 2007)	Open-source
VisualSfM	(Snavely, 2008)	Open-source

Due to its commercial nature, *Metashape* employs a black-box implementation of the SfM photogrammetry pipeline and thus its underlying processes are not explicitly described. However, previous research (Peppa, 2018) has highlighted the possible underlying functions of the sparse point cloud and dense point cloud reconstruction phases, as will be demonstrated below.

Sparse point cloud reconstruction

According to (Deseilligny and Clery, 2011; Fonstad *et al.*, 2013; Woodget *et al.*, 2015; Peppa, 2018), *Metashape* most probably employs a combination of the well-known SIFT-RANSAC algorithms for image alignment and sparse point cloud reconstruction. In SIFT (Lowe, 2004), multiple-level image processing is carried out to interest points by the detection of local extremes created by the difference-of-Gaussian smoothing functions. The invariance of the feature detection scheme of SIFT makes it a desirable choice.

Dense point cloud reconstruction

According to (Remondino *et al.*, 2014; Eltner and Schneider, 2015; Peppa, 2018), *Metashape* also most likely employs an SGM-like algorithm for dense point cloud reconstruction. The SGM algorithm was developed by Hirschmüller (2008) and as a main assumption has that for a stereo pair, neighbouring pixels are expected to have similar disparities. In the algorithm, all the pixels belonging to the epipolar line [4]are searched. The potential correspondent disparity is

[4] Epipolar geometry is equivalent to the coplanarity condition which is the plane formed in a stereo model by the intersection of two optical rays together with their baseline between their camera perspective centres.

based on so-called cost assignment; costs calculated from pixel value differences of nearest neighbours.

3.1.2 Structural surveying with SfM photogrammetry workflow

In the following paragraphs, the employment of the SfM photogrammetry pipeline within *Metashape* for the structural surveying of masonry structures is detailed, according to Figure 3-3.

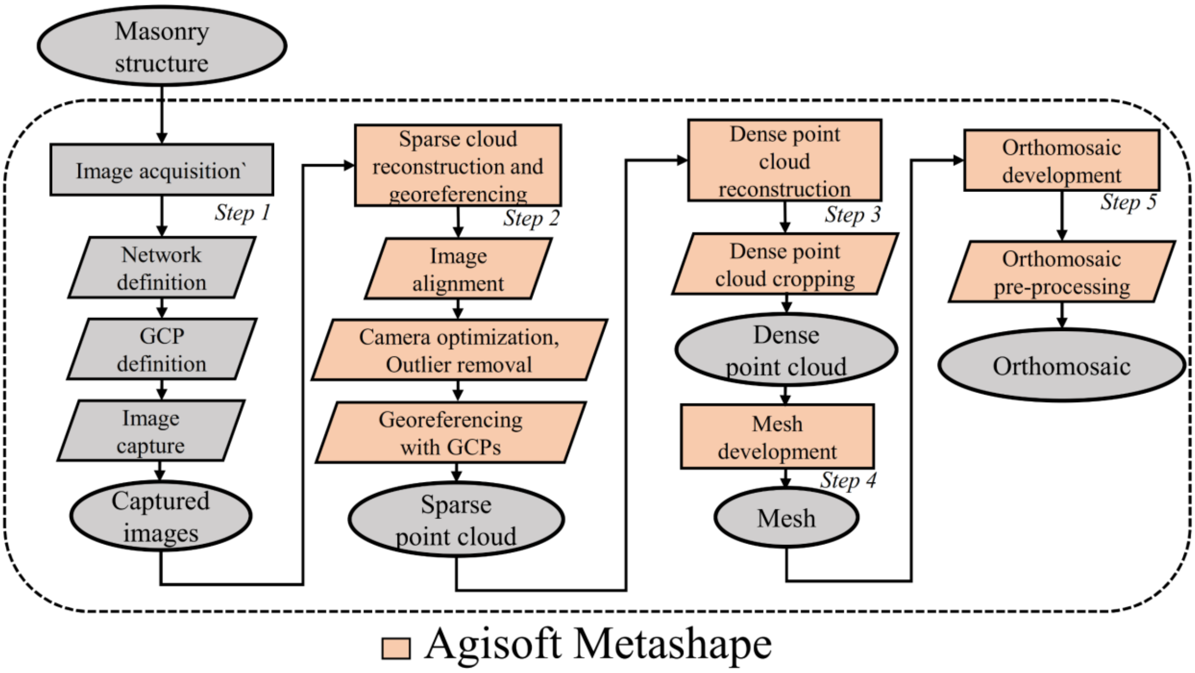


Figure 3-3: Structural surveying with SfM photogrammetry, employing Metashape.

Image acquisition

During step one, the images to be employed in the SfM photogrammetry pipeline are captured. This consists of defining the image capture network, the GCP definition and capturing the images. The following paragraphs describe each process.

<u>Network definition:</u> The network definition regards defining the locations at which the image shall be captured and is dependent upon the two factors: a) the inherent camera properties; and b) and the image network-dependent parameters. Concerning the inherent camera properties, two cameras were employed in this investigation, the first camera being a Sony IMX260 dual-pixel, integrated into Samsung Galaxy S7 smartphone (herein termed smartphone for abbreviation) whilst the second camera was a consumer-grade digital single-lens reflex (DSLR) Canon EOS6D camera (herein called DSLR for abbreviation). The reasoning behind the choice

of this equipment was to assess the framework's performance with both a low-cost and high-quality consumer-grade sensor, often available to structural engineers, in a real-world scenario. The inherent camera properties of both the cameras employed are reported in Table 3-2, where VSS is the vertical sensor size, HSS is the horizontal sensor size and f is the nominal focal length. Furthermore, the horizontal viewing angle of the lens (β_h) and vertical viewing angle of lens (β_v) are calculated (Chen *et al.*, 2019) from the following equation:

$$\beta_v = 2 \times \ tan^{-1}(\frac{VSS}{2 \times f}) \tag{3.1}$$

$$\beta_h = 2 \times \ tan^{-1}(\frac{HSS}{2 \times f}) \tag{3.2}$$

Table 3-2: Characteristics of the camera sensors.

Camera characteristic	Unit	Smartphone	DSLR
Horizontal sensor size (HSS)	mm	6.7	35.9
Vertical sensor size (VSS)	mm	5.5	24
Nominal focal length (f)	mm	4.2	24
Horizontal pixel number (HN)	Pixels	4032	6240
Vertical pixel number (VN)	Pixels	3024	4160
Horizontal viewing angle of lens (β_h)	°	89.0	89.0
Vertical viewing angle of lens (β_v)	°	68.9	65.2

Concerning the image network-dependent parameters, they are the camera's tilt angle (α_t) and distance from the structure, the so-called working distance (WD). Here the term, maximum ground sampling distance (GSD_{max}) is introduced, calculated from the following equation, according to Chen *et al.* (2019):

$$GSD_{max} = \frac{WD \times HSS \times cos(\beta_v)}{f \times HN \times cos(\alpha_t + \beta_v/2)} \tag{3.3}$$

In non-contact sensing, the GSD_{max} equals the distance between the centre of two consecutive pixels on the target surface and is a spatial resolution used to describe the quality of the produced orthoimagery. Effectively, the smaller GSD_{max}, the better the quality of the end-product of the SfM-photogrammetry is. Concurrently, the field-of-view (FOV) must also be taken into consideration while defining the image network, which effectively defines the quantity of the target surface area captured. This is also calculated according to Chen *et al.* (2019) as:

$$FOV = \frac{WD^2 \times HSS}{2 \times f} \times (\frac{cos(\beta_v/2)}{cos(\alpha_t - \beta_v/2)} + \frac{cos(\beta_v/2)}{cos(\alpha_t + \beta_v/2)}) \times (tan(\alpha_t + \beta_v/2) + tan(\alpha_t - \beta_v/2)) \tag{3.4}$$

The combination of the FOV and GSD_{max} finally determines the definition of the image network for a given WD and (α_t). Indicatively, to be able to identify suitable working distances during this investigation, various values GSD_{max} and FOV are plotted as a function of WD in Table 3-3 for α_t equal to zero.

Table 3-3: The variation of the field of view (FOV) and ground-sampling-distance (GSD_{max}) for the variation of the working distance (WD) for the employed camera sensors ($\alpha_t = 0°$).

WD m	Smartphone		DSLR	
	FOV (m^2)	GSD_{max} (mm)	FOV (m^2)	GSD_{max} (mm)
0.50	0.57	0.09	0.29	0.19
1.00	2.29	0.19	1.15	0.38
1.50	5.16	0.28	2.58	0.57
2.00	9.17	0.38	4.59	0.76
2.50	14.33	0.47	7.17	0.95
3.00	20.63	0.57	10.33	1.14
3.50	28.09	0.66	14.05	1.33
4.00	36.68	0.76	18.36	1.52
4.50	46.43	0.85	23.23	1.72
5.00	57.32	0.95	28.68	1.91

Ground control point (GCP) definition: For the scaling and orientation of SfM photogrammetry point cloud, the control information of this investigation was provided through ground control points (GCPs), using indirect georeferencing. GCPs were surveyed with a Leica MS60 total station (by intersection) with a mm-level accuracy. The definition of the GCP position is strategically carried based on two criteria: a) equal distribution of GCPs upon the surveyed structures; and b) GCPs with locations that facilitated their surveying with the total station in a minimal effort.

Image capture: The properties of the image capture were common with a previous doctoral investigation employing *Metashape* (Peppa, 2018). Specifically, fixed shutter speed and aperture were adopted while the ISO value was equal to 100 and the exposure interval (otherwise termed exposure value) was also equal to zero. Furthermore, the image overlap that produced at least a 60 % forward and a 60 % lateral overlap was considered suitable.

Sparse point cloud reconstruction and georeferencing

During step two, the image alignment is carried out, resulting in a sparse point cloud (tie points in 3D). *Metashape* provides multiple settings for image alignment of which the high alignment setting was chosen from a previous investigation (Peppa, 2018). After the image alignment, camera optimisation follows, together with outlier removal. For the outlier removal, the gradual selection tool is used to retain points with: a) a reconstruction uncertainty less than or equal to 10: b) a projection uncertainty less than or equal to 10; and c) reproduction error less than or equal to 1 (Peppa, 2018). For remaining outliers, manual cropping is carried out. Finally, the georeferencing of the sparse point cloud is carried out by introducing the GCPs of the total station in a text format.

Dense point cloud reconstruction

During step three, the dense point cloud is reconstructed. *Metashape* offers various options of smoothness and detail of which, the high-quality point cloud reconstruction setting in conjunction with the aggressive smoothness level were used (Peppa, 2018). After the dense point cloud is developed, the remaining noise is removed with the scissor tool of *Metashape*.

Orthomosaic development

During step four, the orthomosaic is developed with *Metashape*. The use of orthomosaics for geometric model development is considered as advantageous due to their longstanding use in masonry (Tsilimantou *et al.*, 2016; Chiabrando *et al.*, 2017). In *Metashape*, a mesh is developed based on the dense point cloud, and then orthomosaic development can follow with a user-defined resolution, termed $Res_{orth.}$. The selection of the mesh face number n_{faces} must be: a) large enough to accurately represent and accommodate the shape of the structures; yet b) small enough to not render the mesh computationally unmanageable. Furthermore, the selection of the orthomosaic resolution $Res_{orth.}$ is such that enables to clearly define the discontinuities of the structure in the orthomosaic, analogous to the GSD_{max} selection. This control is also carried out in *Metashape*. At this stage, a control the orthomosaic for noise, especially on the boundary of the surveyed structure is carried out. This is particularly important before employing the geometric model development workflows of Section 3.2, as to retain only structural members within the structural analysis.

3.1.3 Structural surveying with TLS workflow

In the following paragraphs, the workflow of structural surveying of masonry structures with TLS is detailed in a step-wise fashion according to Figure 3-4.

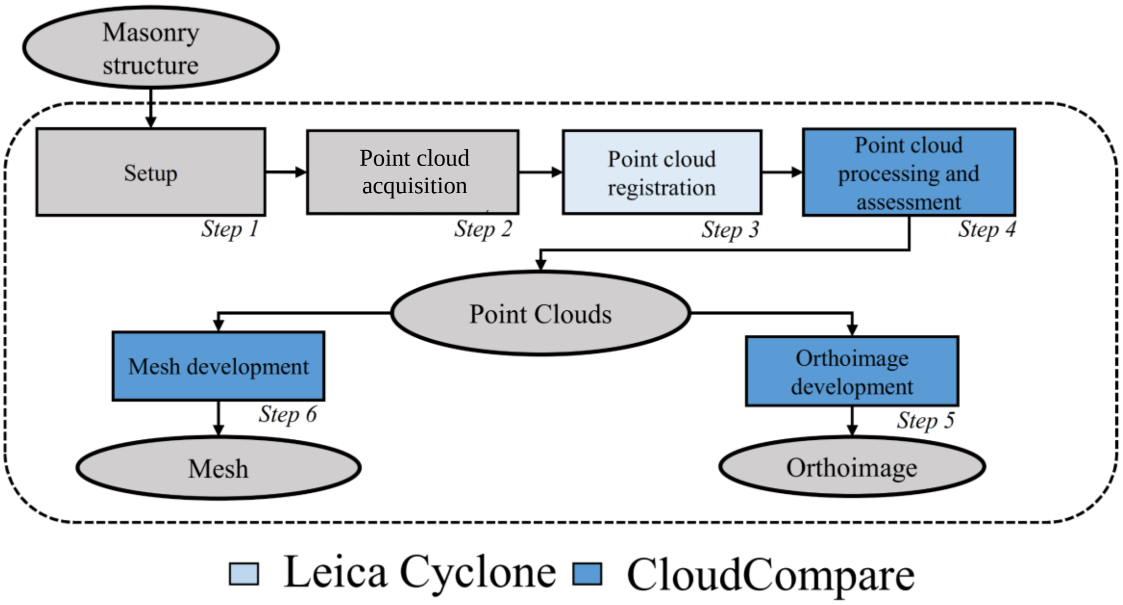


Figure 3-4: Structural surveying with TLS.

Setup

During step one, planning is carried out including preparation of the TLS equipment and the control points. In this investigation, a Leica P40 was considered, of which the properties are detailed in Table 3-4. The selection of this equipment was based on its common employment for masonry and consideration as a high-level laser scanner within recent studies (Chen *et al.*, 2019). Regarding the control points, the scanner's factory default 4.5 mm retroreflective targets were employed. It is notable that the control point placement must be strategically carried out so that for each scan station, there is an overlap with at least one other scan station of at least three control points.

Table 3-4: Characteristics of the Leica P40 scanner.

	Characteristic	Value
Accuracy	Range accuracy	1.2 mm + 10 ppm over the full range
	Angular accuracy	8" horizontal; 8" vertical
	3D position accuracy	3 mm at 50 m; 6 mm at 100 m
Distance measurement system	Scanning type:	Time-of-flight enhanced by Waveform Digitising (WFD) technology
	Wavelength	1550nm (invisible) / 658nm (visible)
	Laser class	1 (in accordance with IEC 60825:2014)
	Beam divergence	< 0.23mrad (FWHM, full angle)
	Beam diameter at the front window	≤ 3.5 mm (FWHM)
	Range and reflectivity	Minimum range of 0.4 m
	Scan rate	Minimum range of 0.4 m
	Range noise *	0.4 mm RMS at 10 m, 0.5 mm RMS at 50 m
	Field-of-view: horizontal/vertical	Hor. :360° Vert.: 290°
	Data storage capacity	256 GB internal solid-state drive (SSD) or external USB device
Imag. System	Resolution	4 megapixels per each 17°×17° colour image; 700 megapixels for panoramic image
	Pixel size	2.2 μm

Point cloud acquisition

During step two, the dense point cloud data is acquired by carrying out the TLS survey through various scan stations. Scan stations are carefully planned with the objective is to capture all the surfaces of the structure, including the control points in a manageable time. For this task, various examples exist from previous investigations on masonry structures, such as for a masonry arch bridge surveyed by Arias *et al.* (2010). After the TLS survey, the Leica P40 scanner's factory default software *Cyclone* version 9.1 (Leica, 2019) is employed for co-registering the TLS point cloud by which the various scan stations are aligned and joined, based on the retroreflective targets that are automatically detected within *Cyclone*. Here, the quality of the TLS is apparent in the RMSE alignment error of the targets. The accuracy of the TLS survey is the maximum RMSE error of all targets after co-registration, of which an acceptable value is considered to be equal to 5 mm or less than (Peppa, 2018).

Point cloud processing and assessment

During step three, the processing and assessment of the TLS point clouds are carried out with the open-source software *CloudCompare* (CloudCompare, 2019) which is useful to make the point clouds standardised and comparable. Firstly, to perform noise reduction, the Statistical Outlier Removal (SOR) tool (Radu Bogdan *et al.*, 2007) is used. This finds the average distance ri$^{(K)}$for each point pi (i=1…n), in the dataset, considering K-nearest neighbours (KNN). The sigma rule is used on the dataset which means that a point is treated as an outlier if the result is not within N standard-deviations from the mean. Secondly, for the case where TLS point clouds needed alignment with other point clouds (i.e. of SfM photogrammetry), the iterative closest point (ICP) alignment tool (Rusinkiewicz and Levoy, 2001) is adopted in *CloudCompare*. The ICP tool co-registers point clouds in a pair-wise manner. It effectively searches for pairs of nearest points in two-point clouds and calculates the transformation matrix based upon them. Thirdly, cleaning and cropping of the point cloud are carried out to remove noise and irrelevant points (e.g. non-structural elements such as vegetation, etc.) in *CloudCompare*. This is an important task since only the points relating to the structure being assessed should be considered in structural analysis. This is achieved with relative care as to solely retain stages of the masonry structure inside the final point cloud that are for structural analysis. Finally, to assess the TLS point clouds' qualities, their surface density is calculated within *CloudCompare*. For instance, the surface point density can be calculated in pts/cm^2 by using the surface density calculator of *CloudCompare* with circular radii of R equal to 0.005641 m, corresponding to an area of 1 cm^2.

Orthoimage development

During step four, orthoimages of the intensity data are extracted using the raster image development function of *CloudCompare*. A requirement is that the point cloud be aligned with the plane of orthoimage development. Then, the orthoimage is developed by the projection of the selected face of the point cloud, with a user-defined resolution.

Mesh development

During step five, which is optional, a watertight mesh can be constructed using the Poisson surface reconstruction algorithm of the point cloud. The mesh reconstructed with a plugin within *CloudCompare* based on the well-known Poisson reconstruction algorithm (Kazhdan and Hoppe, 2013). The settings of mesh construction were an octree (the term octrees refers to the partitioning of the 3D space by recursively subdividing it into eight octants-octrees which are a 3D analogy of quadtrees), depth of 12, samples per node equal to 1.5, full depth equal to 5, the point weight equal to 4.0 and the boundary set as free.

3.2 Stage 2: Geometric model development workflow

The second stage of the three-stage methodological framework entails geometric model development from the geometric data of structural surveying. The term geometric model refers to the geometry used in structural analysis, as previously mentioned in Chapter 2. Three workflows are presented in the following Sections for geometric model development, according to various type of geometrical data.

3.2.1 Geometric model development with the point-based workflow

The workflow of this Section concerns geometric model development from discrete points, such as those from a total station or direct measurement with a tape. This type of workflows is most commonly employed in the block-based numerical modelling studies, such as the aforementioned in Section 2.2.2. The processes of the workflow are shown in Figure 3-5.

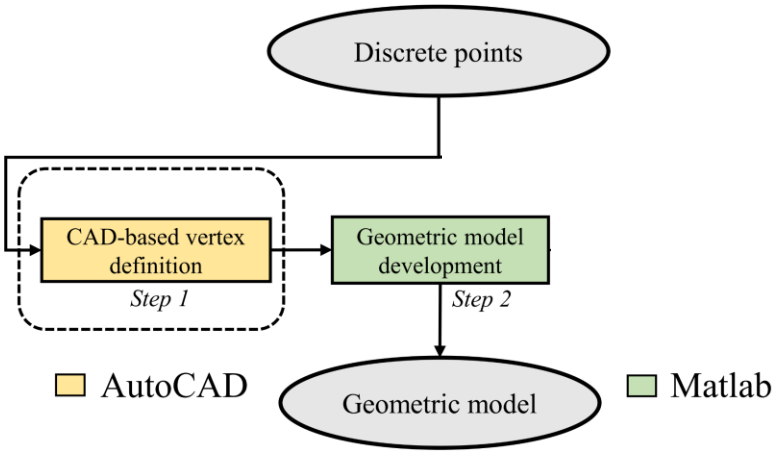


Figure 3-5: The point-based workflow of geometric model development.

Block vertex definition

During step one, the block vertices are defined through a manual CAD-based vertex definition. This process consists of manually defining the blocks of the masonry structure by adopting the *polyline* command of *AutoCAD* (Autodesk, 2019) with the given discrete points of a CAD reproduction. The reasoning for the employment of CAD software such as AutoCAD (Autodesk, 2019) was given to both its popularity and facility of implementation. Due to the nature of the numerical employed modelling method, the regions have to be convex, as will be detailed in the forthcoming Section 3.3.1 of structural analysis. The block vertices are then automatically extracted from the *polylines* with an existent *Matlab* script (Wischounig, 2020). Then the block vertices are stored in the herein termed *block vertex arrays* in a subsequent *Matlab* script of Section A.1.1.

Geometric model development

During step two, the geometric model to be used for the structural analysis model is developed in a format suitable for the employed numerical method for structural analysis (e.g. FEM, LA, DEM). In this investigation, a geometric model compatible with the employed DEM is developed. Namely, blocks are developed adopting the *polyhedron* command of the software *3DEC* (Itasca, 2019a). The polyhedron development is based upon assigning the polyhedron nodes in clockwise order, for the two faces with the *block vertex arrays* of Section 3.2.1. This is an automated process within a developed *Matlab* script of Section A.1.2.

3.2.2 Geometric model development with the image-based workflow

In the following paragraphs, the image-based workflow is presented in which geometric model development entails the employment of orthoimages. The processes of the workflow are herein detailed in a step-by-step fashion (shown in Figure 3-6).

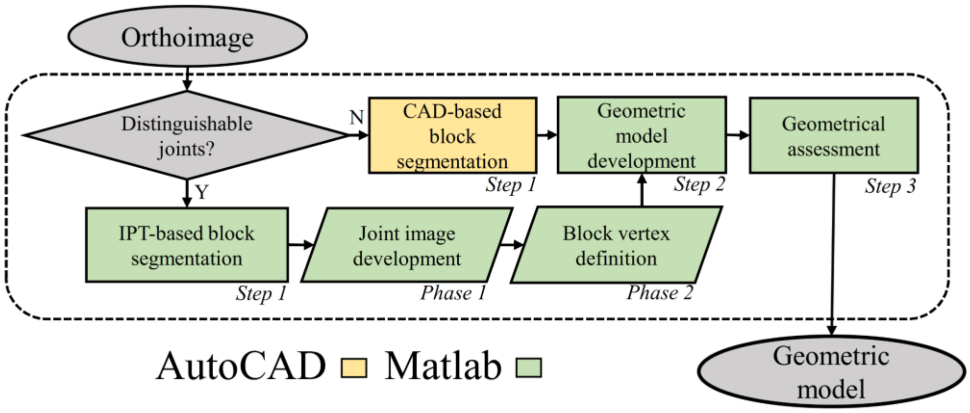


Figure 3-6: The image-based workflow.

Block segmentation

During step one, the block segmentation is carried out. The objective of this process is to define each block’s vertices within the given orthoimage. Here the blocks are classified into two types: distinguishable and non-distinguishable. Namely, non-distinguishable (Figure 3-7) are the blocks which do not have a clear separation of blocks and mortar or dry joints, as for instance, in Figure 3-7. Though block segmentation may be carried out with naked-eye on this type of image, they not able to be segmented by colour-based image segmentation.

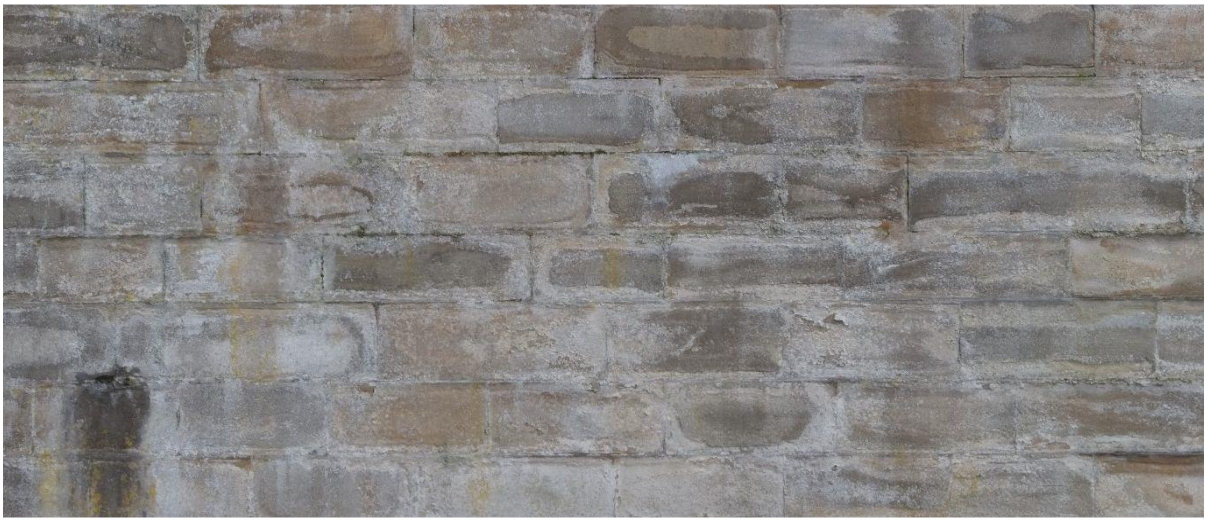


Figure 3-7: Non-distinguishable blocks of a masonry structure.

On the contrary, in the case where the blocks and mortar in the image are of a distinct colour (i.e. a block-joint colour contrast is existent), then the blocks are termed distinguishable. As a consequence, distinguishable blocks can be segmented with colour-based edge detection due to consistent block-joint colour contrast (as shown in Figure 3-8). It is however to be noted that the IPT-based approach is not fully-automated yet. This is given to difficulties such as the necessity of: a) cleaning noise of the orthoimagery prior to geometric model development; b) segmenting the structure for analysis for the orthoimagery; and c) simultaneously segmenting various types of masonry (i.e. distinguishable and non-distinguishable) which may be found on full-scale structure.

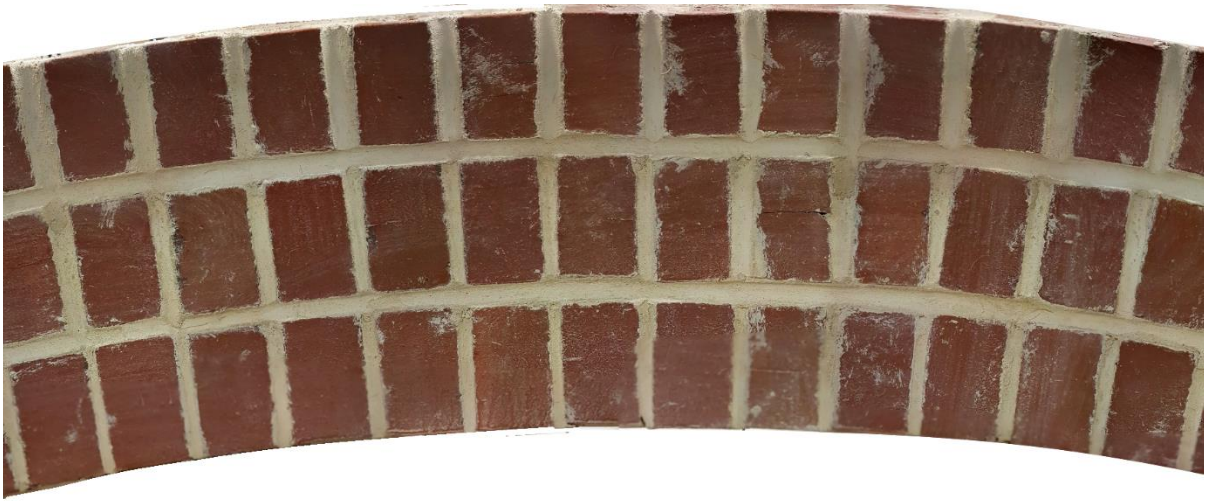


Figure 3-8: Distinguishable blocks of a masonry structure.

Manual CAD-based block segmentation: When the blocks are non-distinguishable then manual block segmentation must be carried out. This is by inserting the orthoimage into *AutoCAD* and defining the block vertices with superimposed points on the orthoimage. Following this, the points of each block are traced with a *polyline*, as with the point-based workflow (Section 3.2.1).

IPT-based block segmentation: When the blocks are distinguishable, block segmentation can be automated with the use of IPTs, employing functions of the well-known image processing toolbox of *Matlab* (Mathworks, 2019). IPT-based block segmentation consists of two phases, as detailed in the forthcoming paragraphs.

The first phase of IPT-based segmentation entails the joint image development, of which the *Matlab* script is found in Section A.2.1. Effectively, this is the development of an image made of the centrelines of the joints and borders of the block, having an equal width in pixels. The processes of the joint image development are shown in Figure 3-9a-g, which consist of: a) pre-processing of the greyscale orthomosaic (*roifill* function) to remove noise; b) edge detection of the pre-processed image (*edge* function) to highlight block edges; c) the mask creation (*imdil* function), to define the borders of the area occupied by the masonry structure; d) joint line detection with a Hough transformation (*hough* function), to make joints continuous; e) joint lines fusing with the mask (*imfuse* function); and f) joint line homogenisation and joint image development with watershed segmentation (*watershed* function).

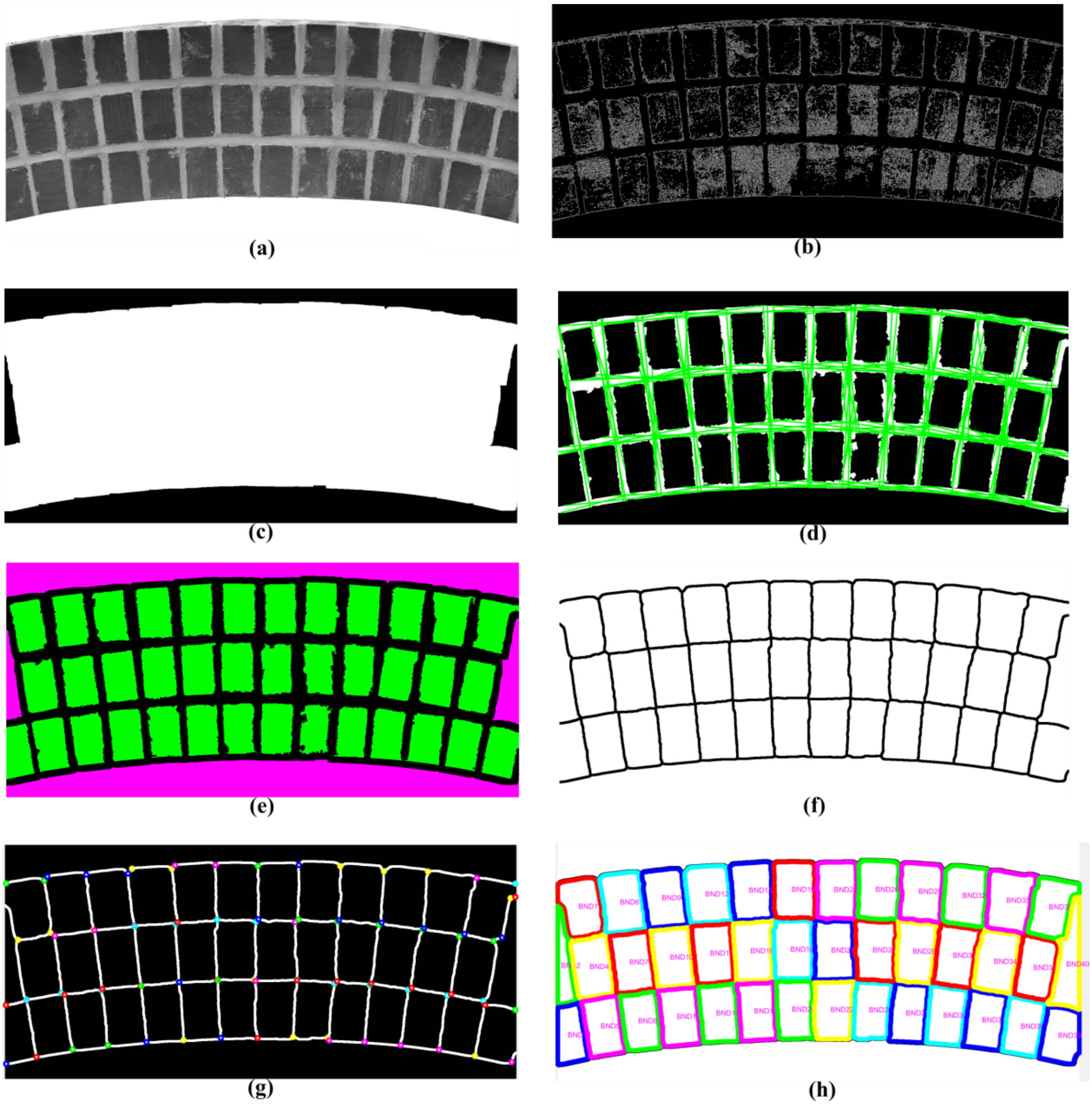


Figure 3-9: Block segmentation with IPTs for mortared joints. Joint image development: (a) pre-processing of the greyscale image; (b) edge detection; (c) the mask creation; (d) joint line detection; (e) joint lines and block border; and (f) joint line homogenisation and joint image development with watershed segmentation (watershed function). Block vertex extraction: (g) desired block vertices; and (h) permissible block vertices.

The second phase of IPT-based block segmentation concerns the block vertex definition, of which the *Matlab* script is also found in Section A.2.1. This effectively consists of extracting the block vertices from the joint image and assigning them to a corresponding data structure, the so-called $block\ array$. Specifically, where n_block is the total number of blocks, the $block\ array$ is a data structure composed of n_blocks double arrays, with the x and y vertices occupying the first and second rows respectively. For a given block l, the length of the corresponding array is equal to the number of vertices of the block, $n\_vertices$ (e.g.

$n\_vertices \ = 4$, for a tetrahedron). The process of extracting the block vertices from the joint image commences by finding the crossing of the joint image with the image skeleton (*imskel* function) and branch-points (*bwmorph* function) functions. These are herein termed the desirable block vertices, located on the centreline of the discontinuity. All the desirable block vertices are then detected and stored in an array, termed the $node\ array$, having a total number of vertices equal to n_nodes. To develop each block of the $block\ array$, firstly the boundaries of the joint image (*boundary* function) are calculated, termed the $boundary\ array$. This is a data structure of n_blocks, herein termed permissible block vertices. The permissible block vertices correspond to the borders of the areas of the blocks of the joint image. Although the vertices of $boundary\ array$ are conveniently organised in a clockwise or anticlockwise order per block, they cannot be employed for block development due to being: a) superfluous (i.e. 180); and b) not located on the centreline (i.e. if employed for geometric model development, there would be a gap between blocks).

To define the vertices of the block array, two processes are carried out. For a given block l, the first process is a loop, as shown in lines 1-9 of Algorithm 1, which entails replacing values of the $block\,array$ with values of the $node\,array$, within a user-defined threshold of Euclidian distance, equal to *threshold*. The second process, as shown in lines 10-12 of Algorithm 1, entails ascertaining controlling that $block\,array$ values are unique and only belonging to the $node\ array$.

Algorithm 1: Assigning node array values to the boundary array and assigning unique node array values to block array

```
$block \leftarrow l$
for $i = 1{:}\,n\_vertices$
    for $j = 1{:}\,n\_nodes$
        if $\mathbf{d}(boundary\ array(i,:,block), node\ array(j,:)) <= threshold$ then
            $boundary\ array(i,1,block) \leftarrow node\ array(j,1)$
            $boundary\ array(i,2,block) \leftarrow node\ array(j,2)$
        end if
    end for
end for
if intersect $(boundary\ array(:,:,block), node\ array)$ # 0
    $block\ array \leftarrow$ intersect$(boundary\ array(:,:,block))$
end if
```

Geometric model development

During step two, the geometric models are developed from the data of the $block\ array$. Before this, the blocks of the manual CAD-based segmentation and the IPT-based segmentation must be scaled to their actual size $block\ array_{scaled}$, from equation (3.5) below.

$$block\ array_{scaled}\ = (block\ array) \times Res_{orth.} \tag{3.5}$$

After the block scaling, geometric model development follows, identically to the process of Section 3.2.1 using the *block array* of the workflow.

Geometrical assessment

During step three, the geometrical assessment is carried out of the developed geometric models. However, before this, alignment of the developed (i.e. assessed) geometric models is necessary with a that of a reference geometric model (with the use of user-defined reference point). It is additionally important to note that the reference geometric models are previously developed, either from the image-based or point-based workflows. Alignment is subsequently carried out by translating all the coordinates of the $block\ array_{scaled}$ in the x and z-axis direction equal to dx_{align} dz_{align}, from equations (3.6) and (3.7) below. Only translation is considered necessary to align the geometric models as it is anticipated employed orthomosaic has already been subject to scaling and orientation during the structural surveying workflow. The blocks of the $block\ array_{aligned}$, found in (3.8), are aligned.

$$dx_{align} = x_{orth.} - x_{geom.model.} \tag{3.6}$$

$$dz_{align} = z_{orth.} - z_{geom.model.} \tag{3.7}$$

$$\begin{bmatrix} block\ array_{aligned}(:,1) \\ block\ array_{aligned}(:,2) \end{bmatrix} = \begin{bmatrix} block\ array_{scaled}(:,1) - dx_{align} \\ block\ array_{scaled}(:,2) - dz_{align} \end{bmatrix} \tag{3.8}$$

To calculate the geometric uncertainty, three metrics are considered absolute uncertainty (AU), the relative uncertainty (RU), and normalised uncertainty (NU) with the following equations (3.9), (3.10) and (3.11) as shown below.

$$AU = \ x^{ref} - x \tag{3.9}$$

$$RU = \frac{(x^{ref} - x)}{x^{ref}} \times 100 \quad (3.10)$$

$$NU = (x^{ref} - x)/max\,(x^{ref}) \quad (3.11)$$

Where x^{ref} and x are geometrical measures of the reference and assessed geometric models, respectively whilst max (x^{ref}) is the maximum value of the assessed group of geometrical properties.

Two types of geometric model properties are considered here according to Figure 3-10, the variable and invariable geometrical properties. The invariable geometry properties are considered those that describe the structure as a continuum (e.g. an arch's span, thickness and rise. The variable geometrical properties are those that describe of the blocks and joints, such as the: a) block centroid location; b) block volume; c) joint length; and d) joint inclination (herein termed joint dip) angle. It is noteworthy that the joint length is defined as the distance between the two extremities of the joint while the joint dip is defined as the angle of the joint with the x-axis, between zero and 90 degrees.

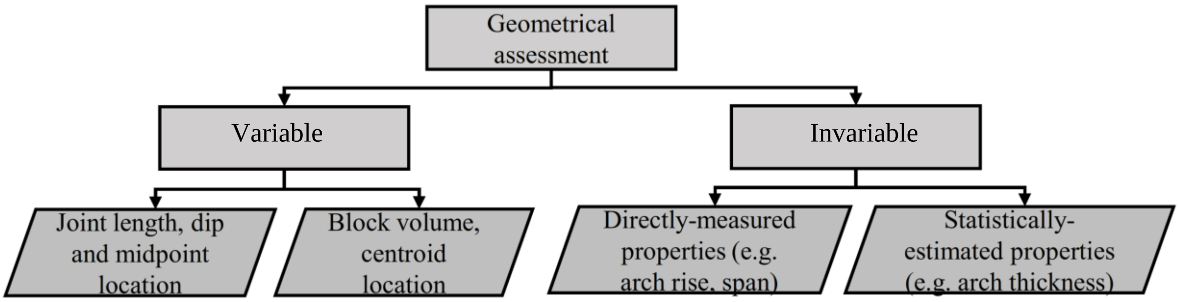

Geometrical assessment
Variable
Invariable
Joint length, dip and midpoint location
Block volume, centroid location
Directly-measured properties (e.g. arch rise, span)
Statistically-estimated properties (e.g. arch thickness)

Figure 3-10: Properties of geometrical assessment of image-based geometric models.

3.2.3 Geometric model development with the cloud-based workflow

In the following paragraphs, the cloud-based workflow is presented in which geometric model development entails the employment of point cloud. Conversely to the previously reported workflows which regard numerical modelling of regular masonry, the cloud-based workflow regards rubble masonry. It is important to reaffirm that the regular masonry refers to masonry that has a clearly defined anisotropy; in this case, the blocks can be segmented with image-processing, as demonstrated in Section 3.2.2. Whilst the employment of point cloud processing is possible for segmenting regular masonry, it is deemed signficantly more complex than image processing and thus only employed for rubble, as a manner of ficitisoully defining the random nature of the masonry Whilst the employment of point cloud processing is possible for segmenting regular masonry, it is deemed significantly more complex than image processing and thus only employed for rubble, as a manner of fictitiously defining the random nature of the masonry. The processes of the cloud-based workflow are herein detailed in a step-by-step fashion (shown in Figure 3-11).

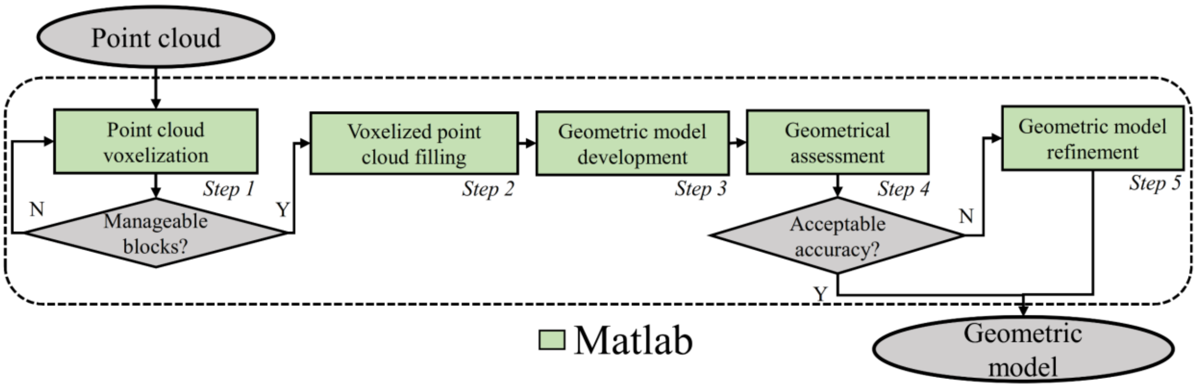


Figure 3-11: The cloud-based workflow.

Point cloud voxelization

During step one, voxelization of the dense point cloud is carried out. The developed voxelization algorithm herein presented is a point-based type, theoretically similar to a previous masonry modelling approach (Hinks *et al.*, 2012). It involves the down-sampling of the point cloud into a sum of equidistant points that have a common global axis orientation. The first process of point cloud voxelization consists of the selection of the voxel size, $Grid$. This defines the actual voxel dimension so its appropriate choice is necessary for the correct accuracy and manageability of the structural analysis. Specifically, as will be further on demonstrated, this value must be controlled as it defines the number of blocks which above a certain threshold can make the numerical simulation computationally unmanageable. Indicatively, in Chapter 6, maximum block thresholds are stated for given computational resources. After voxel size selection, the next process of voxelization consists of finding the bounding box of the dense point cloud. The bounding box is composed of minimum and maximum spatial coordinates (x_{max}, x_{min}), (y_{max}, y_{min}), $\left(z_{max,}\ z_{min}\right)$ of the dense point cloud in meters. Then the bounding box is subdivided into a grid with the user-defined voxel size equal to Grid for the x, y, and z-axis, respectively. The number of voxels for each axis (N_x, N_y, N_z) is defined by the following equations (3.12), (3.13) and (3.14). Ceil is the ceiling function used and Grid, the voxel size in meters.

$$N_x = ceil(\frac{(x_{max}-x_{min})}{Grid}) \tag{3.12}$$

$$N_y = ceil(\frac{(y_{max}-y_{min})}{Grid}) \tag{3.13}$$

$$N_z = ceil(\frac{(z_{max}-z_{min})}{Grid}) \tag{3.14}$$

Due to the number of voxels being an integer due to the rounding of the ceiling function (3.12), (3.13) and (3.14) the voxel size and actual voxel dimensions cannot coincide. Thus the actual voxel dimensions for the x, y and z axes are defined $(\varDelta_x, \varDelta_y, \varDelta_z)$ respectively by the equations (3.15), (3.16) and (3.17):

$$\varDelta_x = \frac{(x_{max}-x_{min})}{N_x} \tag{3.15}$$

$$\varDelta_y = \frac{(y_{max}-y_{min})}{N_y} \tag{3.16}$$

$$\varDelta_z = \frac{(z_{max}-z_{min})}{N_z} \tag{3.17}$$

Now, similar to the approach of (Hinks *et al.*, 2012), the final processes of voxelization are described. First of all, the dense point cloud P, must be defined according to Volodine (2007) as an unordered collection of *n* points $\{P_i\}_{i=1}^{n}$ in 3D Euclidean space, resulting from the scanning of an object and representing the surface of that object. Within the voxelization, the columns representing the x, y and z axes of this dense point cloud P_x P_y and P_z are divided by their corresponding actual voxel dimension and rounded using the round function as in (3.18), (3.19) and (3.20) Dim_x, Dim_y, and Dim_z are then horizontally concatenated and compose the rounded point cloud Dim, as in (3.21). This is effectively a dimensionless array that indicates which voxel grid each point belongs with an integer index for x, y, and z axes. As there are multiple points for each voxel, recurring points are removed by finding the unique rows of the rounded point cloud Dim using the unique function. This results in the so-called dimensionless voxelized point cloud DVC, with only one occasion of each voxel as in (3.22). Then the so-called empty voxelized point cloud EVC, is found which is the dimensionless voxelized point cloud multiplied by the respective actual voxel dimensions, $(\varDelta_x, \varDelta_y, \varDelta_z)$, as in (3.23).

$$Dim_x = round(\frac{(P_x)}{\varDelta_x}) \tag{3.18}$$

$$Dim_y = round(\frac{(P_y)}{\Delta_y}) \tag{3.19}$$

$$Dim_z = round(\frac{(P_z)}{\Delta_z}) \tag{3.20}$$

The rounded point cloud is composed of the results of (3.18), (3.19) and (3.20) horizontally concatenated:

$$Dim = [Dim_x, Dim_y, Dim_z] \tag{3.21}$$

The dimensionless voxelized point cloud is found from the following equation:

$$DVC = unique\ [Dim_x, Dim_y, Dim_z] \tag{3.22}$$

The empty voxelized point cloud EVC, is equal to the dimensionless voxelized point cloud, DVC, multiplied by the corresponding actual voxel dimension as in the following equation:

$$EVC = [Dim_x \times \Delta_x, Dim_y \times \Delta_y, Dim_z \times \Delta_z] \tag{3.23}$$

The empty voxelized point cloud, EVC is the final product of voxelization and is essentially a point cloud which describes the dense point cloud as a sum of the active voxels. The term active voxel specified that it is occupied by at least on point of the dense point cloud.

If the voxel size ($Grid$) is smaller than the available point cloud surface density, then there will be voids where the gridline is not occupied by active points. It is useful for the mean surface densities of the dense point cloud to be measured before voxelization to find the smallest permissible voxel size with the given dense point cloud. This can be done by measuring the population within *CloudCompare* with the density measure function, as discussed in Section 3.1.3.

Voxelized point cloud filling

During step two, the empty voxelized point cloud is filled with points. This is the key step of the workflow since the internal geometry of the three-dimensional structure is not hollow and thus internal points of the empty voxelized point cloud need to be defined. Similar to Castellazzi *et al.* (2015), the empty voxelized point cloud is treated as an assembly of orthoimages with common pixel size and dimension and characterised by a specific height z.

To fill the entirety of the masonry structure’s domain, a so-called raster image approach is adopted. Namely, for each level (i.e. heights of voxels) of the empty voxelized point cloud, a corresponding raster image corresponding to the x-y axes plane is extracted and morphologically processed. Effectively, for each raster image (i.e. for every level of voxels), all the inactive pixels found to be contained inside the perimeter of the masonry structure are converted into active pixels. Then, the filled raster images are stacked together, resulting in the filled voxelized point cloud. It is noteworthy that the raster image approach can be applied to any type of point cloud, whilst the described procedures are fully automated and incorporated into the *Matlab* script of Section A.3.1. The perimeter of the empty voxel cloud needed to be continuous, so that its contained area may later be filled. If not continuous, it could be easily modified to be. By using the raster image approach, the tedious 3D surface becomes a 2D problem.

Geometric model development

During step three, the cloud-based geometric model is developed. Each block of the geometric model is defined as an 8-noded *polyhedron*, based upon assigning the polyhedron nodes in clockwise order, for two parallel faces of each voxel of the cloud. This is an automatic procedure within the same *Matlab* script of Section A.3.1.

Geometrical assessment

During step four, the accuracy of the geometric models is assessed. The level of geometrical accuracy (and the consequent geometric features included in the model) is directly correlated to the voxel size. The employed voxel size governs the amount of detail with which geometrical features of the structure can be described. For this investigation, a satisfactory geometrical representation is considered when obtaining: a) an accurate structural volume, to accurately represent the mass of the structure; and b) accurate cross-sections, to provide a representative structural behaviour. Geometrical assessment is carried out by comparing the geometrical properties of the geometric models with that of a reference mesh and comparing: a) the volumetric difference of the geometric model with the mesh; and b) the cloud-to-mesh distance of the geometric model with the reference mesh in *CloudCompare* with the dense point cloud before voxelization. Specifically, the cloud-to-mesh (C2M) is a way of geometrically assessing a point cloud by measuring the distance between it and a reference mesh. In this way, a mean scalar value determines the overall accuracy of the geometric model’s cross-sections. Finally, it is notable that during the voxelization, there is no translation or rotation of the point cloud is involved and thus for comparison between the voxelized point clouds and reference mesh, alignment is not required.

Geometrical model refinement

During step five, the geometric models are optionally refined. A disadvantage of this workflow is made apparent in this step. Namely, to obtain a geometric model of high geometrical accuracy (e.g. with less than 5% geometrical error), the adopted voxel size may cause the numerical simulation to be either unmanageable or non-executable. Furthermore, geometric models of larger voxel size are often of a high geometrical error and must be refined to be considered geometrically accurate. A procedure of geometric model refinement is herein presented to increase the volumetric accuracy of the voxel models. The idea of this process is to first calculate the volumetric error and then subtract it from the geometric model, by reducing the actual voxel dimensions. The voxel dimensions are corrected by multiplying the actual voxel dimensions $(\mathit{\Delta}_x, \mathit{\Delta}_y, \mathit{\Delta}_z)$ by the so-called volume adjustment coefficient (VAC), as obtained from equation (3.24). Finally, to control the geometric refinement, the refined voxel model's volume can be obtained from equation (3.25).

$$VAC = \sqrt[3]{\frac{(Watertight\ mesh\ volume)}{(Voxelized\ model\ volume)}} \tag{3.24}$$

$$Refined\ voxelized\ model\ volume = (Voxelized\ model\ volume) \times VAC^3 \tag{3.25}$$

3.3 Stage 3: Structural analysis

The third stage of the three-stage methodological framework concerns the structural analysis of masonry structures, departing from a given geometric model. This Section commences with the detailing of the theoretical formulation of the DEM within the software *3DEC*. Then a step-by-step workflow is provided for structural analysis on a given geometric model with *3DEC*.

3.3.1 The discrete element method in 3DEC

Various commercial and open-source software packages, as is evident in Table 3-5 were considered for this investigation which all belong to the DEM. Although a variety of other software can provide elegant solutions for problems of masonry, the commercial software package *3DEC* was favoured. The reasoning for its selection was: a) it employs a user-friendly, quasi "black-box" workflow, which facilitates implementation; b) its effectivity has been demonstrated in vast the majority of investigations on masonry with the DEM (as was found from Section 2.2.2); c) as opposed to *UDEC*, *3DEC* enables the three-dimensional structural analysis, which is particularly advantageous for masonry structures of a highly complex geometry; and d) as opposed to *YADE*, *PFC* it employs polyhedral and polygonal elements, which better-suited for blocks of masonry structures (rather than particles).

Table 3-5: Discrete element method software.

Software	Source	Type	Numerical method	Block dimensions and type
3DEC	(Itasca, 2019a)	Commercial	DEM	3D Polyhedral
UDEC	(Itasca, 2019c)	Commercial	DEM	2D Polyhedral
PFC	(Itasca, 2019b)	Commercial	DEM	3D Spherical
YADE	(Šmilauer *et al.*, 2010)	Open-source	DEM	3D Spherical

In the next paragraphs, the numerical formulation of the DEM within *3DEC* is presented. To pursue this aim, the following items are detailed: a) the physical entities and the data structure; b) contact identification and detection; c) the calculation cycle; d) mechanical damping; e) numerical stability; and f) mass scaling. The interested reader is also referred to the software manual of *3DEC* (Itasca, 2019a) as well as published studies and reviews (Jing, 2003; Zhu *et al.*, 2007; Zhu *et al.*, 2008; Bobet *et al.*, 2009; Lisjak and Grasselli, 2014).

The physical entities and the data structure

The core physical entities of *3DEC* are the blocks and contacts which are linked to the data structure by pointers. It is important to note that the data structure is, in turn, a single main array that holds real, integers and mixed numbers of which the basic function is to rapidly retrieve data for the mechanical calculations of the simulations carried out in *3DEC.* Since the data structure solely employs computational random-access memory (RAM), it becomes apparent that to obtain manageable simulation times within *3DEC*, the RAM size is crucial.

Blocks: Blocks are the fundamental geometric entities for the distinct element calculation. In *3DEC*, blocks are either polygonal or polyhedral. A limitation of *3DEC* is made apparent here in that it can only simulate convex blocks. While concave blocks can be simulated, they must, however, be developed as an assembly of kinematically constrained convex blocks throughout the simulation. Consequently, this presents a major challenge in the geometric model development process. Within *3DEC*, the polyhedral and polygonal blocks are represented by the terms *polyhedron* and *polygon* respectively. Concerning the deformability of the blocks, both deformable and rigid blocks can be employed in *3DEC*, but not in the same system. While all blocks have six degrees-of-freedom (three translational and three rotational) at the centroid, deformable blocks have a further three translational degrees of freedom at each *vertex* (or node), due to the subdivision in tetrahedra. Finally, it is to be noted that the block faces (i.e. planar polygons) are termed *faces* and the nodes making up the polygons, *vertices*. Within the data structure, a circular list corresponds to the *vertices* each *face*, organised in clockwise or anti-clockwise order.

In the case of deformable blocks, some additional physical entities exist. First of all, the *faces* are further discretised into tetrahedral finite-difference *zones* while mechanical changes (e.g., stress/strain) are calculated within each *zone*, in accordance with the finite element discretisation. Moreover, further than *vertices*, deformable blacks entail *gridpoints* which are associated with the corners of the tetrahedral finite-difference *zones* (or *subzones*). Four *gridpoints* are associated with each *zone* with a set of x, y, z coordinates, specifying the exact location of the finite-difference zones.

Contacts: When two blocks come into contact, an element is assigned which is defined as a *contact*. This corresponds to the physical contact between the two blocks, which has a corresponding data structure composed of relative information of physical contact including forces, stresses and displacement. Furthermore, there are subdivisions of the initial contacts into smaller elements, termed *sub-contacts*. The geometrical location to which the contact belongs to will be detailed further (in the so-called common-plane (CPL) algorithm). Presently it is to be noted that, if a block face is in contact with the common-plane, then it is automatically discretised into sub-contacts. For the case of rigid blocks which are of more interest to this investigation, *faces* are triangulated to create the *sub-contacts*.

Contact identification and detection

During the calculation cycle[5], one of the main tasks is that of identifying the existence and classifying the contacts type between blocks which is achieved by the so-called cell mapping and searching which is detailed in the following paragraphs accordingly.

Contact neighbour identification by cell mapping and searching: The system of blocks is subdivided into an imaginary set of 3D cells in which the addresses of blocks occupying it are stored. The process of identification of neighbouring blocks commences by controlling each cell for the existence of multiple addresses. Two problems are made apparent here relating to the size of the cell: a) if the cell size is smaller than the block, the identification process will not find any neighbours; and b) if the cell size is too big, too many neighbours will be found. Both these problems are overcome through selecting the average bounding box (termed *bounding envelope*) of the blocks. Tests and remapping for contact neighbours are carried out recursively during each calculation cycle. To identify contacts as neighbours, a variable for contact identification $u_{acc.}$, is also employed as such:

$$u_{acc.} = u_{acc.} + max\,\{abs(du)\} \tag{3.26}$$

Where C_{tol} is a user-defined contact tolerance valuable, the variable for contact identification is also limited as such:

$$u_{acc.} \geq C_{tol} \tag{3.27}$$

Contact detection: If a given contact is identified, its type must be classified. Since a contact’s type significantly influences its mechanical response, this is crucial for the application of the correct calculation procedure. Contacts are classified according to the number of block vertices touching the so-called common-plane (in Figure 3-12, an illustration of an edge-to-edge contact is shown) which is detailed in the following paragraph. Also, characteristic types of contacts are additionally listed in Table 3-6.

[5] The calculation cycle is detailed in the following paragraphs, on page 78.

Table 3-6: Contact types in 3DEC.

Number of vertices touching the common plane		Contact Type
Block A	Block B	
0	0	null
1	1	vertex-vertex
1	2	vertex-edge
1	>2	vertex-face
2	1	edge-vertex
2	2	edge-edge
2	>2	edge-face
>2	1	face-vertex
>2	2	face-edge
>2	>2	face-face

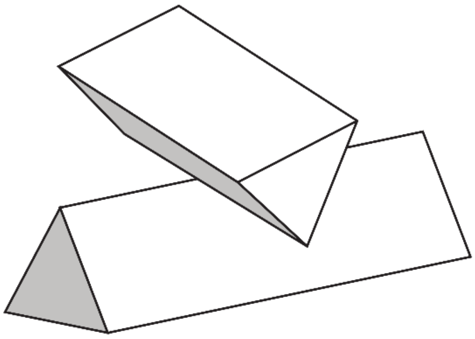


Figure 3-12: An example of the rare case of edge-to-edge contact (Itasca, 2019a).

The task of contact detection, if performed using a direct test, is a particularly onerous computational procedure, especially in 3D. In *3DEC*, direct contact is overcome with the notion of the common plane (CPL) which is a numerical device, conceived analogous to a metal plate between two blocks (Figure 3-13). The logic of the numerical device is as such: the pressing the blocks together (presupposing that the blocks are both convex), requires that they come into contact at a single point and angle, as if an imaginary metal plate were between them. This imaginary metal plate represents the notion of the CPL.

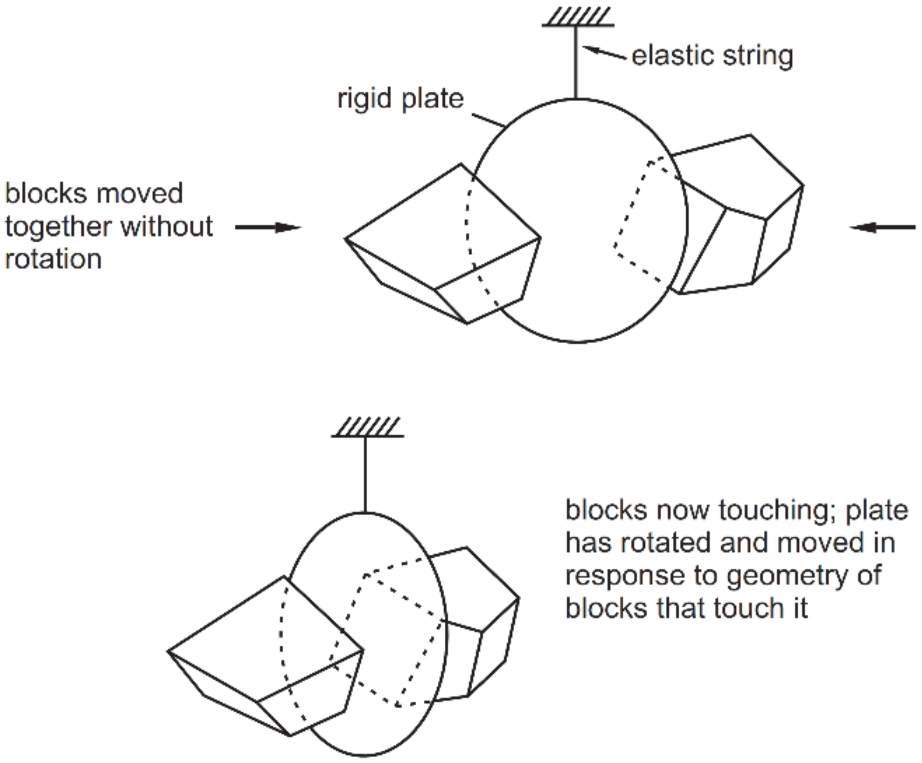


Figure 3-13: Illustration of the notion of the common-plane (symbolised CPL) (Itasca, 2019a).

The problem of rapid and robust contact type classification is solved by finding the vertex-face contacts between each block and its relative common-plane, whilst the algorithm which carries out this process is the termed *common-plane algorithm*. For non-overlapping blocks, the algorithm aims to maximize the gap between the CPL and the closest vertex whilst for overlapping blocks, it aims to minimize the overlap between the CPL and the vertex with the greatest overlap. The position of the CPL is initially estimated at the midpoint between the two blocks whilst the actual position is calculated through iteration. Although there are reported cases where iterations are more computationally expensive than direct contact detection, the common-plane algorithm is almost always advantageous (Itasca, 2019a). The exact equations of the common-plane algorithm are found in Itasca (2019a) whilst an indicative illustration of the CPL is founding in Figure 3-14.

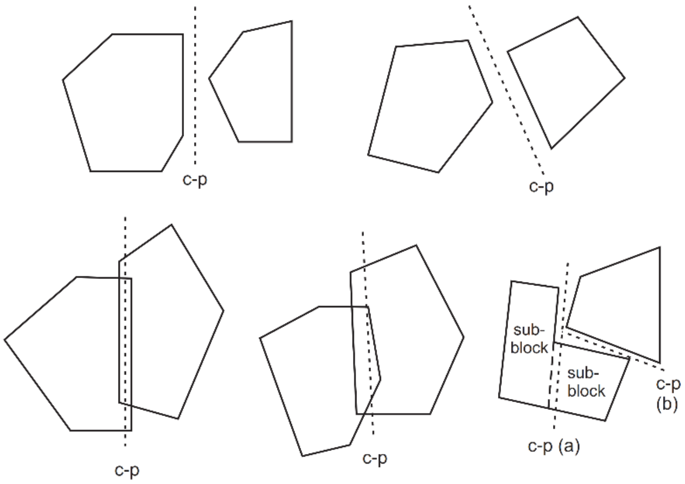


Figure 3-14: Possible combinations of the common-plane (symbolised CPL) (Itasca, 2019a).

The calculation cycle

Simulation is carried out in *3DEC*, through the so-called calculation cycle per each timestep, as illustrated in Figure 3-15. For a given simulation of a physical problem, several calculation cycles are usually needed to reach a solution, which is termed convergence. The calculation cycle is composed of four cyclic processes: a) the application of the equation motions per block centroid; b) block centroid acceleration, velocity and displacement calculation through the integration of the law of motion; c) contact forces updating with the sub-contact force-displacement law; and d) block centroid force recalculation. It's noted that for simplification, only the calculation cycle for the case of rigid blocks is presented, though for the case deformable blocks, the calculation is similar.

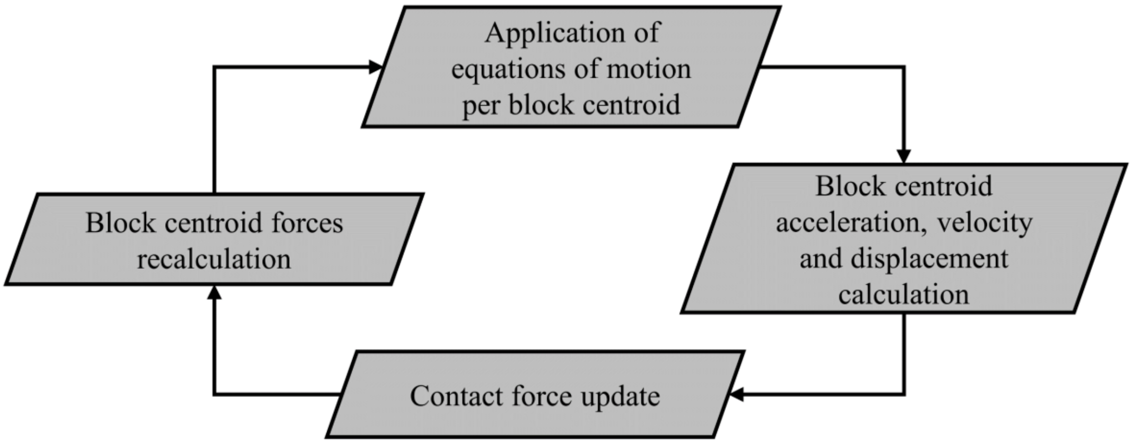


Figure 3-15: The calculation cycle of 3DEC.

Application of equations of motion per block: These are two equations, the equation of translation and rotation. The following is the equation of translation for a rigid block:

$$\ddot{x}_i + a \times \dot{x}_i = \frac{F_i}{m} + g_i \tag{3.28}$$

Where $\ddot{x}_i$ and $\dot{x}_i$ are the block acceleration and velocity respectively, a is the mass-proportionate damping constant, F_i is the sum of contact and joint forces acting on the block, m is the block mass, g_i is the gravity acceleration vector, and the index i corresponds to values of 1 to 3 of the global axes.

The following equation describes the equation of rotation (motion is referred to the global axes and because velocities are small, the non-linear term is dropped):

$$\dot{\omega}_\iota + a \times \omega_i = \frac{M_i}{I} \tag{3.29}$$

Where $\dot{\omega}_\iota$ is the angular acceleration, ω_i the velocity of the blocks, a is the mass-proportionate damping constant, M_i is the sum of torque and I is the moment of inertia of the block

Block acceleration, velocity, and displacement calculation: A central finite-difference time integration method is applied to integrate the aforementioned equations of motions. At a time t (in terms of the values at mid-intervals), the translational and rotational velocities are:

$$\dot{x}_i(t) = \frac{1}{2} \times [\dot{x}_i \left[t - \frac{\Delta t}{2}\right] + \dot{x}_i \left[t + \frac{\Delta t}{2}\right]] \tag{3.30}$$

$$\omega_i(t) = \frac{1}{2} \times [\omega_i \left[t - \frac{\Delta t}{2}\right] + \omega_i \left[t + \frac{\Delta t}{2}\right]] \tag{3.31}$$

Furthermore, block accelerations are:

$$\ddot{x}_i(t) = \frac{1}{\Delta t} \times [\dot{x}_i \left[t - \frac{\Delta t}{2}\right] + \dot{x}_i \left[t + \frac{\Delta t}{2}\right]] \tag{3.32}$$

$$\dot{\omega}_i(t) = \frac{1}{\Delta t} \times [\dot{x}_i \left[t - \frac{\Delta t}{2}\right] + \dot{x}_i \left[t + \frac{\Delta t}{2}\right]] \tag{3.33}$$

Finally, the block displacements are:

$$x_i\ (t + \Delta t)\ = x_i(t) + \Delta x_i \tag{3.34}$$

$$x_i^v\ (t + \Delta t) = x_i^v(t) + \Delta x_i + e_{ijk} \times \Delta \vartheta_j \times [x_k^v(t) - x_k\ (t)] \tag{3.35}$$

Where e_{ijk} is the permutation tensor, and vertices position are symbolised with v. It's also notable that the indices i, j and k correspond to values from 1 to 3 and refer to the global axes, while t is the variable of time, Δt is the time interval, Δx_i is the translational increment and $\Delta \vartheta_i$ is the rotational increment which are both given from the following:

$$\Delta x_i = \dot{x}_i \times [t + \frac{\Delta t}{2}] \times\ \Delta t \tag{3.36}$$

$$\Delta \vartheta_i = \dot{\omega}_i \times [t + \frac{\Delta t}{2}] \times\ \Delta t \tag{3.37}$$

<u>Contact force update:</u> Considering a two-block system of blocks A and B, if the two blocks A and B are in contact, then the contact velocity, V_i (defined as the velocity of block B relative to block A at the sub-contact location) is defined by:

$$V_i = \dot{x}_i^B + e_{ijk} \times\ \omega_j^B \times (C_k - B_k) - \dot{x}_i^A - e_{ijk} \times\ \omega_j^A \times (C_k - A_k) \tag{3.38}$$

Where C_k is the location of the CPL, e_{ijk} is the permutation tensor, the indices i , j and k correspond to values of 1 to 3 of the global axes and ω_j is the rotational velocity. From the contact velocity, where n_i is the contact normal vector, the absolute, normal, and shear displacement increments $\Delta U_i, \Delta U_i^n, \Delta U_i^s$ are calculated from the equations:

$$\Delta U_i = V_i \times \Delta t \tag{3.39}$$

$$\Delta U_i^n = \Delta U_i \times n_i \tag{3.40}$$

$$\Delta U_i^s = \Delta U_i \times \Delta U_j \times n_i \times n_j \tag{3.41}$$

Thereafter, from the displacement increments, the contact force increments $\Delta F^n, \Delta F_i^s$ are calculated from:

$$\Delta F^n = -K_n \times \Delta U^n \times A_c \tag{3.42}$$

$$\Delta F_i^s = -K_s \times \Delta U_i^s \times A_c \tag{3.43}$$

Finally, the normal and shear contact forces F^n, F^s are calculated as such:

$$F^n = F^n + \mathit{\Delta} F^n \tag{3.44}$$

$$F_i^s = F_i^s + \mathit{\Delta} F_i^s \tag{3.45}$$

Block force calculation: Once the sub-contact forces are calculated, they are added to the forces and moments which are already acting on the centroids of the blocks (e.g. gravity etc.). For a two-block system, the sub-contact force F_i to be added to the block would be essentially the total force acting on block A from B, as follows:

$$F_i = -(F^n \times n_i + F_i^s) \tag{3.46}$$

Thus for a given block A, where c_j is the position vector of the sub-contact, the sums of force F_i^A and moment M_i^A are therefore updated as such:

$$F_i^A = F_i^A - F_i \tag{3.47}$$

$$M_i^A = M_i^A - e_{ijk} \times (c_j - A_j) \times F_k \tag{3.48}$$

Mechanical Damping

Mechanical damping is employed to primarily simulate the actual physical damping of the system (and consequently lead to convergence) and secondarily to ensure numerical stability. *3DEC* provides damping types for both static and dynamic problems. For this investigation, velocity-proportionate damping is employed, similar to the so-called dynamic relaxation of Otter *et al.* (1966), however with some alterations. Namely, despite the effectivity of this velocity-proportionate damping, its direct application is found to cause three problems with *3DEC*: a) it introduces erroneous forces; b) modal analysis is necessary for the definition of the optimum proportionality constant; and c) the application of the damping is erroneously global, (e.g. to every node of the problem). To overcome these difficulties, *3DEC* provides the following two forms of conventional dynamic relaxation.

The first damping type is termed *adaptive global damping*, in which the viscosity constant of the viscous damping forces is continuously adjusted during the simulation. This is through controlling the power absorbed by damping, P, relative to the rate of change of kinetic energy, $\dot{E}_k$ according to the following ratio, R:

$$R = \frac{\Sigma P}{\Sigma \dot{E}_k} \tag{3.49}$$

The second damping type is termed *local damping*, in which an additional numerical term is added to the blocks' motion equations. This is the so-called damping force and is proportional to the magnitude of the unbalanced force. In this case, the orientation of the damping force is such that decreases block acceleration and increases block deceleration, leading to continuous energy dissipation. According to Cundall (Cundall, 1987) the three stated problems of conventional dynamic relaxation velocity-proportional damping are overcome with *local damping* due to: a) body force diminishing for steady-state conditions; b) the damping constant is independent of properties or boundary conditions and dimensionless in magnitude; and c) variable amount of damping is applied for each point.

Numerical Stability

Due to the nature of the explicit time integration method, *3DEC* is conditionally stable. This means that in the time-stepping solution, the magnitude of the timestep Δt influences the numerical stability of the problem. Thus, to ensure numerical stability, the timestep must satisfy a criterion governed by both the node-defined timestep, Δt_n system-defined timestep Δt_b, as found in the following equations:

$$\Delta t_n = 2 \times min\ (\frac{m_i}{k_i})^{1/2} \tag{3.50}$$

$$\Delta t_b = 2 \times (frac) \times (\frac{m_{min}}{k_{max}})^{1/2} \tag{3.51}$$

$$\Delta t = min\ \{\Delta t_n, \Delta t_b\} \tag{3.52}$$

Where m_i and k_i are the mass and stiffness associated with a blocks node, i and m_{min} is the mass of the smallest block and k_{max} is maximum stiffness of the system.

Mass scaling

An inherent problem of the DEM is made apparent here (which applies for any high-level numerical modelling approach), which is the computational burden. Within *3DEC*, for the optimisation of computational resources, a numerical device is employed termed *mass scaling*. Effectively, this is an approach to increase of the timestep's value to accelerate the simulation by decreasing the total number of timesteps until convergence. From equations (3.50) and

(3.51), an attractive solution to increasing the timestep's value is by reducing the mass of the system. Thus, with *mass scaling*, an artificial reduction of the density of all the systems blocks is carried out. It must be noted that this is only effective for non-uniform models, without notable improvement of a uniform model (i.e. in a problem of densities of the same magnitude).

3.3.2 Numerical model development and structural analysis workflow

This Section proposes a workflow of structural analysis of masonry structures using *3DEC*, commencing from a given geometric model. Specifically, the processes of the workflow of numerical model development and structural analysis with *3DEC* are detailed in a step-by-step fashion, according to Figure 3-16. The first four steps regard numerical model development while the final step regards (which is also the result of the methodological framework) structural analysis.

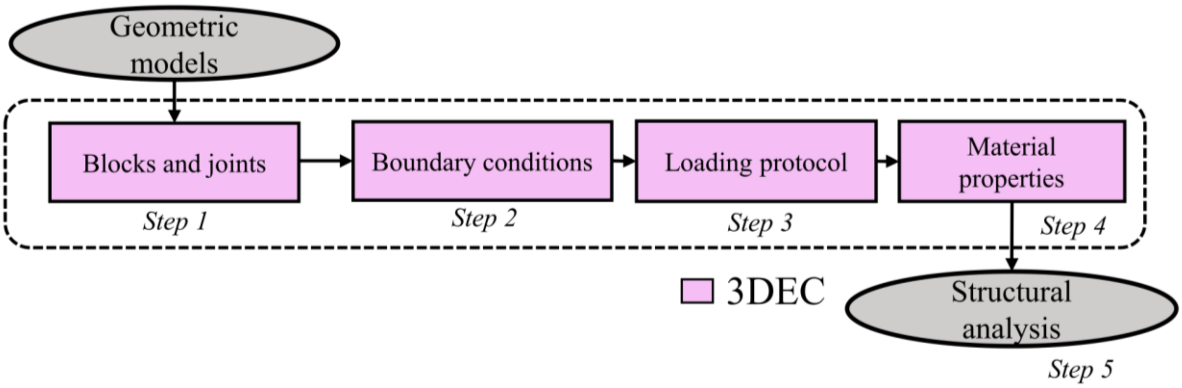


Figure 3-16: Numerical model development and structural analysis workflow.

Blocks and joints

During step one, the definition of the block and joint type is carried out. Since these two properties significantly affect both the predicted structural behaviour and computational resources needed, strategic selection should be carried out per each masonry structure. It is noteworthy that, while the constitutive models of both the blocks and joints defined here, the material property definition is carried out in another following step.

<u>Blocks type selection:</u> Within *3DEC*, the masonry units are represented as either rigid or deformable blocks. With rigid blocks, all deformability of the system is theoretically lumped at the joints. However, due to the fact of overlapping of contacts, a small amount of structural plasticity is observed, even for rigid blocks. For deformable blocks, the block themselves can deform accommodating further structural deformation. The selection of block type is based on:

a) the type of problem, whether dynamic or static; b) the computational resources available; and c) the number of blocks needed to be simulated (Sarhosis *et al.*, 2016d). In comparison with rigid blocks, deformable blocks present a challenge of a higher computational burden due to the need for additional physical entities within the calculation cycle (i.e. the tetrahedral finite-difference zone). In this investigation, rigid blocks were considered advantageous due to their lesser computational burden and compatibility with the nature of the addressed problems.

Blocks constitutive models: For the modelling of deformable blocks in *3DEC*, various block constitutive models are available. Since most of the investigations with *3DEC* on masonry have been carried out with the default block constitutive model (Çaktı *et al.*, 2016; Pulatsu *et al.*, 2016; Bui *et al.*, 2017; Forgacs *et al.*, 2017; Forgács *et al.*, 2018; Pulatsu *et al.*, 2018) the *3DEC*, it was also selected for this investigation. It can be noted that in the particular block constitutive model, strain increments are generated by stress increments according to the linear and reversible law of Hooke.

Joint constitutive models: Joints can be either dry or mortared which are both represented within the block-based modelling strategy as zero-thickness interfaces, corresponding to the centreline of the actual masonry joints. The definition of the joint constitutive models consists of defining the stress-displacement constitutive laws that govern the contacts between the blocks (Itasca, 2019a). Though more complex contact constitutive models are available such as that of Pulatsu *et al.* (2019b), the joint constitutive model adopted in this investigation is the *3DEC* default, Coulomb friction law (Figure 3-17). This was also adopted based on its widespread employment in the majority of masonry studies with *3DEC* (Çaktı *et al.*, 2016; Pulatsu *et al.*, 2016; Bui *et al.*, 2017; Forgacs *et al.*, 2017; Forgács *et al.*, 2018; Pulatsu *et al.*, 2018).

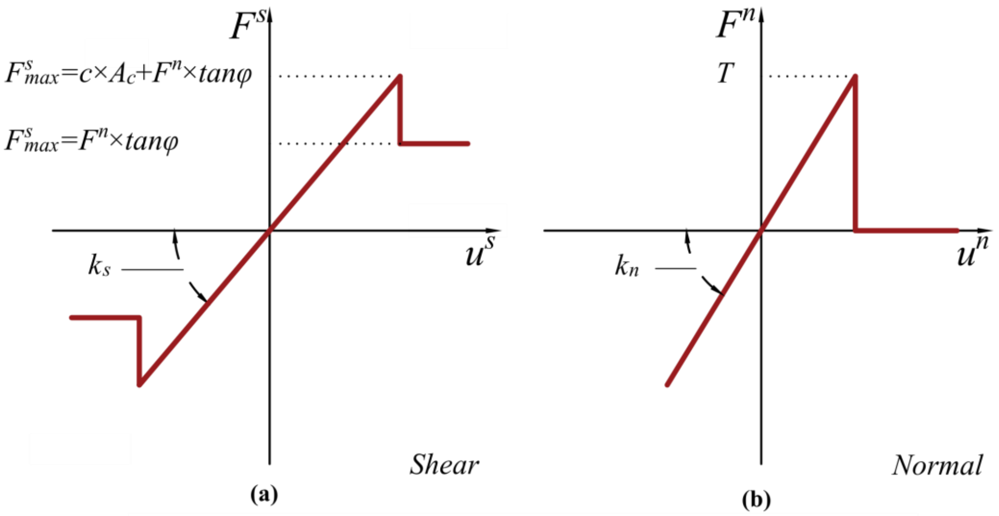


Figure 3-17: Joint constitutive model adopted: shear (a) and normal direction (b).

Within the specific joint constitutive model, both shear and tensile failure are considered, whilst joint dilatation can also be included. According to the adopted joint constitutive model, the joints' behaviour is governed by the joint normal and shear stiffnesses, K_n and K_s in the elastic range. Damage to the joints consists of either tensile or shear failure. The factory default joint constitutive model adopts both a residual cohesive and tensile strength equal to zero, whilst the residual friction is equal to the initial friction angle. For undamaged joints, the tensile normal force is limited to T_{max} and the shear force is limited to F^s_{max}, as found in the following equations:

$$T_{max} = -T \times A_c \tag{3.53}$$

$$F^s_{max} = c \times A_c + F^n \times tan\,\varphi \tag{3.54}$$

Where T is joint tensile strength the A_c is the sub-contact area, c is joint cohesive strength and φ is the joint friction angle. Tensile or shear failure of the sub-contact triggers the joint tensile and cohesive strength to become equal to zero. and consequently, the forces T_{max} and F^s_{max} to become:

$$T_{max} = 0 \tag{3.55}$$

$$F_{max}^{s} = F^{n} \times tan\,\varphi \tag{3.56}$$

For the case of tensile failure, the forces T_{max} and F_{max}^{s} are:

$$T_{max} = 0 \tag{3.57}$$

$$F_{max}^{s} = 0 \tag{3.58}$$

For the case of shear failure without tensile failure, there is slip at the joint and the joint shear force F_{i}^{s} becomes:

$$F_{i}^{s} = F_{i}^{s} \times \frac{F_{max}^{s}}{F^{s}} \tag{3.59}$$

Where F^{s} is the shear force magnitude, as such:

$$F^{s} = (F_{i}^{s} \times F_{i}^{s})^{1/2} \tag{3.60}$$

Upon shear failure, the joint normal force must be corrected to account for dilation taking place because of slip. The joint normal displacement due to dilation, $\Delta U^{n}(dil)$ from calculated from:

$$\Delta U^{n}(dil) = \Delta U^{s} \times tan\,\psi \tag{3.61}$$

Where ψ is the dilation angle and the magnitude of the joint shear displacement ΔU^{s} are given from:

$$\Delta U^{s} = (\Delta U_{i}^{s} \times \Delta U_{i}^{s})^{1/2} \tag{3.62}$$

Finally, the joint normal force is given as such:

$$F^{n} = F^{n} + K_{n} \times A_{c} \times \Delta U^{n}(dil) \tag{3.63}$$

Boundary conditions

During step two, the boundary conditions of the structure are imposed by defining the fixity of the geometric model. In *3DEC,* this is carried out by kinematically restraining the degrees of freedom of blocks, in two manners: a) by assigning a zero velocity to the blocks that are to be fixed along the x, y, and z-axes; and b) by assigning the command *fix* to the blocks that are to be fixed.

Loading protocol

During step three, the loading protocol of the model is defined, which consist of three main aspects: a) load application; b) monitored point definition; and c) unbalanced force monitoring, which is detailed in the forthcoming paragraphs accordingly.

Load application: Various load applications are considered for masonry structures in this paragraph, among which: a) a quasi-static point load; b) a two-dimensional tilt plane analysis; and c) a three-dimensional tilt plane analysis.

For a quasi-static point load, a loading element is placed in the required loading position. The load is applied by gradually altering the loading element's velocity in increments. The applied load P_l is calculated by retrieving the contact forces between the loading element and the structure. Starting from a value of P_l equal to 0 (no loading), the application of constant velocity leads to the increase of the load until the maximum value is reached $P_{l,max}$ which is the collapse load.

On the other hand, for a two-dimensional tilt plane analysis, the rotational angle of the plane (θ_t) is applied in the form of a horizontal acceleration of equal to $\lambda_h \times g$ (see equation (3.64)) and altering the vertical acceleration of gravity from g to a magnitude of $\lambda_v \times g$ (see equation (3.65)). The horizontal and vertical collapse multipliers λ_h and λ_v were obtained from the following:

$$\lambda_h = sin(\theta_t) \tag{3.64}$$

$$\lambda_v = cos(\theta_t) \tag{3.65}$$

For a three-dimensional tilt plane analysis, the azimuth of theoretical rotation must also be defined. This is effectively the direction of the loading imposed on the structure. Tilt-plane loading is thus defined by the following definitions:

$$g_{hx} = g \cdot \lambda_h \cdot cos\,\psi \tag{3.66}$$

$$g_{hy} = g \cdot \lambda_h \cdot sin\,\psi \tag{3.67}$$

$$g_{vz} = g \cdot \lambda_v \tag{3.68}$$

In particular, equations (3.66), (3.67) and (3.68) describe the x-axis horizontal component, y-axis horizontal component, and vertical component of gravity, respectively. So, for any given azimuth of rotation ψ, the rotational angle (θ_t) is proportional to the horizontal component of gravity applied on the structure. The resulting destabilisation is common with that of a tilt-table, parallel to the azimuth of rotation (ψ). In any case of a tilt plane analysis, starting from a value of θt equal to 0 (no inclination), step-by-step, increments are added to the inclination angle (θ_t) until the maximum value is reached $\theta_{t,max}$, which is the critical inclination angle. Finally, to facilitate result interpretation, the critical inclination angle multiplier $\lambda_{h,max}$ is also employed as an index of load-bearing capacity.

Monitored point selection and displacement monitoring: During the simulation, the displacements of selected points are monitored (termed monitoring points). The strategic selection of these monitored points is paramount to ensure reliable information about structural behaviour, including cases of global and local failure. Furthermore, any given monitored points A, B and C the displacement corresponding to the structure's collapse load, termed the critical horizontal displacements $U^A_{h,max}$, $U^B_{h,max}$ and $U^C_{h,max}$ are a metric of quantifying the structures deformation capacity.

Unbalanced force monitoring: During structural analysis, the unbalanced inertial force (Itasca, 2019a) is a metric employed to evaluate the mechanical static equilibrium state of the model (and subsequent occurrence of the joint slip or plastic flow). Static equilibrium of the model is achieved when either the net nodal force vectors at each block centroid or gridpoint are equal to zero and this is monitored in form of: a) the maximum nodal force vector termed the "unbalanced" or "out-of-balance" force; alternatively, b) the ratio of the unbalanced force towards the representative forces of the system, termed "unbalanced force ratio". The unbalanced force will never be equal to zero, yet when a small value (concerning the system's forces) is reached after multiple consequent calculation cycles, then the equilibrium is achieved. Conversely, when the unbalanced force tends to a constant nonzero value, and the unbalanced force ratio is relatively high (e.g. 0.1), this is an indication of the either: a) joint slip occurrence; or b) block failure and plastic flow occurrence within the model. During the structural analysis of this investigation, an unbalanced force ratio equal to 1e-4 is employed which effectively means that during the simulation, loading can only proceed when the unbalanced force ratio is smaller to or equal to 1e-4.

Material properties

During step four, the material properties of the numerical model are defined. For rigid blocks and the block constitutive model, the only user-defined block material property is the density ρ_b. For deformable blocks, the Young modulus $E_b$and shear modulus G_b $a$re also needed. When using dry-mortared joints, the joint material properties are the joint normal and shear stiffnesses K_n, K_s and the joint friction angle, φ. When using mortared-joint, the joint tensile and cohesive strengths T, c are also included in the numerical model.

Material properties are defined in two manners: a) directly from small-scale experiments with materials of analysed masonry structure itself; b) from the so-called iterative calibration procedure in which by the load-displacement curve and failure mode of the numerical model are calibrated with existing experimental data; and c) from existent numerical investigations with common masonry and employed numerical method.

With regards to the derivation of material properties, some relations have been formulated in literature. For the case of mortared blocks, where E_b, E_m, G_b and G_m are the brick and mortars' Young and shear moduli and $h_m$is the mortar thickness, the joint initial values of the stiffnesses are defined as such (Lourenco, 1996):

$$K_n = \frac{E_b \times E_m}{h_m \times (E_b - E_m)} \tag{3.69}$$

$$K_s = \frac{G_b \times G_m}{h_m \times (G_b - G_m)} \tag{3.70}$$

These values often form the first of the iterative calibration procedure. For the case of dry-jointed masonry structures, since the mortar is inexistent, the initial values have to be retrieved from previous investigations or estimated at an initial value. However, for dry-jointed masonry structures, the joint stiffnesses can be related by the following, according to Sarhosis *et al.* (2015):

$$K_s = K_n/2.38 \tag{3.71}$$

Structural analysis

During step five, structural analysis is carried out which concerns assessing the structural capacity of the developed numerical model through various metrics of the structural behaviour. These are herein defined as well as the measures of structural behaviour uncertainty.

Structural behaviour indices: The metrics of structural behaviour proposed in this investigation were divided into two, the so-called invariable and variable structural behaviour indices. The variable structural behaviour indices are the: a) collapse load/ load multiplier; b) the load per hinge/ crack formation; and c) the structural stiffness at the first hinge/crack formation. These are calculated according to the previous loading protocol and load-displacement curves. The variable structural behaviour indices are the: a) joint forces, stresses and displacements; and a) block forces, stresses and displacements. These are calculated with a *FISH* script, such as that of Section A.1.3. In Figure 3-18 all the proposed measures of structural analysis are presented.

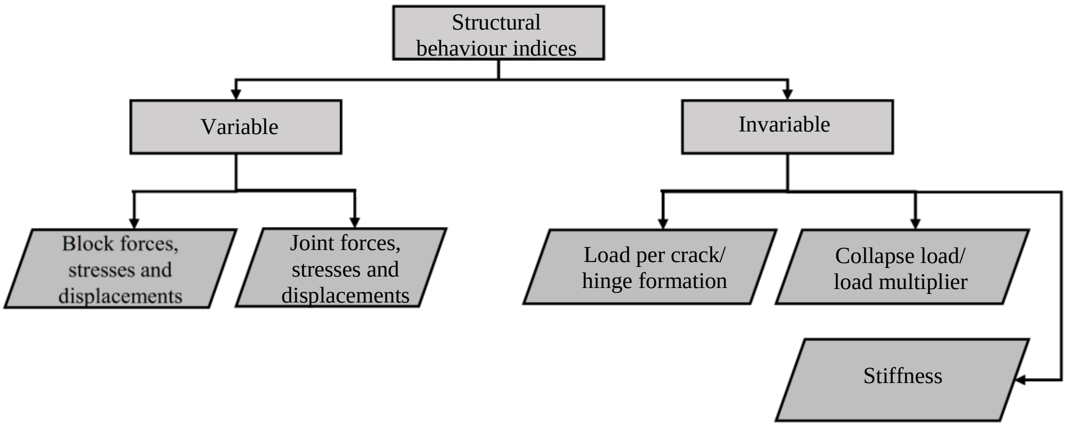

Structural behaviour indices
Variable
Invariable
Block forces, stresses and displacements
Joint forces, stresses and displacements
Load per crack/ hinge formation
Collapse load/ load multiplier
Stiffness

Figure 3-18: Proposed structural behaviour indices.

Structural behaviour uncertainty: In the case of comparing multiple numerical models from various sources (i.e. from various geospatial techniques or of varying numerical properties), relative structural behaviour discrepancies are calculated. This is done by structural analysis of each numerical model and comparison with its corresponding reference. Where y^{ref} and y are structural behaviour indices of the reference and assessed numerical model respectively, the relative uncertainty (RU), absolute uncertainty (AU) and normalised uncertainty (NU) are calculated with the following formulae:

$$RU = \frac{(y^{ref} - y)}{y^{ref}} \times 100 \tag{3.72}$$

$$AU = y^{ref} - y \tag{3.73}$$

$$NU = (y^{ref} - y)/max\,(y^{ref}) \tag{3.74}$$

3.4 Summary

Chapter 3 has presented the three-stage methodological framework entailing the structural surveying, geometric model development and structural analysis of masonry structures. The first stage of the framework concerned structural surveying with various geospatial techniques. The second stage of the framework entailed geometric model development through, three workflows, the: point-based, image-based, and cloud-based workflows respectively. This was the core component of the methodological framework which permits the semi-automated and geometrically accurate numerical modelling. The final stage entails structural analysis with the DEM, whilst employing a given geometric model. In the forthcoming chapters, the three distinct approaches (of Figure 3-1) will be implemented on relative case studies.

Chapter 4. Quantifying the effect of geometric uncertainty on the robustness of structural analysis[6]

In the previous Chapter, the three-stage methodological framework was presented. As discussed, various geospatial techniques and approaches can be employed for the structural analysis of regular masonry structures. However as found in Chapter 2, a lack of comprehensive understanding remains on the effect of geometric uncertainty and as a consequence, the justification of the employment of accurate geometric models still lacks clarification.

Specifically, it was demonstrated in Section 2.2.3 that the employment of ad-hoc geometric models can significantly affect the structural behaviour. Concurrently, recent evidence also suggests that geometric uncertainty between geospatial techniques can be equally important. For instance, Morer *et al.* (2013) developed geometric models of masonry arches from a total station and SfM photogrammetry. An error of the arch span equal to 2.32% was associated with an error in collapse load and the critical load position of 4% and 6% respectively. Furthermore, Riveiro *et al.* (2013) found the collapse load and critical load position varied by up to 19% between geometric models of arches developed by SfM photogrammetry and GPR. These studies demonstrate that even when employing accurate geometric models, geometric uncertainty exists and can significantly affect the structural behaviour, especially the collapse load and critical load position (i.e. the ultimate limit states). However, a lack of comprehensive understanding remains on the effect of geometric uncertainty for both the serviceability (i.e. in-service loads, stiffness and internal forces) and ultimate limit states of masonry structures. Moreover, the critical types and quantities of geometric uncertainty, significantly affecting the structural behaviour of masonry structures, remain unknown.

[6] The study in this Chapter has been submitted to the following research journal paper:

Kassotakis, N., Sarhosis, V., Peppa, M.V. and Mills, J. (2020b). Quantifying the effect of geometric uncertainty on the structural behaviour of arches developed from tape measurement and Structure-from-Motion (SfM) photogrammetry. Engineering Structures, - Doi: under review – response from reviewers: minor corrections

This Chapter quantifies the effect of geometric uncertainty on the structural behaviour of geometric models developed from SfM photogrammetry and direct measurements (with tape measurements). To pursue this aim, the manual image-based-approach (with SfM photogrammetry) and the point-based approach (with tape measurements) of the methodological framework are implemented according to Figure 4-1. Firstly, the structural surveying of 25 arch specimens carried out with SfM photogrammetry is reported including imagery acquisitions with the Samsung S7 smartphone camera and the procedures leading up to the acquirement of an orthomosaic within *Metashape*. It is noteworthy that the structural surveying with tape measurements was carried out in a previous investigation (Stockdale et al., 2018), is also described here. Next, geometric models of the 25 arch specimens are developed from tape measurements and SfM photogrammetry. The geometric uncertainty between the geometric models of SfM photogrammetry and tape measurements is calculated here. The variables of geometric uncertainty are the: a) joint inclination angle, joint midpoint location and joint length; and b) block volume and block centroid location. It should be noted that the determination of the exact reason behind the geometric uncertainty is a difficult task, which exceeds the scope of this investigation; which is to merely document the existing geometric uncertainty and the resulting structural behaviour uncertainty. Following the numerical model development, structural analysis follows and the structural behaviour uncertainty between the geometric model of SfM photogrammetry and tape measurements is calculated. The variables of structural behaviour uncertainty investigated are the: a) stiffness; b) load multipliers; and c) normal forces between joints at each hinge formation (i.e. the 1st,2nd,3rd, and 4th). Finally, a correlation between the geometric uncertainty and structural behaviour uncertainty is investigated.

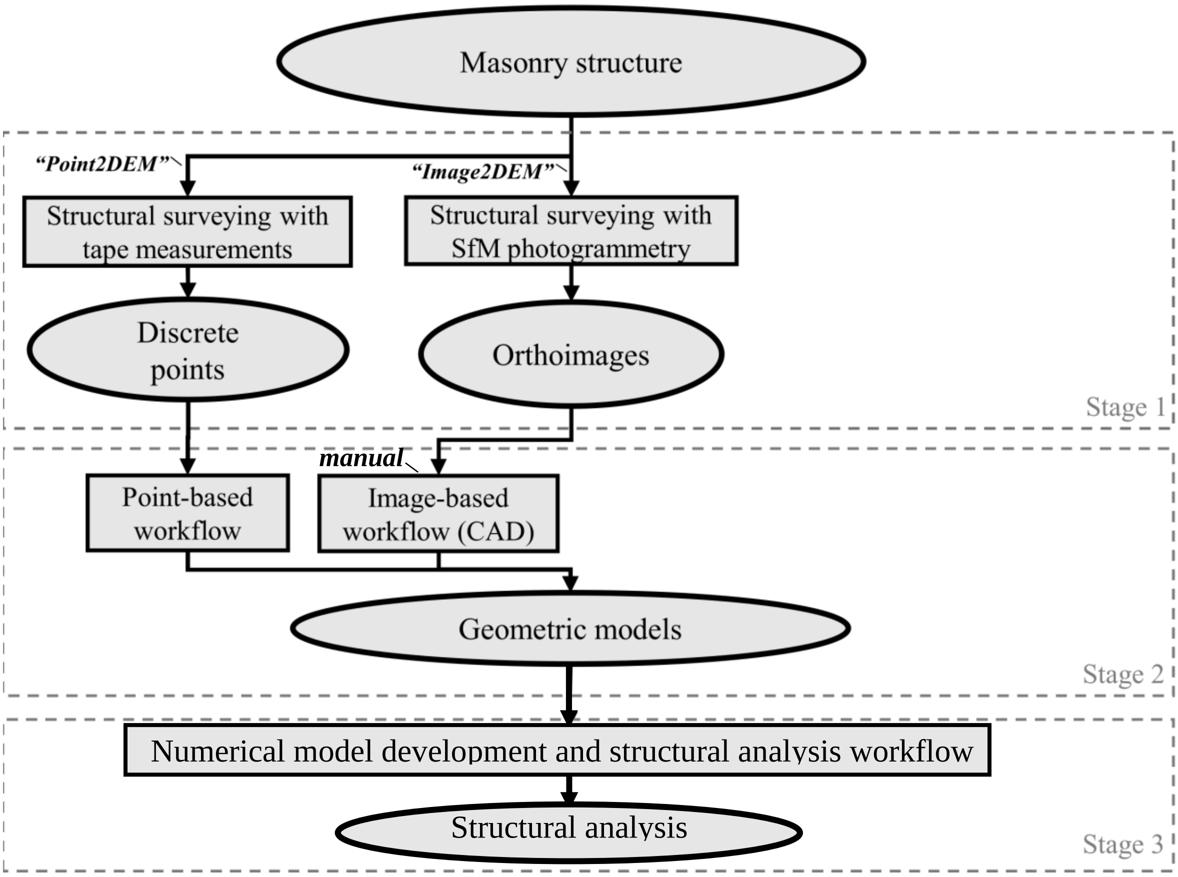

Masonry structure
"Point2DEM"
"Image2DEM"
Structural surveying with tape measurements
Structural surveying with SfM photogrammetry
Discrete points
Orthoimages
Stage 1
manual
Point-based workflow
Image-based workflow (CAD)
Geometric models
Stage 2
Numerical model development and structural analysis workflow
Structural analysis
Stage 3

Figure 4-1: Implementation of the methodological framework in Chapter 4. The point-based and manual image-based approaches, "Point2DEM" and "manual Image2DEM" respectively.

4.1 Case study one: The experimental testing of 25 arch specimens[7]

The experimental testing of 25 arch specimens carried out at Newcastle University by Stockdale *et al.* (2018). It was employed as a case study due to the arches' discontinuous nature. Tilt-table analyses were carried out on each arch specimen (termed arch hinge-sets in Stockdale *et al.* (2018)) to investigate the influence of geometry upon their structural behaviour. The arch was constructed of 25 wooden blocks in a method shown in Figure 4-2a-d.

[7] The experimental testing and structural surveying data with direct measurement herein presented were captured within a previous investigation of Stockdale *et al.* (2018).

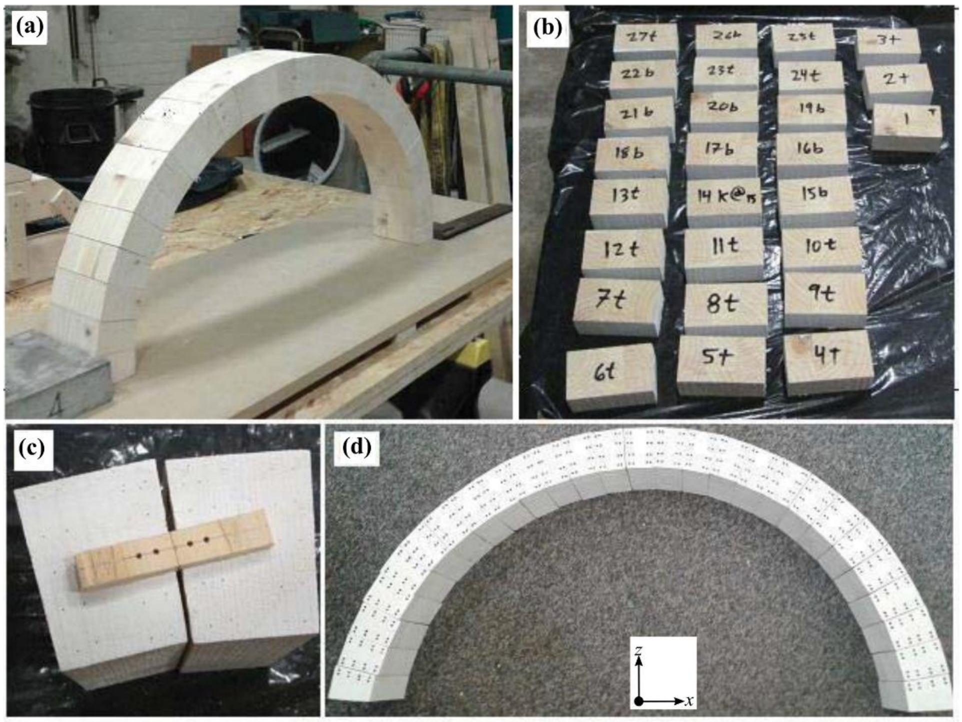


Figure 4-2: Construction of initial arch assembly (Stockdale et al., 2018): (a) block cutting; (b) block measurement and denomination; (c) block alignment; and (d) initial arch assembly.

After the construction of the initial arch assembly, 25 arch specimens were developed, consisting of 3 free-moving blocks and 4 open joints. This was by various permutations of joined blocks and open joints according to Table 4-1). Indicatively, arch specimen number one is shown in Figure 4-3c-d with the variable and invariable geometrical properties.

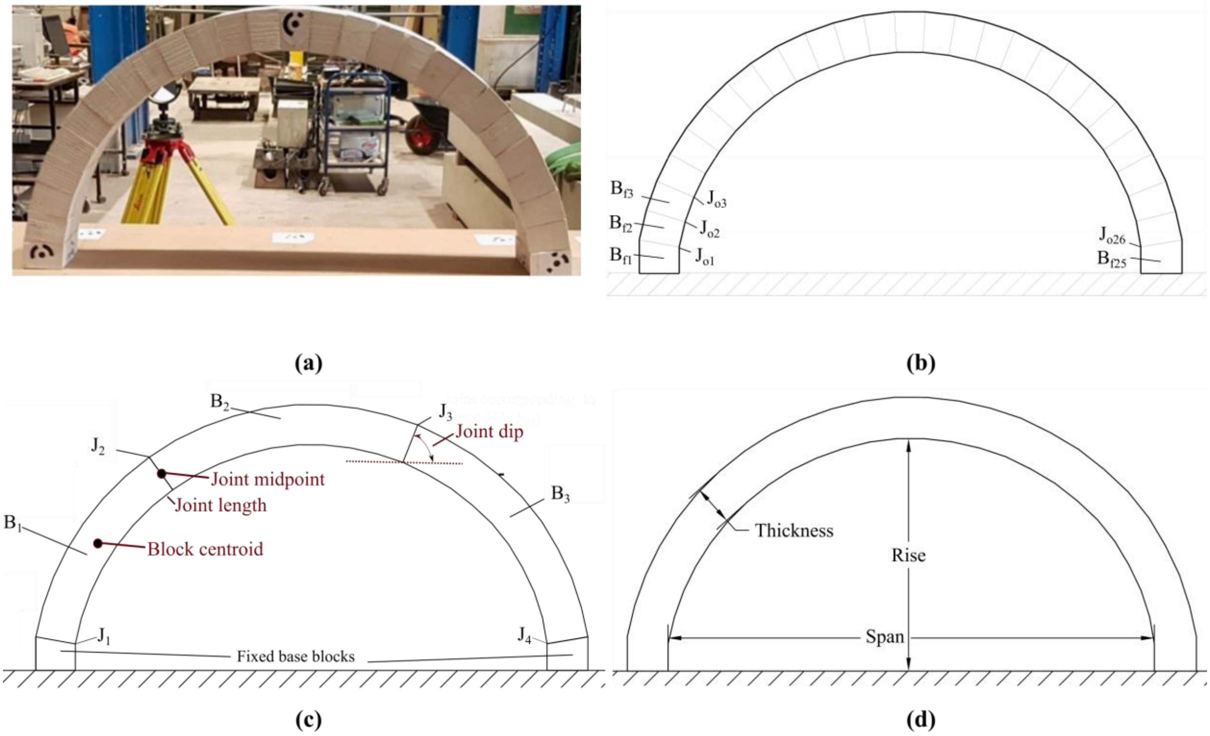


Figure 4-3: Construction of 25 arch specimens. Initial arch assembly: (a) in Newcastle University structures lab (Stockdale et al., 2018); and (b) blocks and joints. Arch specimen number one: (c) variable geometrical properties; and (d) invariable geometrical properties.

Table 4-1: Joints per arch specimens (J_1, J_2, J_3, J_4) corresponding to the joints of the initial arch assembly (J_{o1}, J_{o2}, $J_{o3,\ldots,}$ J_{o26}) (Stockdale et al., 2018).

Arch specimen	Joint 1 (J_1)	Joint 2 (J_2)	Joint 3 (J_3)	Joint 4 (J_4)
1	J_{o1}	J_{o8}	J_{o17}	J_{o26}
2	J_{o1}	J_{o8}	J_{o17}	J_{o25}
3	J_{o1}	J_{o8}	J_{o16}	J_{o24}
4	J_{o1}	J_{o8}	J_{o16}	J_{o23}
5	J_{o1}	J_{o8}	J_{o16}	J_{o22}
6	J_{o2}	J_{o8}	J_{o17}	J_{o22}
7	J_{o2}	J_{o9}	J_{o17}	J_{o23}
8	J_{o2}	J_{o9}	J_{o17}	J_{o24}
9	J_{o2}	J_{o9}	J_{o17}	J_{o25}
10	J_{o2}	J_{o9}	J_{o18}	J_{o26}
11	J_{o3}	J_{o10}	J_{o18}	J_{o26}
12	J_{o3}	J_{o10}	J_{o18}	J_{o25}
13	J_{o3}	J_{o10}	J_{o17}	J_{o24}
14	J_{o3}	J_{o9}	J_{o17}	J_{o23}
15	J_{o3}	J_{o9}	J_{o18}	J_{o22}
16	J_{o4}	J_{o10}	J_{o18}	J_{o22}
17	J_{o4}	J_{o10}	J_{o19}	J_{o23}
18	J_{o4}	J_{o10}	J_{o19}	J_{o24}
19	J_{o4}	J_{o11}	J_{o19}	J_{o25}
20	J_{o4}	J_{o11}	J_{o19}	J_{o26}
21	J_{o5}	J_{o12}	J_{o20}	J_{o26}
22	J_{o5}	J_{o11}	J_{o20}	J_{o25}
23	J_{o5}	J_{o11}	J_{o20}	J_{o24}
24	J_{o5}	J_{o11}	J_{o19}	J_{o23}
25	J_{o5}	J_{o11}	J_{o19}	J_{o22}

4.2 Stage 1: Structural surveying

4.2.1 Structural surveying with tape measurements

During the experimental testing, the initial arch assembly was measured (and not the 25 arch specimens *per se)*. Specifically, the structural surveying with tape measurements consisted of measurements of the invariable geometrical properties and variable properties of the 25 free-moving blocks and open joints. In Table 4-2 the invariable geometrical properties (span, rise, thickness as shown in Figure 4-3c) are reported. These were also common for the arch specimens. In Table 4-3 and Table 4-4 the joint and block properties (as shown in respectively) are reported also reported.

Table 4-2: Invariable geometrical properties of the initial arch assembly (Stockdale et al., 2018).

Arch Shape	Span (cm)	Rise to Span Ratio	Width (cm)	Thickness (cm)
Semi-Circular	66.70	1:2	9.96	5.63

Table 4-3: Joint properties of the initial arch assembly (Stockdale et al., 2018).

Joint	Dip (°)	Dip dir. (°)	Joint midpoint x (cm)	Joint midpoint z (cm)	Joint length (cm)
1	9.60	90.00	2.91	3.92	5.88
2	13.94	90.00	3.75	8.12	5.65
3	21.75	90.00	5.03	12.05	5.65
4	28.15	90.00	6.81	15.77	5.64
5	34.72	90.00	8.91	19.31	5.65
6	42.76	90.00	11.49	22.54	5.65
7	49.36	90.00	14.51	25.37	5.64
8	55.61	90.00	17.76	27.91	5.64
9	61.53	90.00	21.31	30.02	5.64
10	68.37	90.00	25.06	31.88	5.65
11	75.15	90.00	28.95	33.14	5.63
12	80.35	90.00	32.96	34.05	5.64
13	86.68	90.00	36.99	34.51	5.64
14	86.58	270.00	41.08	34.53	5.64
15	79.50	270.00	45.14	34.03	5.64
16	73.17	270.00	49.11	33.04	5.64
17	67.44	270.00	52.97	31.66	5.64
18	60.88	270.00	56.67	29.90	5.64
19	54.21	270.00	60.11	27.68	5.64
20	48.05	270.00	63.28	25.13	5.64
21	41.74	270.00	66.15	22.25	5.64
22	35.43	270.00	68.68	19.08	5.64
23	29.12	270.00	70.85	15.65	5.64
24	22.81	270.00	72.62	12.00	5.64
25	16.50	270.00	73.99	8.17	5.64
26	10.19	270.00	74.93	4.22	5.64

Table 4-4: Block properties of the initial arch assembly (Stockdale et al., 2018).

Block	Block centr. x (cm)	Block centr. z (cm)	Volume (cm^3)
1	3.32	6.11	229.22
2	4.31	10.12	230.34
3	5.85	13.94	229.47
4	7.79	17.58	229.18
5	10.13	20.98	230.50
6	12.95	24.01	230.62
7	16.09	26.70	229.59
8	19.50	29.02	230.46
9	23.16	31.02	233.03
10	26.98	32.58	227.49
11	30.94	33.66	229.18
12	34.97	34.36	226.21
13	39.03	34.60	227.42
14	43.12	34.36	228.34
15	47.15	33.60	228.00
16	51.06	32.41	227.83
17	54.85	30.85	228.43
18	58.43	28.85	228.12
19	61.74	26.46	226.66
20	64.76	23.74	226.21
21	67.47	20.71	226.21
22	69.82	17.40	226.21
23	71.80	13.85	226.21
24	73.38	10.11	226.21
25	74.53	6.21	226.21

4.2.2 Structural surveying with SfM photogrammetry

The following paragraphs report the results of the structural surveying carried out with the Samsung S7 smartphone sensor (SM-G930F) and *Metashape* is reported, according to the flowchart of 3.1.2. All the times reported are with computational resources of a consumer-grade workstation with an Intel(R) Xeon(R) CPU @ 3.00 GHz processor with 64 GB memory.

Image acquisition

Image capture consisted of four vertical and two oblique (-45 and 45 degrees) image capture sets at six different heights above floor level. Due to the requirements of the GSD_{max} , a WD of 0.5 m from the face of the arch was calculated from Table 3-3 as suitable. The processing details of the *Metashape* project created are listed in Table 4-5. Given that a high image overlap alongside depth variation within the scene was are an important consideration to ensure that systematic errors will be removed from SfM outputs, the number of optical rays per tie point of Table 4-5 shows a relatively strong and sufficient imaging configuration.

Table 4-5: Processing details of Metashape project using the smartphone sensor (SM-G930F).

Predefined WD (m)	No of images of the region of interest	Estimated GSD_{max} (mm/pix)	Optical rays per point
0.5	166	0.138	3.2

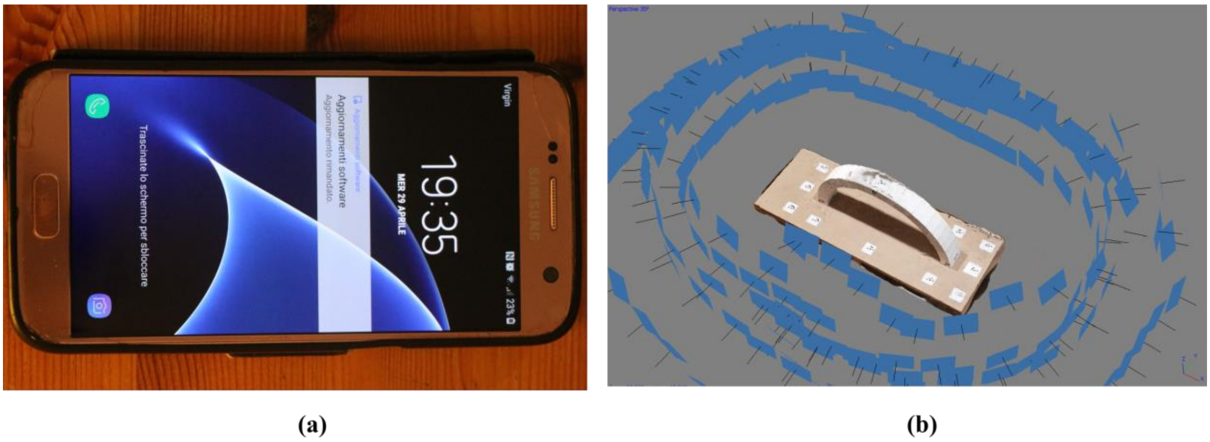


Figure 4-4: Structural surveying with SfM photogrammetry: (a) camera sensor; and (b) image capture network.

The self-calibration properties of the images captured with are found in Table 4-6, where: f is the focal length; C_x and C_y are the principle point's position on the x and y-axis respectively; B_1 is the affinity; B_2 is the non-orthogonality; K_1, K_2 and K_3 are the radial distortion parameters; and P_1 and P_2 are the decentring distortion parameters.

Table 4-6: Self-calibration properties of smartphone sensor (SM-G930F) from Metashape (pixels).

	f	C_x	C_y	B_1	B_2	K_1	K_2	K_3	P_1	P_2
Value	3.1x103	-15.51	68.57	-0.15	2.25	-0.07	0.54	-2.08	2.57	2.1x10-3
Error	0.68	0.44	0.54	0.64	0.38	0	0.01	0.03	0.03	4.7x10-5

Sparse point cloud reconstruction and georeferencing

The sparse point cloud is shown in Figure 4-6a which was reconstructed according to Section 3.1.2 as detailed in Table 4-7. After the sparse point cloud reconstruction, georeferencing was carried out by introducing the GCPs to *Metashape* in a text format. A total of 12 GCPs were surveyed with a Leica MS60 total station (measured by intersection) with a maximum RMSE error equal to 3 mm. Eight points were employed as control points and the remaining four as checkpoints, with the respective errors reported in Table 4-8. The georeferencing errors within *Metashape* are reported in Table 4-9.

Table 4-7: Sparse point cloud reconstruction properties from Metashape: smartphone.

Parameter	
Points	18,762 of 227,318
Key points	No
RMS reprojection error	0.36
Max reprojection error	1.00
Mean key point size	3.37
Average tie point multiplicity	3.20
Accuracy	High
Generic preselection	Yes
Key point limit	400,000
Tie point limit	10,000
Matching time	11 minutes 54 seconds
Alignment time	2 minutes 3 seconds
Optimisation parameters	$f, C_x, C_y, B_1 , B_2, K_1, K_2, K_3, P_1, P_2$
Optimisation time	4 seconds

Table 4-8: Ground control point locations measured with the Leica DS60 total station.

	Target ID	X(mm)	Y(mm)	Z(mm)	RMSE error (mm)
Control points	11	-105.30	28.30	0.00	3.00
	12	877.00	35.80	-5.50	3.00
	13	27.40	-15.00	19.00	0.00
	14	16.20	77.90	16.60	1.00
	18	743.90	81.60	13.10	1.00
	19	-1.40	26.60	26.50	2.00
	20	50.30	36.60	9.90	0.00
	23	772.50	38.90	19.30	3.00
	24	721.40	43.10	12.00	1.00
Check points	2	43.60	178.80	-1.00	3.00
	3	385.30	-116.80	-2.00	3.00
	4	379.20	180.20	-3.00	3.00
	5	722.40	-114.80	-4.00	3.00

Table 4-9: Georeferencing errors of targets from Metashape: smartphone.

	Target ID	X error (mm)	Y error (mm)	Z error (mm)	Total error (mm)	Image error (pix)
Control points	11	1.81	1.23	-1.02	2.42	0.68
	12	-0.17	-0.72	-1.24	1.44	0.79
	13	-1.63	-1.62	1.09	2.55	0.65
	14	3.51	3.41	0.92	4.98	0.33
	18	0.06	-0.09	-0.09	0.14	0.19
	19	-0.07	-0.24	0.32	0.40	0.00
	20	0.68	1.24	-0.03	1.41	0.22
	23	-4.25	-3.45	0.37	5.49	0.39
	24	2.14	1.93	0.78	2.99	0.49
	Mean	1.81	1.23	-1.02	2.42	0.68
Check points	2	2.57	1.19	-3.09	4.19	0.62
	3	0.07	0.68	-0.31	0.75	0.41
	4	1.16	0.12	-3.29	3.49	1.00
	5	-0.90	0.03	0.13	0.91	0.36
	Mean	1.48	0.69	2.26	2.79	0.72

Dense point cloud reconstruction

The dense point cloud reconstructed is shown in Figure 4-6b. From the options of smoothness and detail within *Metashape*, the high-quality point cloud reconstruction setting was used in conjunction with the aggressive smoothness level. Table 4-10 reports the properties of the dense point cloud reconstruction.

Table 4-10: Dense point cloud reconstruction properties: smartphone.

Parameter	
Points	3,567,246
Point cloud reconstruction time	High
Filtering mode	Aggressive
Processing time of depth maps generation parameters	1 hour 32 minutes
Processing time of dense cloud generation parameters	12 minutes 8 seconds

Orthomosaic development

The 3D model and orthomosaic of the arch are shown in Figure 4-6c-d respectively. The 3D model was based on a mesh of 30,000 faces and the orthomosaic resolution equal to 0.14 mm. In Table 4-11, the properties of the mesh and orthomosaic development are listed.

Table 4-11: Mesh and orthomosaic properties: smartphone.

	Parameter	
Mesh	Faces	29,999
	Vertices	15,265
	Surface type	Arbitrary
	Source data	Dense cloud
	Interpolation	Enabled
	Strict volumetric masks	No
	Processing time	3 minutes 26 seconds
Orthomosaic	Size	12,816 x 4,528
	Coordinate system	Local Coordinates (m)
	Blending mode	Mosaic
	Surface type	Mesh
	Processing time	3 minutes 5 seconds

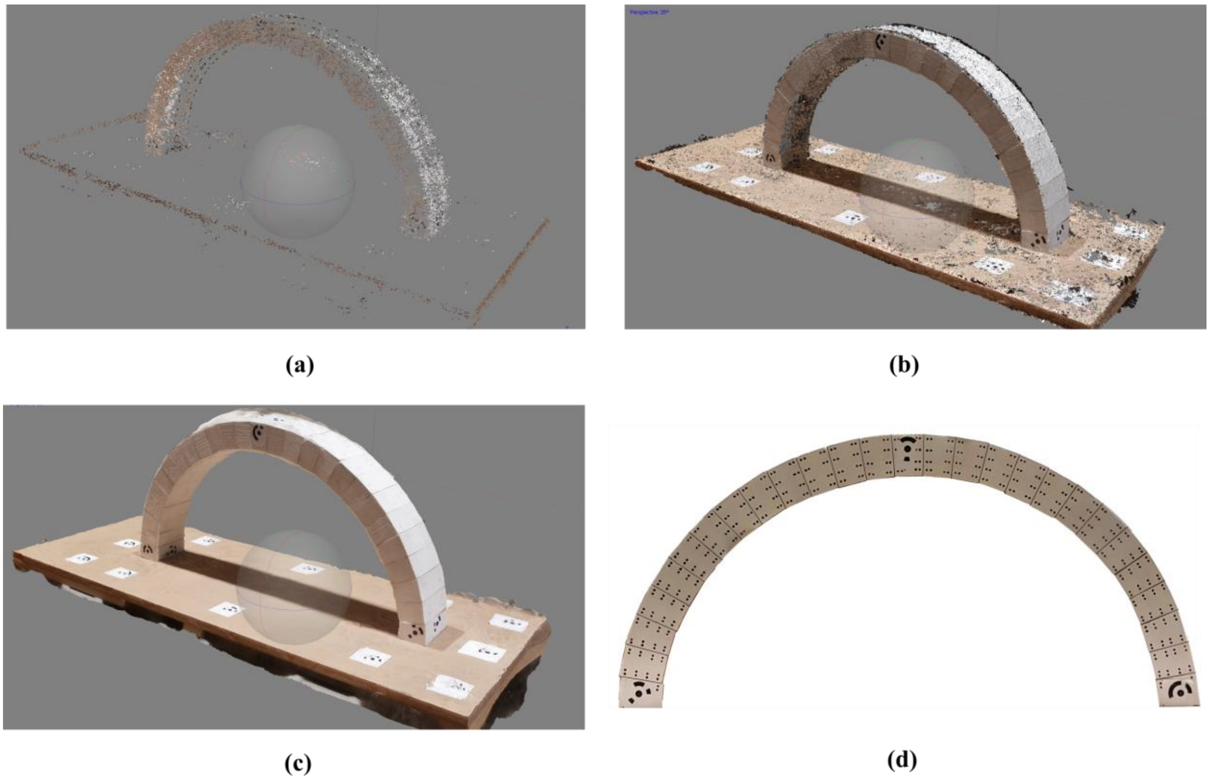


Figure 4-5: Structural surveying with SfM photogrammetry and smartphone camera: (a) sparse point cloud; (b) dense point cloud; (c) 3D model; and (d) orthomosaic.

4.3 Stage 2: Geometric model development

The following paragraphs report the results of the implementation of the image-based and point-based geometric model development workflows with tape measurements and SfM photogrammetry, respectively. After, the geometric models were aligned (as shown in Figure 4-6b-d) and geometric assessment followed, according to the workflow of 3.2.2. Concerning the geometric assessment, the invariable geometrical properties were the arch specimens' spans and rises while the variable geometrical properties were: a) the joint length, joint midpoint location and joint dip (joint inclination angle); b) the block volume and block centroid location (as shown in Figure 4-8d). For all the calculations of the geometric uncertainty, the normalised uncertainty (NU) of (3.11) was used, where $x^{ref}$and x were the geometrical properties of the geometric model from tape measurements and SfM photogrammetry respectively (i.e. the geometric models of the tape measurements were the reference, while the SfM photogrammetry models were assessed). Additionally, max (x^{ref}) of (3.11) refers to the maximum value of the property group (i.e. the arch span for the normalized uncertainty of the block centroid x-axis properties). It is additionally important to note that only the block and joint vertices of the initial arch (i.e. the arch made of free moving blocks of Figure 4-2a) were measured with both tape measurements and SfM photogrammetry. The properties each of the arch specimens (i.e. block volume, block centroid, joint dip, joint length and joint midpoint) were subsequently calculated as a function of the measured vertices

4.3.1 Geometric models from tape measurements

Geometric models from tape measurements were developed according to the point-based workflow of 3.2.1 using the discrete points. Specifically, as in the experiment, a geometric model of the initial arch assembly was developed as an assembly of un-joined blocks, as shown in Figure 4-6b using the vertices of Table 4-3 (the discrete points shown in Figure 4-6a). Then, the geometric models of each specimen were developed by joining the blocks and leaving un-joined joints according to Table 4-1, per arch specimen. A *FISH* script was employed for this, which is reported in Section A.1.4.

4.3.2 Geometric models from SfM photogrammetry

Geometric model development

Geometric models of the SfM photogrammetry were developed following the image-based workflow of 3.2.2 employing the manual CAD-based block segmentation, as shown in Figure 4-6c. As with the geometric models of tape measurements, firstly a geometric model of the initial arch assembly was developed as shown in Figure 4-6d, then each arch specimens were developed with the same *FISH* script of Section A.1.4.

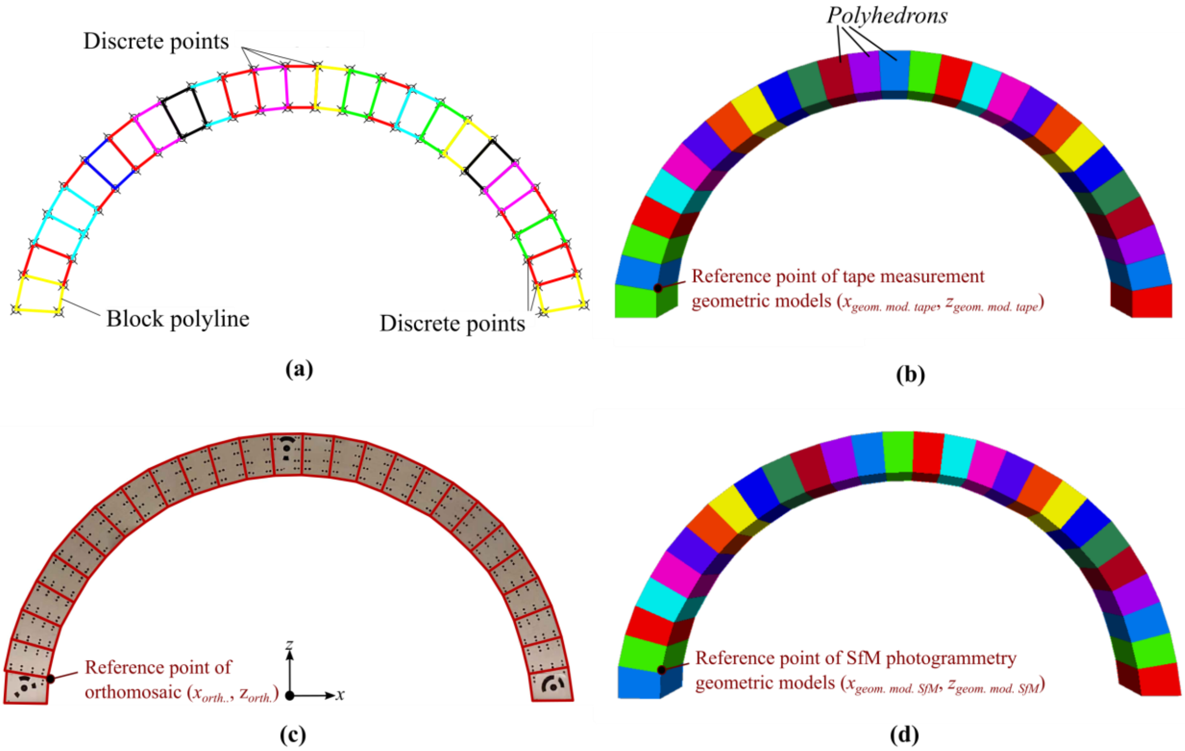

Discrete points
Block polyline
Discrete points
(a)
Polyhedrons
Reference point of tape measurement
geometric models ($x_{geom.\ mod.\ tape}$, $z_{geom.\ mod.\ tape}$)
(b)
Reference point of
orthomosaic ($x_{orth.}$, $z_{orth.}$)
z
x
(c)
Reference point of SfM photogrammetry
geometric models ($x_{geom.\ mod.\ SfM}$, $z_{geom.\ mod.\ SfM}$)
(d)

Figure 4-6: Tape measurements: (a) discrete points and marked blocks; and (b) geometric model (initial arch assembly). SfM photogrammetry: (c) manual CAD-based block segmentation; and (d) geometric model (initial arch assembly).

Geometric uncertainty between SfM and tape measurement models

The span and rise normalised uncertainties were equal to 0.6 and 1.5% respectively, for all the arch specimens. Although it is well-understood that the span and rise affect structural behaviour, they do not concern this investigation since they remain constant. In Figure 4-7a-d and Figure 4-8a-d, the magnitude and distribution of the normalised geometric uncertainty are plotted for all the joint and block parameters, per arch specimen (of which the statistical measures are found in Table B- 1 of Section B.1). Figure 4-7a-d and Figure 4-8a-d, suggest that the quantity of the geometric uncertainty was: a) significant for all the joints and block properties; and b) most significant for the joint length and block volume with minima-maxima intervals of (-4 to 8%) and (4 to 9%). Furthermore, the mostly positive normalized uncertainty of joint lengths and block volumes (shown in Figure 4-7b and Figure 4-8a), suggests that the geometric models of the SfM photogrammetry (i.e. assessed) were slightly overestimated, compared to those of tape measurements (i.e. reference). It is important to note that the nature of the geometric uncertainty is not of concern in this investigation, but only the quantity. Additionally, the cause of geometric uncertainty may have been due to: a) gross errors of tape measurement; b) errors of SfM photogrammetry measurement; and c) errors in alignment (scaling and orientation) of the geometric models. Finally, with regards to the consistency of joints errors of Figure 4-7a-d, this is because, during the experimental testing, only the block and joint vertices of the initial arch assembly were measured, as aforementioned (and not the 25 arch specimens *per se*). Since each arch specimen was measured as a four-block assembly using the blocks and joints of the initial arch assembly, multiple arch specimens shared common joints, leading to consistency in some of the joint errors.

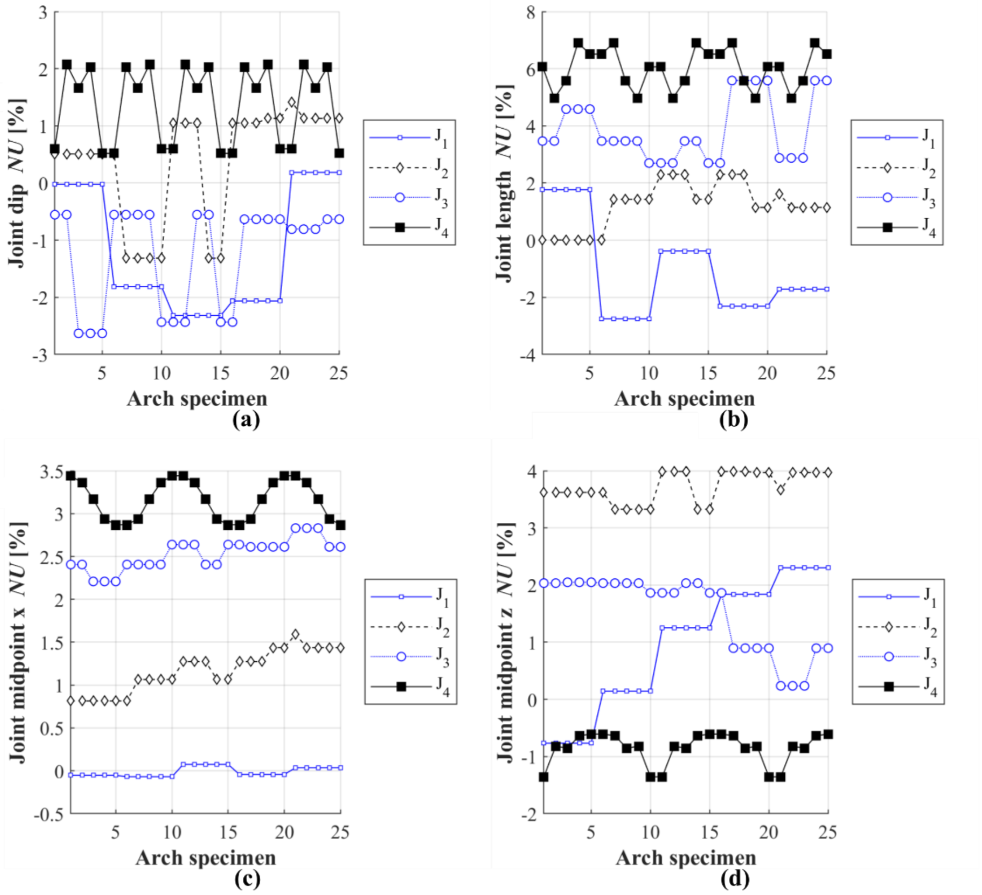


Figure 4-7: Normalised geometric uncertainty of: (a) joint dip; (b) joint length; (c) joint midpoint x; and (d) joint midpoint z.

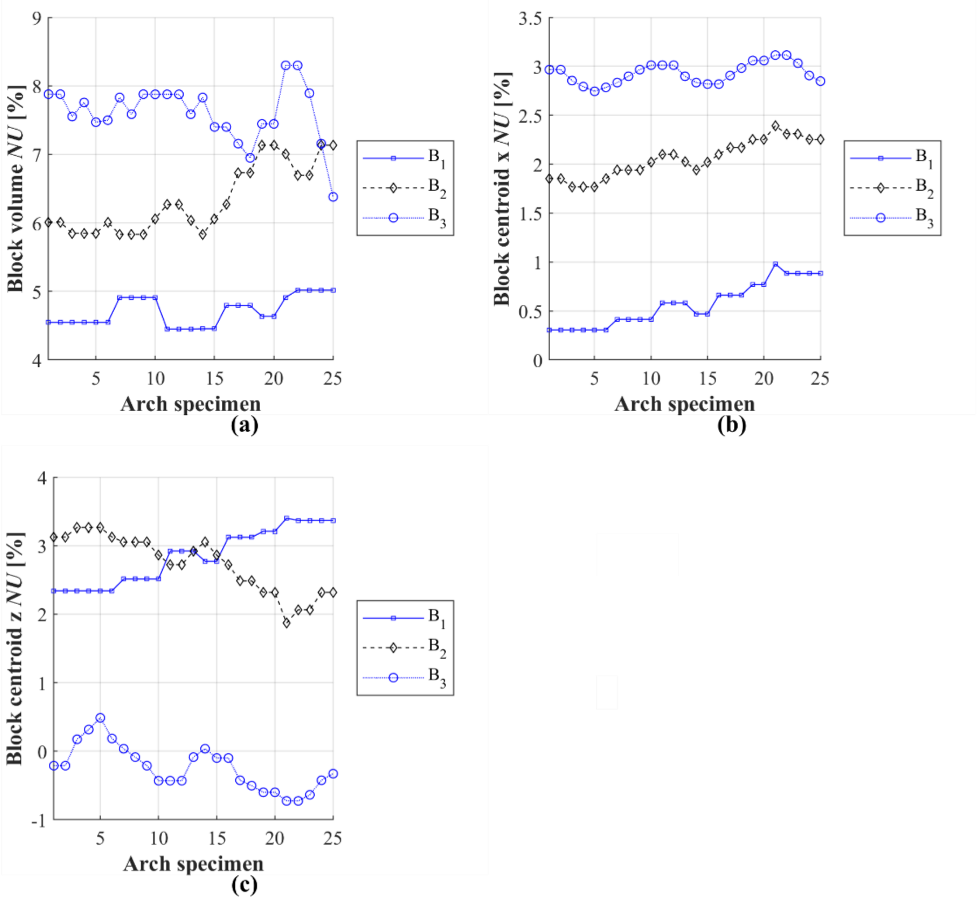


Figure 4-8: Normalised geometric uncertainty of: (a) block volume; (b) block centroid x; and (c) block centroid z.

Control of SfM photogrammetry with TLS geometric models

To obtain a better understanding of the accuracy of the SfM photogrammetry geometric models, they were additionally compared to those of a TLS survey. In specific, a further 25 geoemetric models were developed with the manual image-based approach (as shown in Figure 4-9), however not for structural analysis and only for further geometrical assessment of the SfM photogrammetey models.

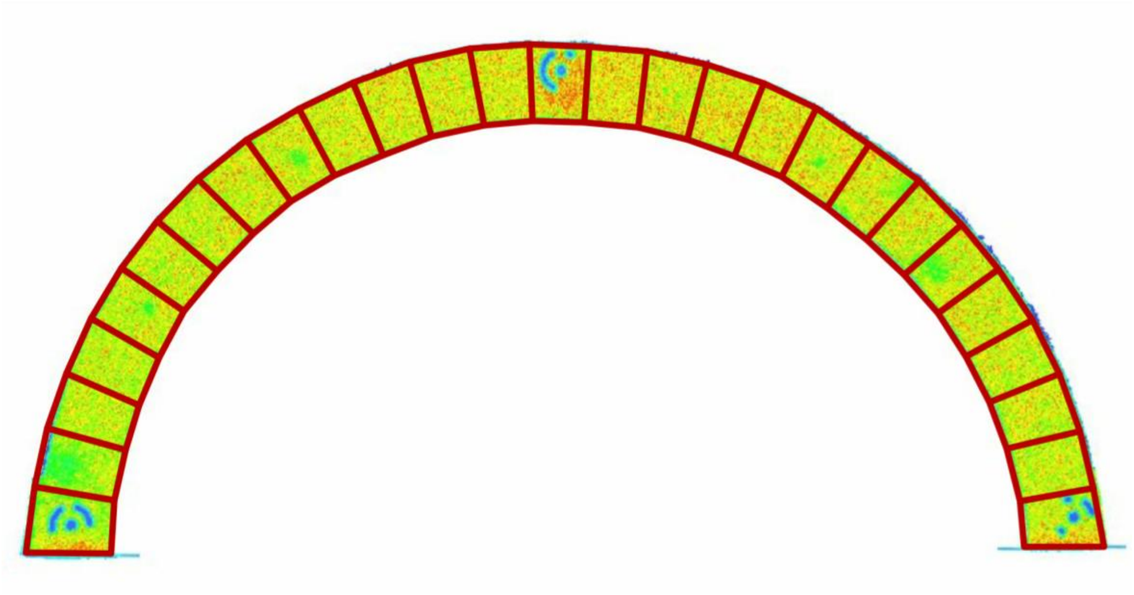


Figure 4-9: TLS data obtained for the control of SfM photogrammetry models.

In a similar manner to the previous paragraph, the span and rise normalised uncertainties were found to be equal to 3 and 1.7% respectively, for all the arch specimens Moreover, in Figure 4-10a-d and Figure 4-11a-d, the magnitude and distribution of the normalised geometric uncertainty are plotted for all the joint and block parameters, per arch specimen. Figure 4-10a-d and Figure 4-11a-d suggest that between the SfM photogrammetry and TLS geometric models, the quantity of the geometric uncertainty was: a) less significant for the joint lengths and block volumes; and b) most significant for all the joints midpoint and block centroid properties) with minima-maxima intervals of (6.8 to 12%) and (7 to 11.5%). The findings show that the SfM geometric models had a good metric agreement with the TLS, however, were not as well-aligned, as with geometric models of tape measurements.

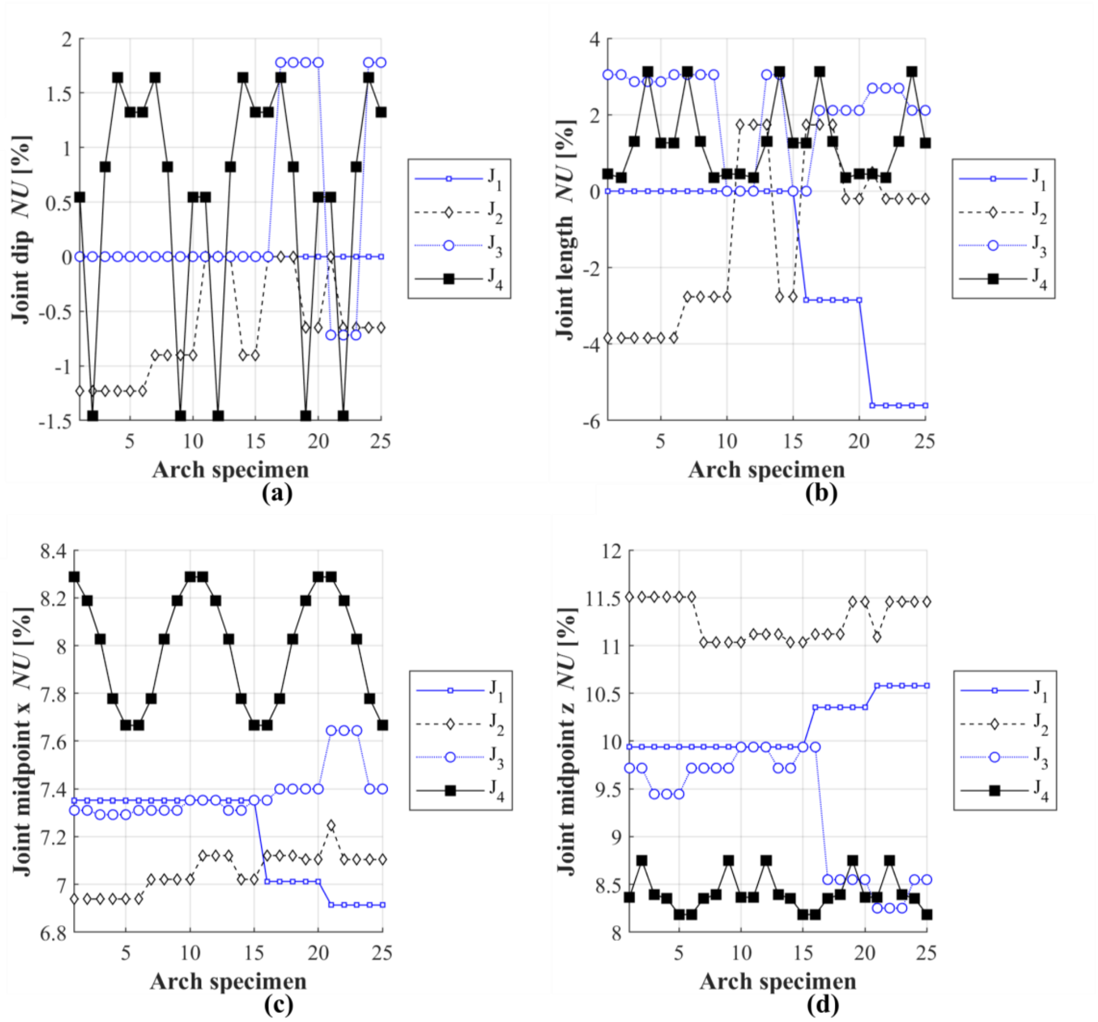


Figure 4-10: Normalised geometric uncertainty (between SfM photogrammetry and TLS) of: (a) joint dip; (b) joint length; (c) joint midpoint x; and (d) joint midpoint z.

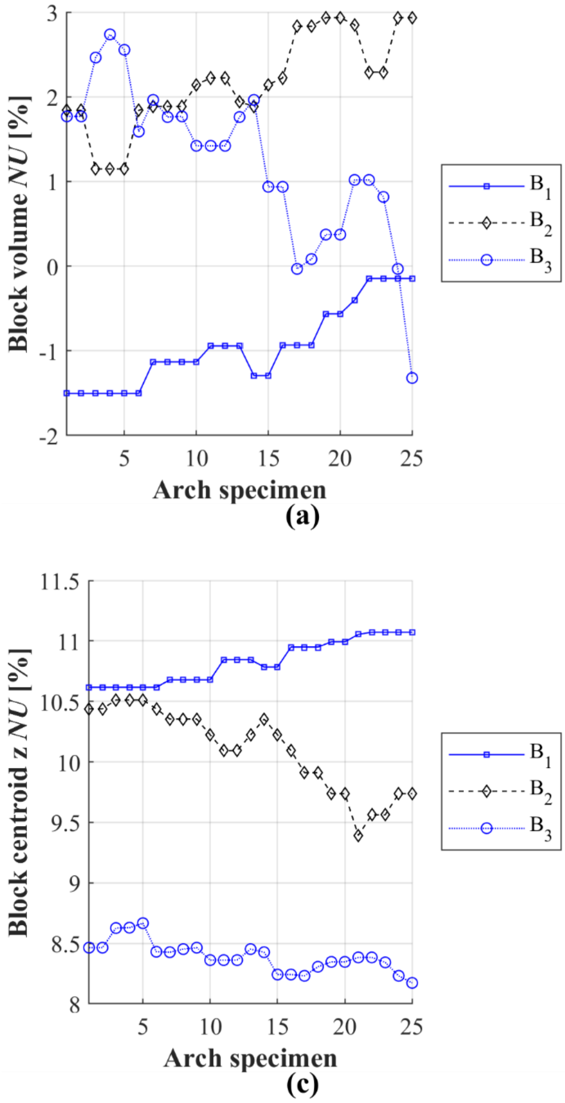


Figure 4-11: Normalised geometric uncertainty (between SfM photogrammetry and TLS) of: (a) block volume; (b) block centroid x; and (c) block centroid z.

4.4 Stage 3: Structural analysis with the discrete element method

As discussed in Section 3.3.2, geometric models are only geometrical entities. The geometric model, once employed by a numerical method and assigned numerical properties (e.g. material properties, boundary conditions etc.) becomes a numerical model. The following paragraphs detail the numerical model development with *3DEC* according to 3.3.2. Then the subsequent structural analysis follows employing geometric models developed from SfM photogrammetry and tape measurements.

4.4.1 Numerical model development and structural analysis with 3DEC

Numerical model development

Numerical model development was carried out following the workflow of Section 3.3.2. Concerning the boundary conditions, the base blocks of the numerical models were fixed, as in the experiment by applying a zero velocity to their centroid (i.e. limiting all degrees of freedom parallel to the x, y and z-axes) within the *3DEC*. The loading of the arch was according to the experimental tilt-table analysis (Dejong, 2009), applied in the numerical model, also according to Section 3.3.2. The material properties were obtained from a calibration procedure of the same Section 3.3.2. Specifically, using the geometric models of tape measurements by calibrating: a) the collapse load multiplier (synonymous with the 4th load multiplier); and b) the failure mode of the experimental-numerical model. Table 4-12 reports the calibration of the collapse loads of the numerical simulation of 10 arch specimens with the respective experimental. The errors between the experimental and numerical models are considered acceptable. In Figure 4-12a-b, the calibration of the experimental and numerical failure mode of arch specimen number six is shown. This failure mode was common for all arch specimens (a four-hinge mechanism). Table 4-13 reports the obtained properties that employed for all numerical models herein presented.

Table 4-12: Calibration of experimental (Stockdale et al., 2018) and numerical collapse loads (with the geometric models from tape measurements).

Arch specimen	Collapse load multiplier (λ^{h4})		% Error
	Experimental	Numerical	
1	0.32	0.29	9.38
2	0.32	0.29	9.38
3	0.32	0.28	12.50
4	0.32	0.30	6.25
5	0.31	0.31	0.00
6	0.43	0.41	4.65
7	0.42	0.41	2.38
8	0.42	0.39	7.14
9	0.42	0.39	7.14
10	0.40	0.39	2.50

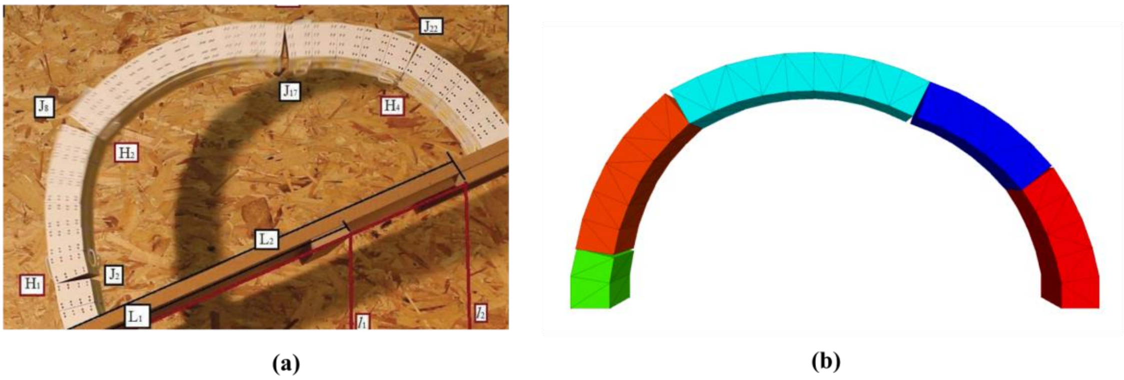


Figure 4-12: Calibration of failure modes of arch specimen number six: (a) experimental; and (b) numerical.

Table 4-13: Material properties of numerical models.

Mechanical property	Symbol	Unit	Model
Density	ρ	Kg/m^3	500
Joint normal stiffness	K_n	GPa/m	0.15
Joint normal stiffness	K_s	GPa/m	K_n /2.38
Joint cohesive strength	C	MPa	0.0
Joint tensile strength	T	MPa	0.0
Joint friction	φ	°	25.0

Structural behaviour indices

For the structural analysis, various metrics were employed for the structural health monitoring of the arch specimens, during loading, according to the proposed structural analysis procedure in Section 3.3.2. To assess the load at which hinge formation occurred, load multipliers were calculated at each hinge formation, symbolised λ^{h1}, λ^{h2}, $\lambda^{h3}$and λ^{h4} respectively (as shown in Figure 4-13a-d). Hinge formation was detected by monitoring the displacement of the joints with the *FISH* script of Section A.1.3. During the loading phase, damage due to loading until the 1st, 2nd and 3rd load multipliers *(*λ^{h1}, λ^{h2}, λ^{h3}) was recoverable (i.e. removal of load results in the initial undamaged state). These load multipliers thus accordingly regard the serviceability limit states of the arch specimens. Moreover, the horizontal stiffness of the arch specimens at the first crack (herein termed stiffness) was calculated by calculating the ratio of the 1st load multiplier (λ^{h1}) toward the horizontal displacement of the block B$_2$ *(*$U_{x,B2}$), as shown in equation (4.1). This also regards the serviceability limit states. Finally, the 4th load multiplier (λ^{h4}) is synonymous with the collapse load and regards the ultimate limit state of the arch specimens.

$$Stiffness = \frac{(\lambda^{h1})}{U_{x,B2}} \tag{4.1}$$

To assess the joint forces of the arch specimens at each hinge formation, the joint forces (maximum normal sub-contact forces) at each joint were calculated, symbolised F_1^n, F_2^n, F_3^n, F_4^n (as shown in Figure 4-13e.). All the above metrics were calculated with automated processes introduced into *3DEC* within the *FISH* script of A.1.3. To calculate the structural behaviour uncertainty, the normalised uncertainty (NU) of (3.74) was employed, where y^{ref} and y are structural behaviour indices of the geometric model from SfM photogrammetry and tape measurements. Also, max (y^{ref}) of (3.74) refers to the maximum value of the property type, in order to normalize the uncertainty (i.e. the collapse load multiplier of arch specimen number 25 for the case of load multipliers).

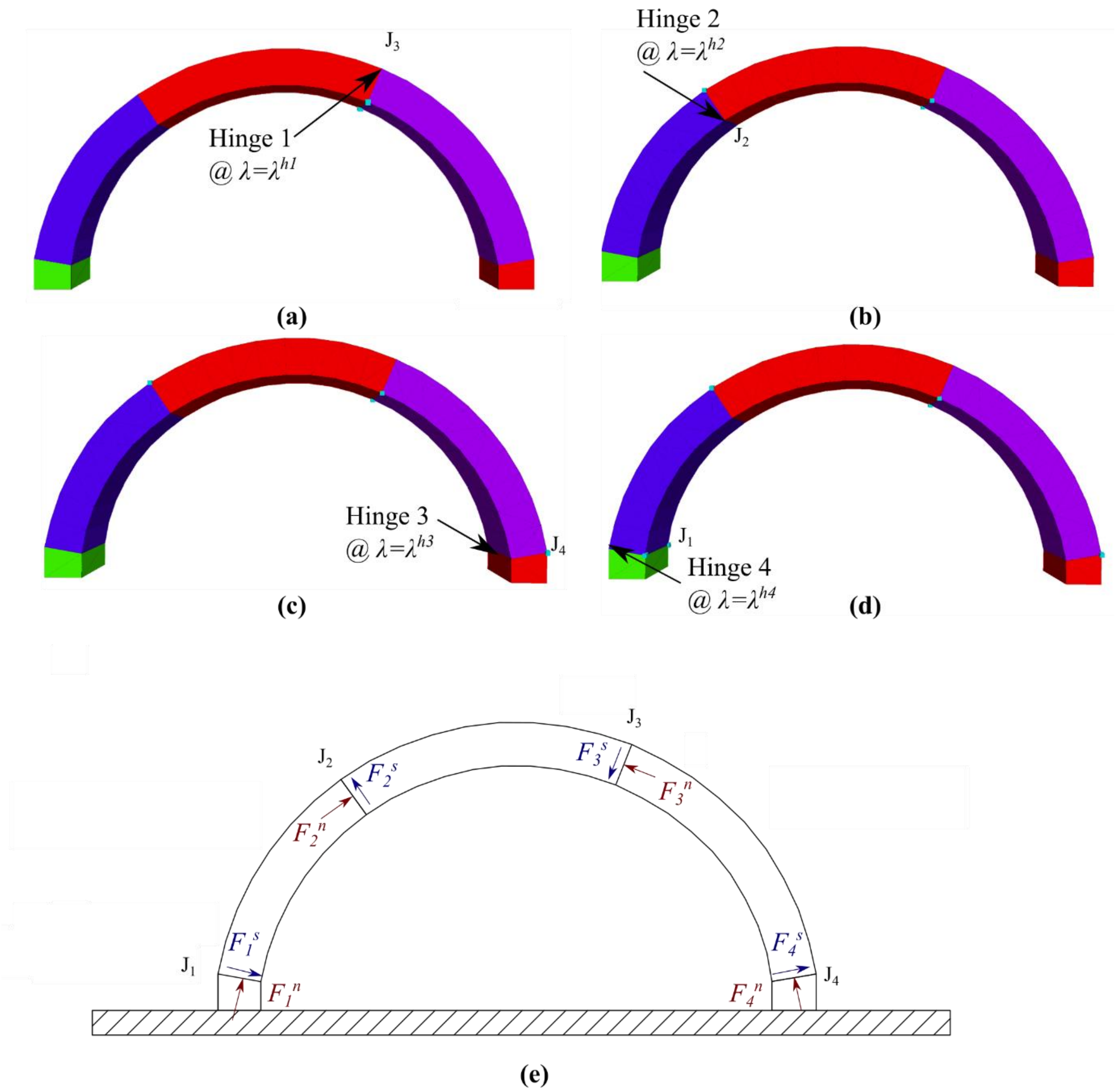


Figure 4-13: Structural behaviour indices. The load multiplier at each hinge formation: (a) 1st load multiplier (λ^{h1}); (b) 2nd load multiplier (λ^{h2}); (c) 3rd load multiplier (λ^{h3}); and (d) 4th load multiplier (λ^{h4}synonymous with collapse load). The joint forces F_1^n, F_2^n, F_3^n, F_4^n at joints J_1, J_2, J_3 and J_4 respectively (e).

4.4.2 Influence of geometrical uncertainty on the load multipliers and stiffness

In Figure 4-14a-e, the magnitude and distribution of all the load multipliers and the stiffness can be seen, of the geometric models from SfM photogrammetry and tape measurements. In Figure 4-14f the normalised uncertainty of all the load multipliers and stiffness is plotted to illustrate the variance between the curves of Figure 4-14a-e (statistical measures found in Table B- 2 of Section B.1). Figure 4-14a-f suggests that the effect of the geometric uncertainty was: a) existent and accumulative for all the load multipliers; b) most significant on the 4th load multiplier (collapse load), i.e. the ultimate limit states, with a minima-maxima interval of (-1 to 10%). Figure 4-14a-f further suggests that the effect of the geometric uncertainty was and even more significant on the stiffness than the load multipliers, with a minima-maxima interval of (2 to 46%). These findings contribute to previous investigation's findings (Morer *et al.*, 2013; Riveiro *et al.*, 2013), that geometric uncertainty significantly affects not only the ultimate limit states (i.e. collapse load λ^{h4}) but also has a significant influence on the serviceability limit states (i.e. λ^{h1}, λ^{h2}, $\lambda^{h3}$and *stiffness*).

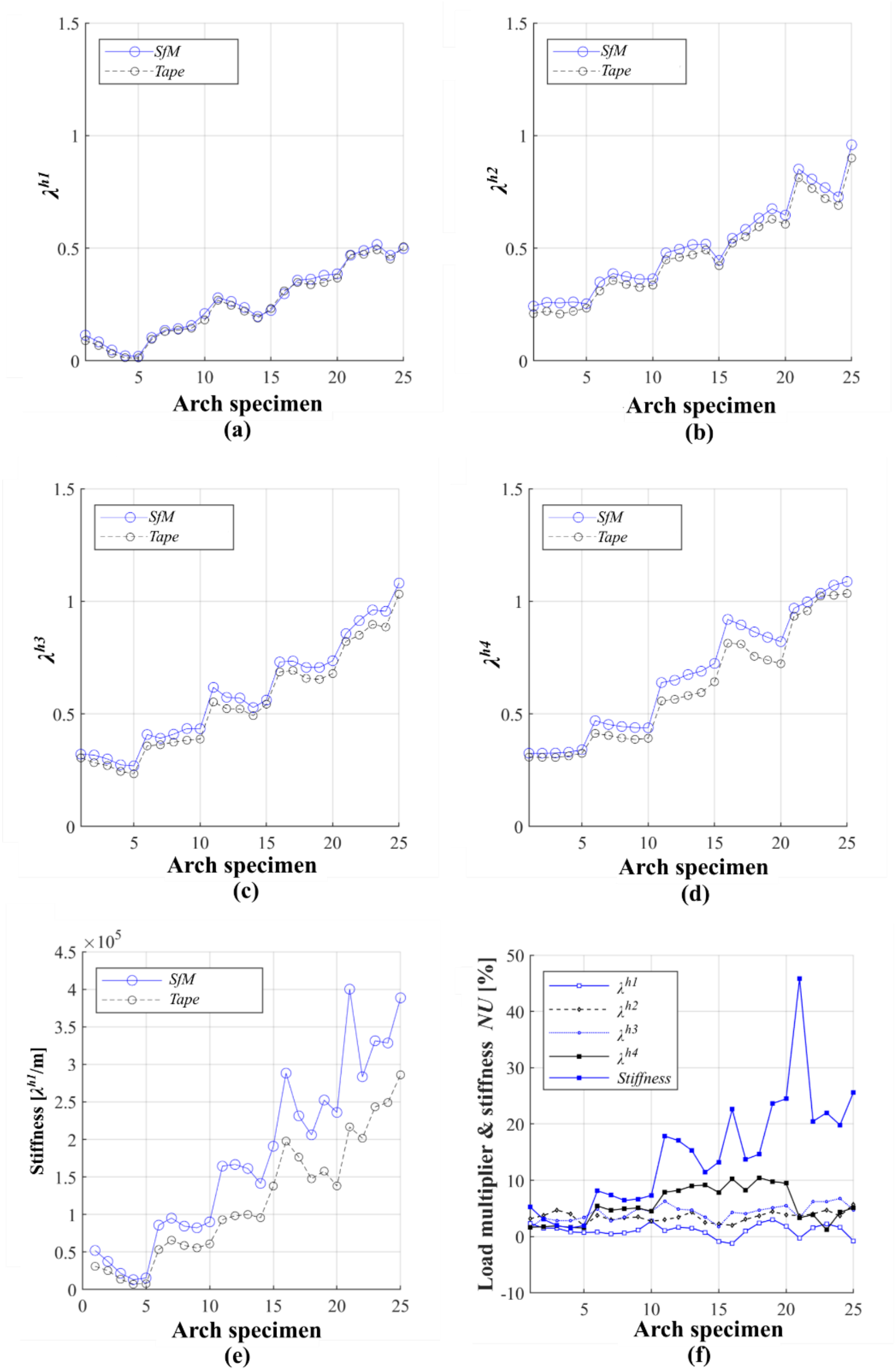


Figure 4-14: Influence of geometric uncertainty on: (a-d) load multipliers and (e) stiffness. Normalised uncertainty of load multipliers and stiffness (f).

4.4.3 Influence of geometrical uncertainty on the joint forces

In Figure 4-15a-d, the magnitude and distribution of the structural behaviour uncertainty for the joint forces can be seen, at each hinge formation, per arch specimen (statistical measures found in Table B- 3 of Section B.1). The curves of Figure 4-15a-d suggest that the effect of geometric uncertainty: a) was significant on the joint forces and disproportionate to the geometric uncertainty, with a minima-maxima interval of normalised uncertainty was calculated at (-15 to 22%) at each hinge formation; and b) remained quasi-constant and present for the all the loading phase. This is another addition to the findings of 4.4.2, that geometrical uncertainty significantly influences the joint forces, for both the serviceability and ultimate limit states.

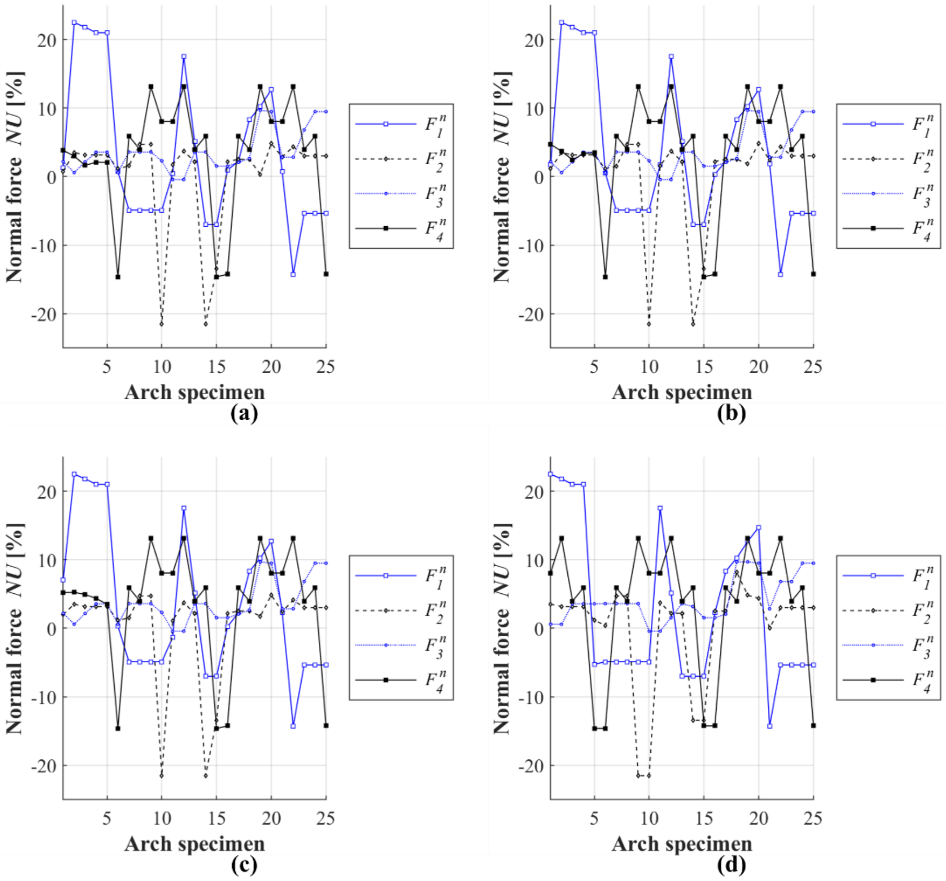


Figure 4-15: Influence of geometric uncertainty on joint forces: (a) first hinge formation ($\lambda = \lambda^{h1}$); (b) second hinge formation ($\lambda = \lambda^{h2}$); (c) third hinge formation ($\lambda = \lambda^{h3}$); and d) fourth hinge formation ($\lambda = \lambda^{h4}$).

4.4.4 Correlation of geometric and structural behaviour uncertainty

Various combinations of the correlation between the geometrical and structural behaviour properties were investigated, as can be found in Table B- 4 and Table B- 5 of Section B.1. From these tables, a correlation was found for the normalised uncertainty of the joint inclination angle of the first joint, (i.e. joint dip J_1) and the normalised uncertainty of the 4th load multiplier λ^{h4} (i.e. collapse load). Indeed, as is also evident from Figure 4-16, a linear trend is visible between the two uncertainties which shows that as the normalised uncertainty of joint dip J_1 decreases from zero, the normalised uncertainty of the 4th load multiplier λ^{h4} increases. Though other causes of structural behaviour uncertainty cannot be ruled out, this suggests that the most probable cause of load multiplier at the formation of the fourth hinge, uncertainty was the joint dip. This is in agreement with a previous investigation (Makris and Alexakis, 2013; Nikolić, 2017; Gáspár *et al.*, 2018) that stereotomy significantly influences the collapse load of arches. This finding suggests that, while developing a geometric model, particular care must be taken to obtain the accurate joint and block (especially joint inclination angle) to ascertain robustness on the predicted collapse load.

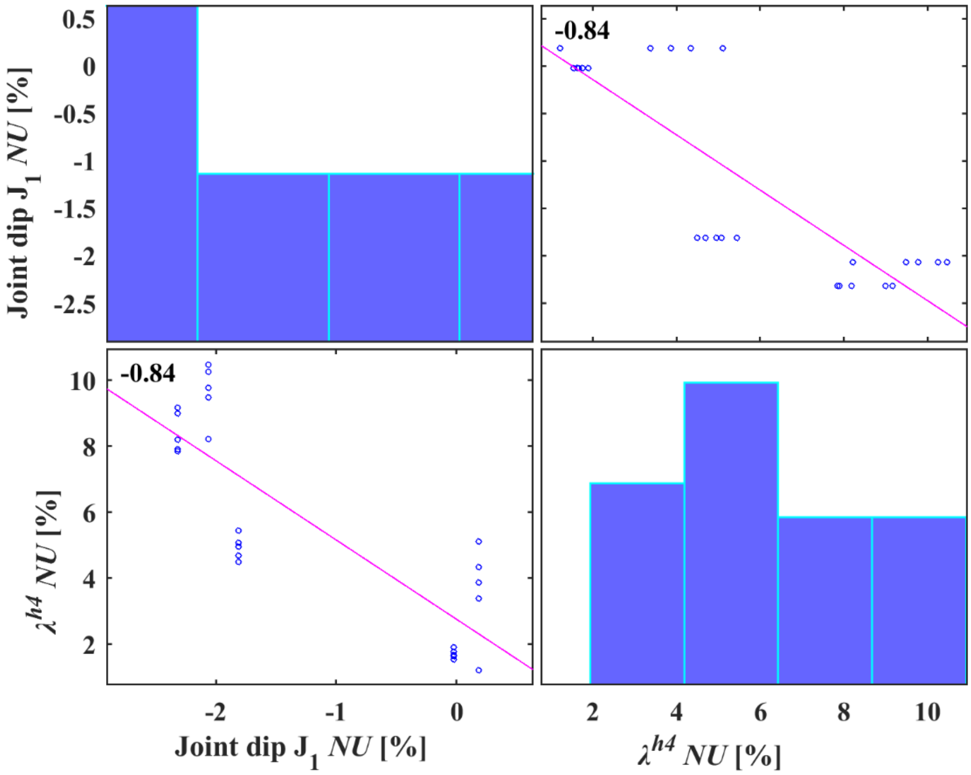


Figure 4-16: Correlation between geometric and structural behaviour uncertainty: Joint dip J_1 - load multiplier λ^{h4} (collapse load) correlation matrix.

4.5 Summary

In this Chapter, the methodological framework was implemented on an experimental structure resembling a regular masonry structure. Geometric models of 25 experimental arch specimens were developed from SfM photogrammetry and compared to those of tape measurements (of structural surveying of a previous investigation). The variables of geometric uncertainty investigated were the: a) joint inclination angle, joint midpoint location and joint length; and b) the block volume and block centroid location After the calculation of geometric uncertainty between the geometric models of SfM photogrammetry and tape measurements, a tilt-plane analysis was numerically simulated with the DEM and the structural behaviour uncertainty between was also calculated. The variables of structural behaviour uncertainty included: stiffnesses (at the first hinge formation), loads multipliers, and normal forces (i.e. between joints).

Due to given differences between the geometric models derived from SfM photogrammetry and tape measurements (-4 to 9%), differences in: a) collapse load (-1 to 10%); b) stiffness (-2 to 46%); and c) normal forces (-15 to 22%) were found. These findings suggest that the employment of accurate geometric models (and consequently geospatial techniques) is important to obtain accurate geometric models and in-turn increase the robustness of the structural analysis. Furthermore, the geometrical uncertainty of the joints, and in specific, the joint inclination angle (i.e. joint dip) was found to directly influence the 4th load multiplier (i.e. collapse load), with a linear trend. This finding suggests that, while developing the geometric model, particular care must be taken to obtain the accurate geometrical properties of the joints to ascertain the robustness of the structural analysis.

Therefore, from the above findings, the importance of employing accurate geometric models to ensure the robustness of the structural analysis of regular masonry structures has been demonstrated. Additionally, through the employment of "*manual Image2DEM*", this study paves the way for the efficient and methodical structural analysis of large-scale regular masonry structures using discrete element modelling.

Chapter 5. A novel approach for the semi-automated numerical modelling of regular masonry "*semi-automated Image2DEM*"

In the previous Chapter, the methodological framework was implemented on an experimental arch structure, demonstrating the importance of accurate geometry for the robust numerical modelling of regular masonry. However, as found in Section 4.5, despite this evident benefit, manual CAD-based block segmentation can be tedious due to its manual nature, and unfeasible for large-scale masonry structures.

This Chapter evaluates the employment of IPTs for the semi-automated and robust discrete element modelling of regular masonry structures. To pursue this aim, both the manual and semi-automated image-based approaches (with manual CAD-based and IPT-based block segmentation) of the methodological framework are implemented according to Figure 5-1. Firstly, the structural surveying of 25 experimentally tested arch specimens is carried out with SfM photogrammetry. Imagery acquisitions obtained with high-quality consumer-camera and the procedures leading up to the orthomosaic development within *Metashape* are detailed. Next, manual CAD-based and IPT-based geometric models of 25 arch specimens are developed from the structural surveying with SfM photogrammetry, employing the manual and semi-automated image-based workflows, respectively. The geometric uncertainty between the geometric models is calculated here, employing the same variables of geometric uncertainty as in Chapter 4. Thereafter, numerical model development and structural analysis are carried out, and the differences in structural behaviour between the manual CAD-based and IPT-based models are calculated, commonly with Chapter 4. Finally, a correlation is also investigated between geometric and structural behaviour uncertainty.

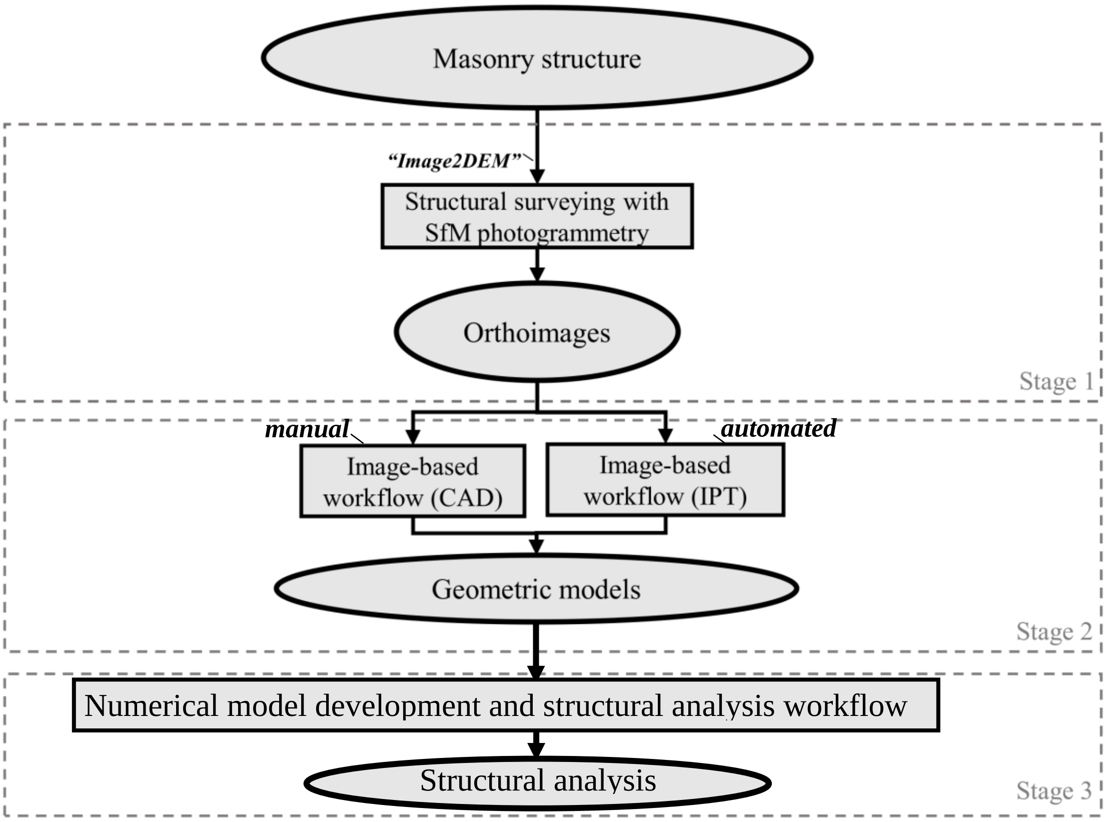

Masonry structure
"Image2DEM"
Structural surveying with SfM photogrammetry
Orthoimages
Stage 1
manual
automated
Image-based workflow (CAD)
Image-based workflow (IPT)
Geometric models
Stage 2
Numerical model development and structural analysis workflow
Structural analysis
Stage 3

Figure 5-1: Implementation of the methodological workflow in Chapter 5. The manual and semi-automated image-based approaches, "manual Image2DEM" and "semi-automated Image2DEM" respectively.

5.1 Stage 1: Structural surveying

The following paragraphs report the results of the structural surveying carried with a digital single-lens reflex (DSLR) Canon EOS6D camera (shown in Figure 5-2) and *Metashape,* according to the flowchart of 3.1.2. The camera shall be herein be termed DSLR camera for abbreviation. It's noteworthy that the reason for employing the DSLR camera was to ensure the obtaining of a better quality orthoimage than that of the smartphone facilitating IPT-based block segmentation. Furthermore, all the times reported are with computational resources of a consumer-grade workstation with an Intel(R) Xeon(R) CPU @ 3.00 GHz processor and 64 GB memory.

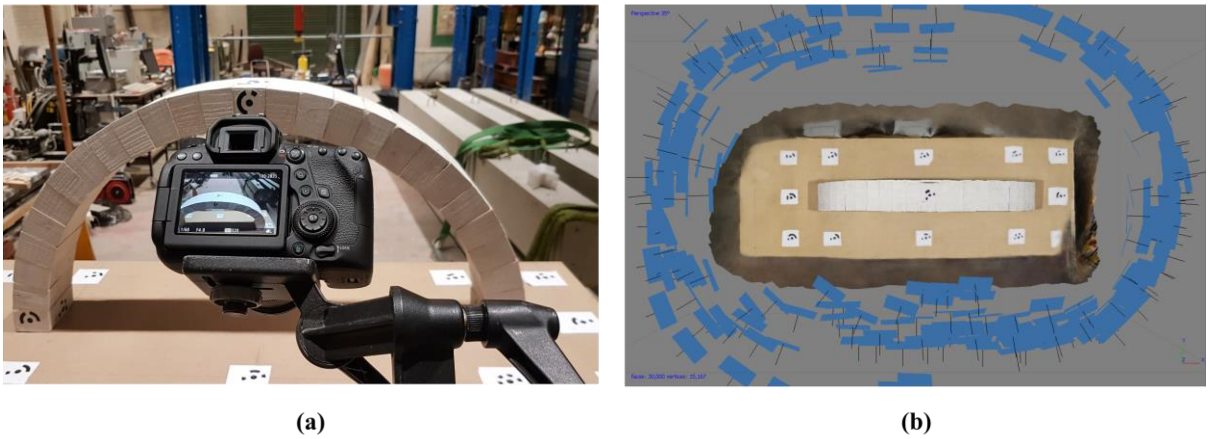
5.1.1 Structural surveying with SfM photogrammetry

Image acquisition

Image acquisition was defined according to the requirements of the GSD_{max} , to comparable to the joint-width of the arch (i.e. sub-mm). To achieve this, a WD equal to 0.5 m from the face of the arch was calculated from Table 3-3 as suitable and The processing properties of the *Metashape* project created are listed in Table 5-1.

Table 5-1: Processing properties of Metashape project using the DSLR camera.

Predefined WD (m)	No of images of the region of interest	Estimated GSD_{max} (mm/pix)	Optical rays per point
0.5	166	0.138	3.0



(a) (b)

Figure 5-2: Structural surveying with SfM photogrammetry: (a) DSLR camera; and (b) image capture network.

The self-calibration properties of the images captured with are found in Table 5-2, where: f is the focal length; C_x and C_y are the principle point's position on the x and y-axis respectively; B_1 is the affinity; B_2 is the non-orthogonality; K_1, K_2 and K_3 are the radial distortion parameters; and P_1 and P_2 are the decentring distortion parameters.

Table 5-2: Self-calibration properties of smartphone sensor (SM-G930F) from Metashape (pixels).

	f	C_x	C_y	B_1	B_2	K_1	K_2	K_3	P_1	P_2
Value	4.2×10^{3}	-11.71	3.01	6.9	-0.99	-0.15	0.12	-0.03	2.1×10^{-3}	2.1×10^{-3}
Error	0.12	0.13	0.15	0.08	0.06	5.3×10^{-5}	1.7×10^{-5}	1.7×10^{-5}	8×10^{-5}	5.7×10^{-5}

Sparse point cloud reconstruction and georeferencing

Sparse point cloud reconstruction was carried out according to Section 3.1.2 and is shown in Figure 5-3a and detailed in Table 5-3. After sparse point cloud reconstruction, the georeferencing followed by introducing the GCPs to *Metashape* in a text format. Concerning georeferencing, the same 12 GCPs surveyed with a Total Station (with their respective errors reported in Table 4-8) were employed. The properties of the georeferencing with *Metashape* are reported in Table 5-4.

Table 5-3: Sparse point cloud reconstruction properties from Metashape with DSLR camera.

Parameter	
Points	142,091
RMS reprojection error	0.16 (0.67 pix)
Mean key point size	3.86 pix
Average tie point multiplicity	2.97
Accuracy	High
Generic preselection	Yes
Key point limit	400,000
Tie point limit	10,000
Adaptive camera model fitting	Yes
Matching time	19 minutes 49 seconds
Alignment time	1 minute 29 seconds
Optimisation parameters	f, C_x, C_y, B_1
Optimisation time	6 seconds

Table 5-4: Georeferencing errors of targets from Metashape with DSLR camera.

	Target ID	X error (mm)	Y error (mm)	Z error (mm)	Total error (mm)	Image error (pix)
Control points	11	1.04	1.49	-0.89	2.02	0.28
	12	-0.30	-0.71	-1.22	1.44	0.14
	13	-2.34	-1.72	1.11	3.11	0.19
	14	3.27	3.38	0.99	4.80	0.09
	18	-0.55	0.04	0.03	0.55	0.13
	19	2.30	-0.40	-0.29	2.36	0.04
	20	0.58	1.27	0.05	1.39	0.15
	23	-4.00	-3.33	0.22	5.21	0.32
	24	1.04	1.49	-0.89	2.02	0.28
	Total	2.22	1.94	0.76	3.04	0.18
Check points	2	1.78	1.85	-2.67	3.70	0.28
	3	-0.31	0.61	-1.15	1.34	0.23
	4	0.86	0.63	-3.20	3.38	0.35
	5	-0.95	0.06	-0.49	1.07	0.40
	Total	1.11	1.02	2.18	2.65	0.31

Dense point reconstruction

The reconstructed dense point cloud is shown in Figure 5-3a and detailed in Table 5-5. From the options of smoothness and detail within *Metashape*, the high-quality point cloud reconstruction setting was used in conjunction with the aggressive smoothness level.

Table 5-5: Dense point cloud reconstruction properties: DSLR camera.

Parameter	
Points	10,199,073
Point cloud reconstruction time	High
Filtering mode	Aggressive
Processing time of depth maps generation parameters	2 hours 58 minutes
Processing time of dense cloud generation parameters	32 minutes 48 seconds

Orthomosaic development

The 3D model of the arch is shown in Figure 5-3c while the orthomosaic is shown in Figure 5-3d which was based on a mesh of 30,000 faces. Concerning the orthomosaic, its resolution was equal to 0.14 mm whilst the definability of the discontinuities was controlled in *Metashape*. The properties of the mesh and orthomosaic development are reported in Table 5-6.

Table 5-6: Mesh and orthomosaic properties with the DSLR camera.

	Parameter	
Mesh	Faces	30,000
	Vertices	15,105
	Surface type	Arbitrary
	Source data	Sparse point cloud
	Interpolation	Enabled
	Strict volumetric masks	No
	Processing time	3 seconds
Orthomosaic	Size	9,981 x 5,099
	Coordinate system	Local Coordinates (m)
	Blending mode	Mosaic
	Surface type	Mesh
	Processing time	3 minutes 5 seconds

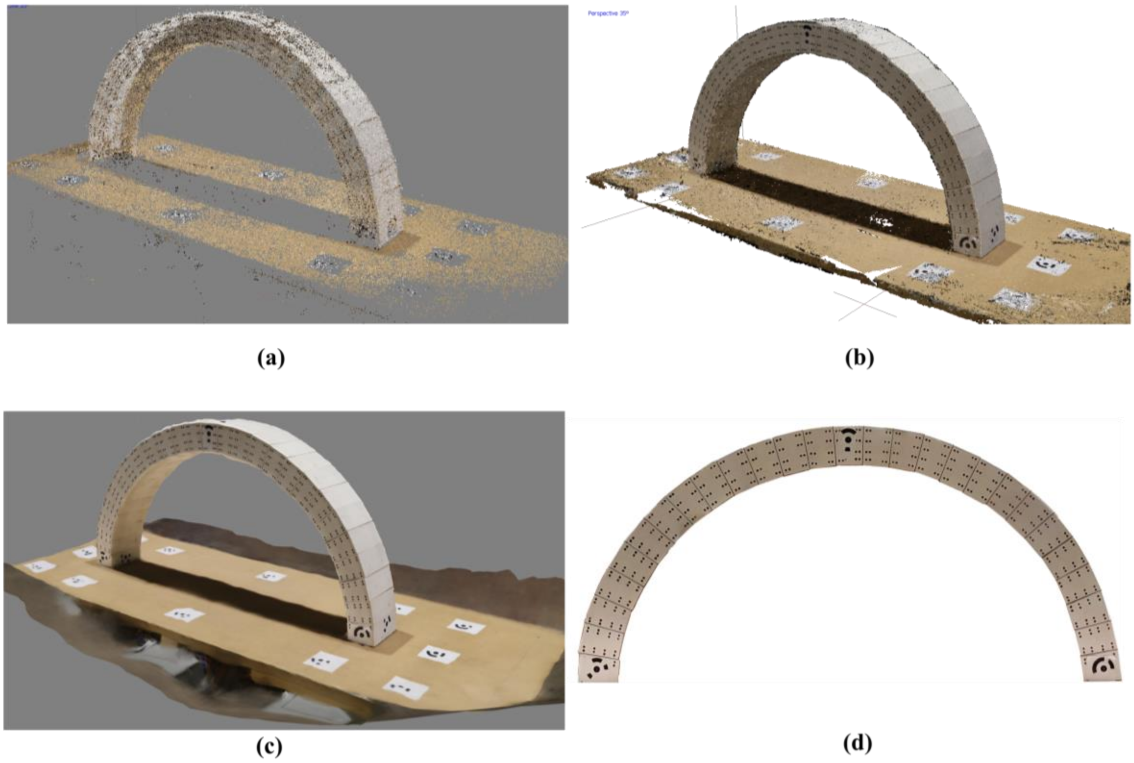


Figure 5-3: Structural surveying with SfM photogrammetry and DSLR camera: (a) sparse point cloud; (b) dense point cloud; (c) 3D model; and (d) orthomosaic.

5.2 Stage 2: Geometric model development

The following paragraphs report the results of the implementation of the manual and semi-automated image-based workflows respectively, resulting in manual CAD-based and IPT-based geometric models, respectively. After geometric model development, the alignment of the geometric models was also carried out, as in Chapter 4 (with a common reference point and common axis convention, as shown in Figure 5-5b-d). Then, the geometric uncertainty between the geometric models of manual CAD-based and IPT-based block segmentation was calculated. Concerning the geometric uncertainties, the same geometrical properties of Chapter 4 were subject to geometrical assessment according to the workflow of Section 3.3.2 which were: the invariable geometrical properties (arch specimens’ span and rises); and the variable geometrical properties calculated (the joint length, joint midpoint location and joint dip (joint inclination angle), the block volume and block centroid location). To calculate the geometric uncertainty, the normalised uncertainty (NU) of (3.11) was used, where $x^{ref}$and x are geometrical properties of the geometric model from the manual CAD-based and IPT-based geometric models respectively. Additionally, max (x^{ref}) of (3.11) refers to the maximum value of the property group (i.e. the arch span for the normalized uncertainty of the block centroid x-axis properties). It is additionally important to note that as in the previous Chapter, only the block and joint vertices of the initial arch (i.e. the arch made of free moving blocks of Figure 4-2a) were measured. The properties each of the arch specimens (i.e. block volume, block centroid, joint dip, joint length and joint midpoint) were subsequently calculated as a function of the measured vertices.

5.2.1 Geometric models of manual CAD-based block segmentation

Geometric models of the manual CAD-based block segmentation were developed following the image-based workflow of 3.2.2 commonly with Chapter 4, as shown in Figure 5-5a. Again as with the geometric models of Chapter 4, firstly a geometric model of the initial arch assembly was developed as shown in Figure 5-5b, then each arch specimens were developed with the same *FISH* script of Section A.1.4.

5.2.2 Geometric models of IPT-based block segmentation

Geometric model development

To carry out the IPT-based block segmentation, the two-phases procedure of Section 3.2.2 was carried out. The results of the joint image development are shown in Figure 5-4a-g, which consist of: a) pre-processing of the greyscale orthomosaic (*roifill* function) to remove noise; b) edge detection of the pre-processed image (*edge* function) to highlight block edges; c) the mask creation (*imdil* function), to define the borders of the area occupied by the masonry structure; d) joint line detection with a Hough transformation (*hough* function), to make joints continuous; e) joint lines fusing with the mask (*imfuse* function); and f) joint line homogenisation and joint image development with watershed segmentation (*watershed* function). These are termed the permissible borders are shown in Figure 5-4h.

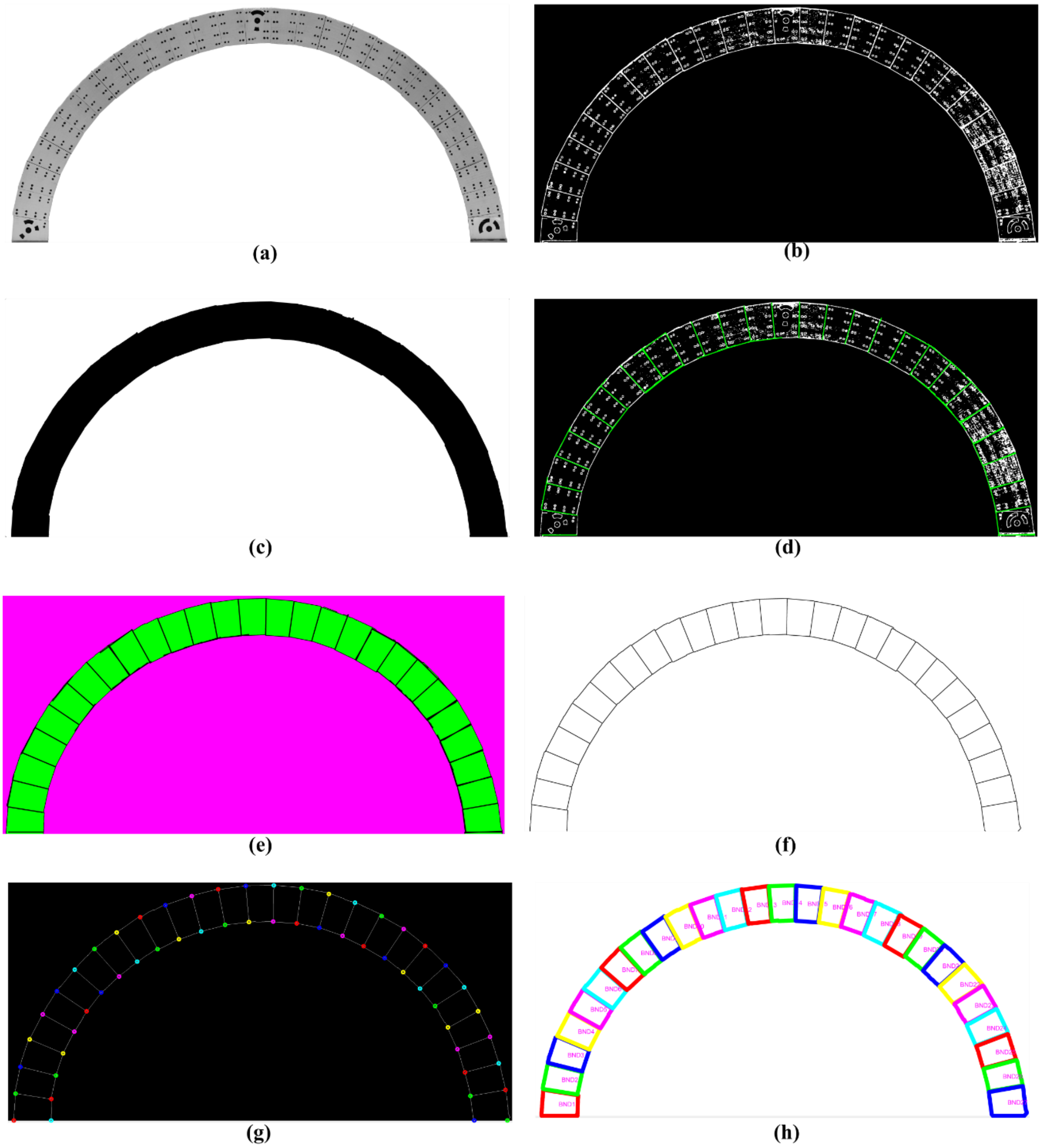


Figure 5-4: Block segmentation with image processing. Joint image development: (a) pre-processing of the greyscale image; (b) edge detection; (c) the mask creation; (d) joint line detection; (e) joint lines and block border; and f) joint line homogenisation and joint image development with watershed segmentation (watershed function). Block vertex extraction: (g) desired block vertices; and permissible block vertices.

The results of the second phase of the IPT-based block segmentation which are block vertices, as shown in Figure 5-5c. Thereafter, geometric models of the IPT-based block segmentation were developed in the same fashion as Chapter 4, as shown in Figure 5-5d.

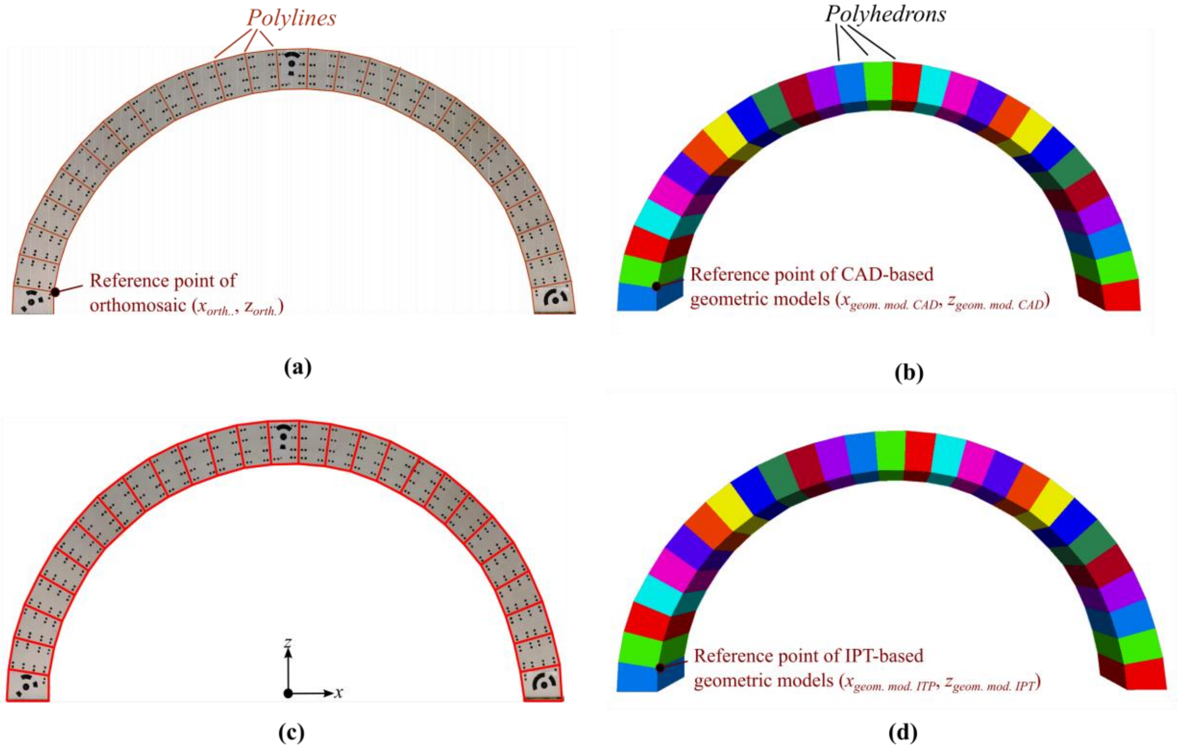

Polylines
Polyhedrons
Reference point of orthomosaic ($x_{orth.}$, $z_{orth.}$)
Reference point of CAD-based geometric models ($x_{geom.\ mod.\ CAD}$, $z_{geom.\ mod.\ CAD}$)
(a)
(b)
z
x
Reference point of IPT-based geometric models ($x_{geom.\ mod.\ ITP}$, $z_{geom.\ mod.\ IPT}$)
(c)
(d)

Figure 5-5: Manual image-based workflow: (a) manual CAD-based block segmentation; and (b) geometric model. Semi-automated image-based workflow: (c) IPT-based block segmentation; and (d) geometric model.

Geometric assessment

Concerning the geometric assessment of the IPT-based geometric models, the same procedure was followed, as in Chapter 4. The span and rise normalised uncertainties were equal to 4.61 and 2.46% respectively, for all the arch specimens. Again, only the geometric uncertainty of the blocks and joints is the only concern in this investigation. In Figure 5-6a-d and Figure 5-7a-d, the magnitude and distribution of the normalised geometric uncertainty is plotted for all the joint and block parameters, per arch specimen (statistical measures found in Table B- 6 of Section B.2). Figure 5-6a-d and Figure 5-7a-d, suggests that the quantity of the geometric uncertainty was: a) significant for all the joints and block properties; and b) most significant for the block volume with minima-maxima interval of (-5 to 10%). The cause of geometric uncertainty is attributed to the IPT block segmentation. Finally, with regards to the consistency of joints errors of Figure 5-6 a-d, this is again attributed the fact that during the experimental testing, only the block and joint vertices of the initial arch assembly were measured, as aforementioned in the previous Chapter (and not the 25 arch specimens *per se*). Since each arch specimen was measured as a four-block assembly using the blocks and joints of the initial arch assembly, multiple arch specimens shared common joints, leading to consistency in some of the joint errors.

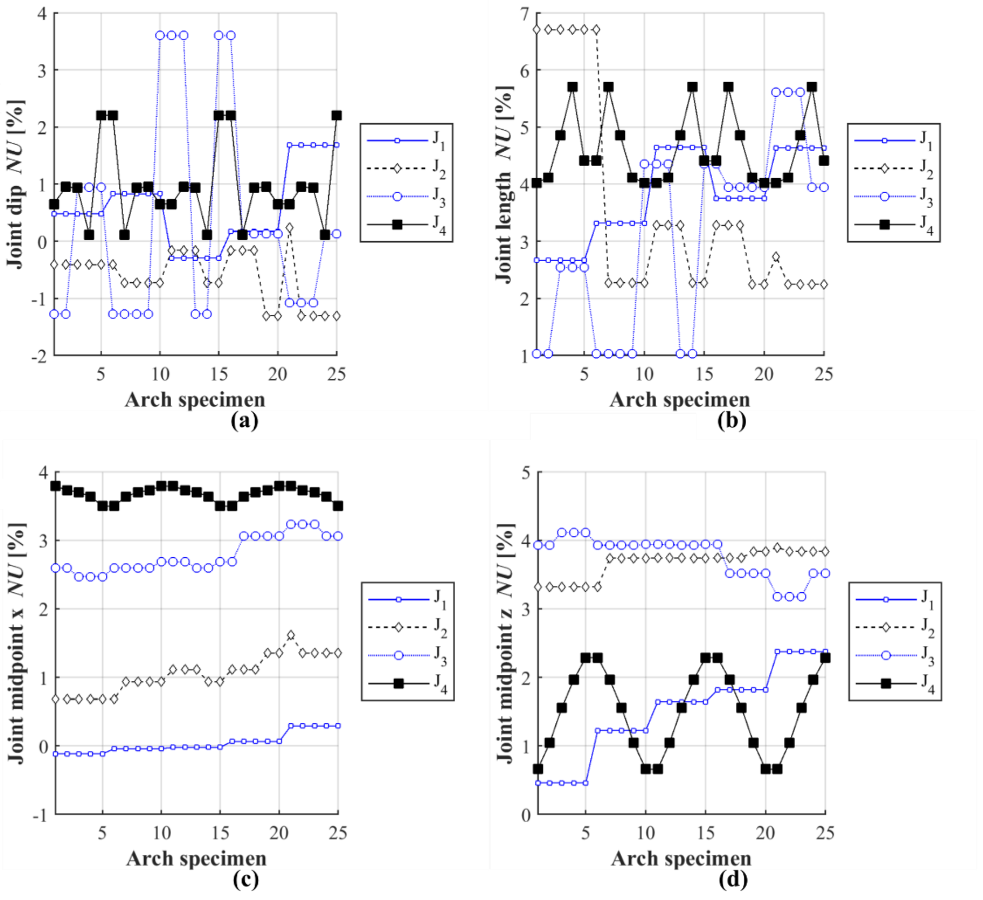


Figure 5-6: Normalised geometric uncertainty of: (a) joint dip; (b) joint length; (c) joint midpoint x; and (d) joint midpoint z.

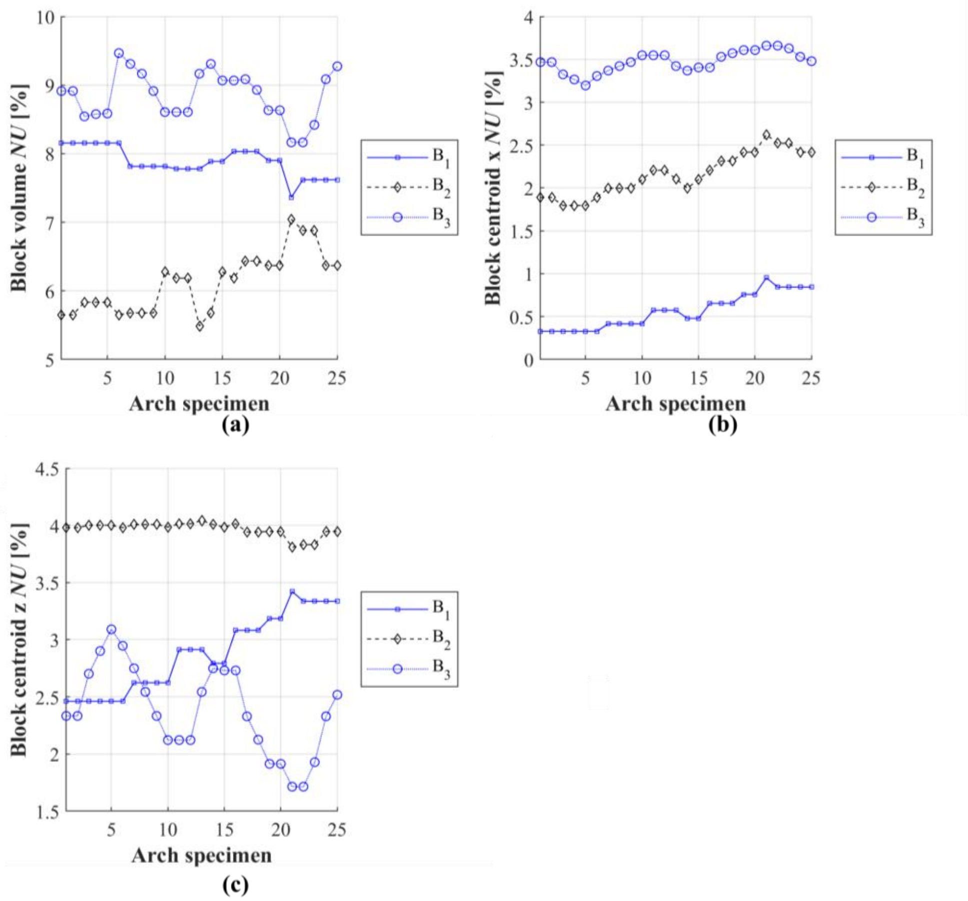


Figure 5-7: Normalised geometric uncertainty of: (a) block volume; (b) block centroid x; and (c) block centroid z.

5.3 Stage 3: Structural analysis with the discrete element method

The following paragraphs detail the numerical model development with *3DEC* according to Section 3.3.2. Then the subsequent structural analysis follows employing geometric models developed from the manual CAD-based and IPT-based block segmentation.

5.3.1 Numerical model development and structural analysis with 3DEC

The numerical models were developed with the same material properties, boundary conditions and loading protocol of Chapter 4. Furthermore, the same structural behaviour indices of Chapter 4 were employed during the structural analysis. The following paragraphs report the results of the structural behaviour uncertainty due to geometric uncertainty between the IPT-based and manual CAD-based geometric models.

5.3.2 Influence of geometric uncertainty on the load multipliers and stiffness

In Figure 5-8a-e, the magnitude and distribution of all the load multipliers and the stiffness can be seen, of the geometric models from IPT and manual CAD-based framework (statistical measures found in Table B- 7 of Section B.2). In Figure 5-8f the normalised uncertainty of all the load multipliers and stiffness is plotted to illustrate the variance between the curves of Figure 4-14a-e. Figure 5-8a-f suggests that the effect of the geometric uncertainty was: a) existent and accumulative for all the load multipliers; b) most significant on the 4th load multiplier (collapse load), i.e. the ultimate limit states, with a minima-maxima interval of (-7.5 to 2%). Figure 5-8a-f further suggests that the effect of the geometric uncertainty was and even more significant on the stiffness than the load multipliers, with a minima-maxima interval of (-2.5 to 25%). These findings contribute to both the previous Chapter's and other investigation's findings (Morer *et al.*, 2013; Riveiro *et al.*, 2013), that geometric uncertainty significantly affects not only the ultimate limit states (i.e. collapse load λ^{h4}) but the serviceability limit states (i.e. λ^{h1}, λ^{h2}, $\lambda^{h3}$and *stiffness*).

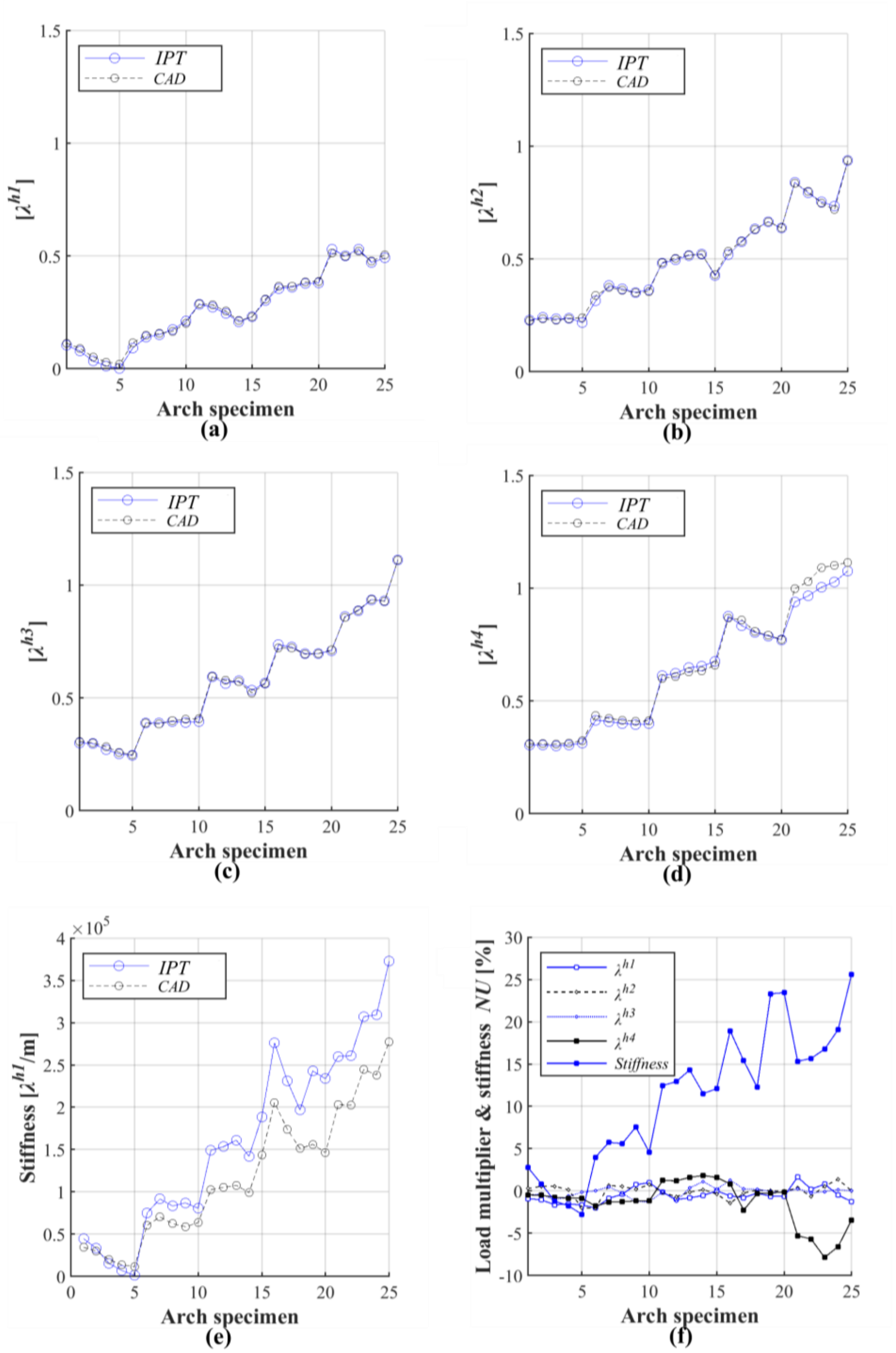


Figure 5-8: Influence of geometric uncertainty on: (a-d) load multipliers; and (e) stiffness. Normalised uncertainty of load multipliers and stiffness (f).

5.3.3 Influence of geometric uncertainty on the joint forces

In Figure 5-9a-d, the magnitude and distribution of the structural behaviour uncertainty for the joint forces can be seen, at each hinge formation, per arch specimen (statistical measures found in Table B- 3 of Section B.2). The curves of Figure 5-9a-d suggest that the effect of geometric uncertainty: a) was significant on the joint forces and disproportionate to the geometric uncertainty, with a minima-maxima interval of normalised uncertainty was calculated at (-10 to 70%) at each hinge formation; and b) remained quasi-constant and present for the all the loading phase. This is another addition to the findings of Chapter 4, that even seemingly minuscule geometrical uncertainty significantly influences the joint forces, for both the serviceability and ultimate limit states.

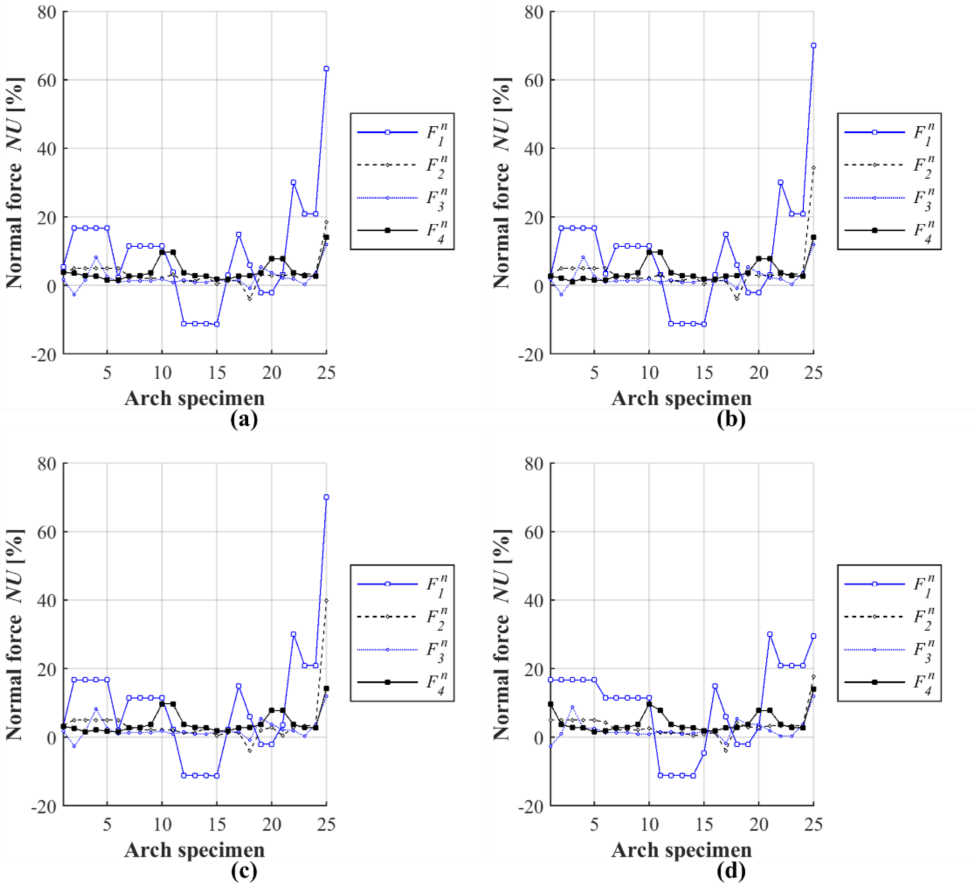


Figure 5-9: Influence of geometric uncertainty on joint forces: (a) first hinge formation ($\lambda = \lambda^{h1}$); (b) second hinge formation ($\lambda = \lambda^{h2}$); (c) third hinge formation ($\lambda = \lambda^{h3}$); and (d) fourth hinge formation ($\lambda = \lambda^{h4}$)

5.3.4 Correlation of geometric and structural behaviour uncertainty

Various combinations of the correlation between the geometrical and structural behaviour properties were investigated, as can be found in Table B- 9 and Table B- 10. of Section B.2. As in the previous Chapter, a correlation (even stronger than the previous Chapter's correlation) was found between the normalised uncertainty of the joint inclination angle of the first joint, (i.e. joint dip J_1) and the normalised uncertainty of the 4th load multiplier λ^{h4} (i.e. collapse load). Indeed as also evident from Figure 5-10, a linear trend is visible between the two uncertainties which shows that as the normalised uncertainty of joint dip J_1 decreases from zero, the normalised uncertainty of the 4th load multiplier λ^{h4} increases. Though other causes of structural behaviour uncertainty cannot be ruled out, this suggests that the most probable cause of load multiplier at the formation of the fourth hinge, uncertainty was the joint dip. This reinforces the findings of the previous Chapter, that while developing the geometric model, particular care must be taken to obtain the accurate joint and block (especially joint inclination angle) to ascertain robustness on the predicted collapse load.

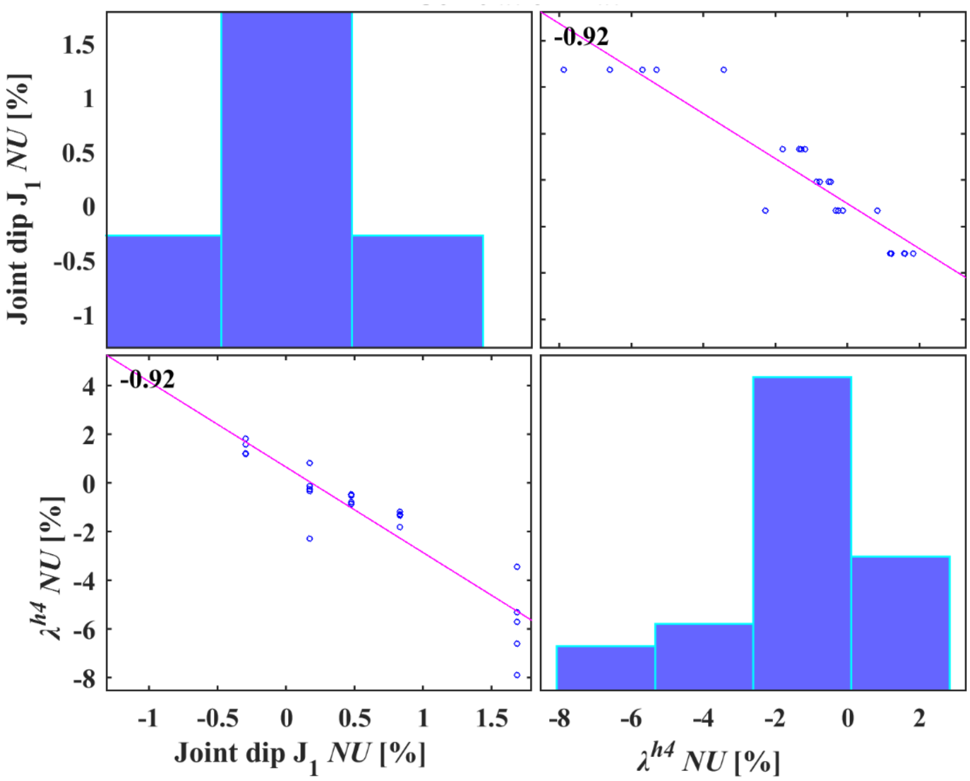


Figure 5-10: Correlation between geometric and structural behaviour uncertainty: Joint dip J_1 - load multiplier λ^{h4} (collapse load) correlation matrix.

5.4 Summary

In this Chapter, the manual and semi-automated image-based approaches of the methodological framework were implemented on an experimental structure, resembling a regular masonry structure. After structural surveying with from SfM photogrammetry, geometric models of 25 experimental arch specimens were developed from both IPT- and manual CAD-based block segmentation. A tilt-plane analysis was numerically simulated with the DEM demonstrated that the obtained stiffness (at the first hinge formation) and load multipliers (at each hinge formation) of the IPT-based geometric models was comparable the manual CAD-based.

Specifically, the IPT-based and manual CAD-based geometric models showed a good agreement in terms of geometry, differences of (-5 to 10%). Moreover, concerning the structural behaviour, the load multipliers of the geometric models also showed a good agreement, with differences of up to 7%. Stiffnesses, however, showed partial agreement, with differences of up to 7% for 10 specimens and 24% for 15 specimens. These findings primarily demonstrate the potential of the employing IPTs to accurately capture both the geometry and structural behaviour of masonry structures, especially regarding the ultimate limit states (i.e. collapse load λ^{h4}). However, these findings also suggest that the effect of geometric uncertainty can be seemingly insignificant upon the ultimate limit states (i.e. collapse load λ^{h4}) yet notable on the serviceability limit states (i.e. *stiffness*, in this case). Therefore, differences in serviceability limit states should be accounted for, even for seemingly accurate geometric models. Finally, the geometrical uncertainty of the joints, and in specific, the joint inclination angle (i.e. joint dip) was again found to directly influence the 4th load multiplier (i.e. collapse load), with a linear trend. This reinforces the findings of the previous Chapter that whilst developing a geometric model, particular care must be taken to obtain reliable geometrical properties of the joints to ascertain the robustness of the collapse load.

Therefore, from the above findings, the potential of the methodological framework to perform semi-automated and robust discrete element modelling of regular masonry structures has been demonstrated. Given that the IPT-based framework was only applied on a small-scale experimental structure, a pilot study is herein demonstrated on a full-scale multiring masonry arch, carried out by the Author to demonstrate the future potential of the approach.

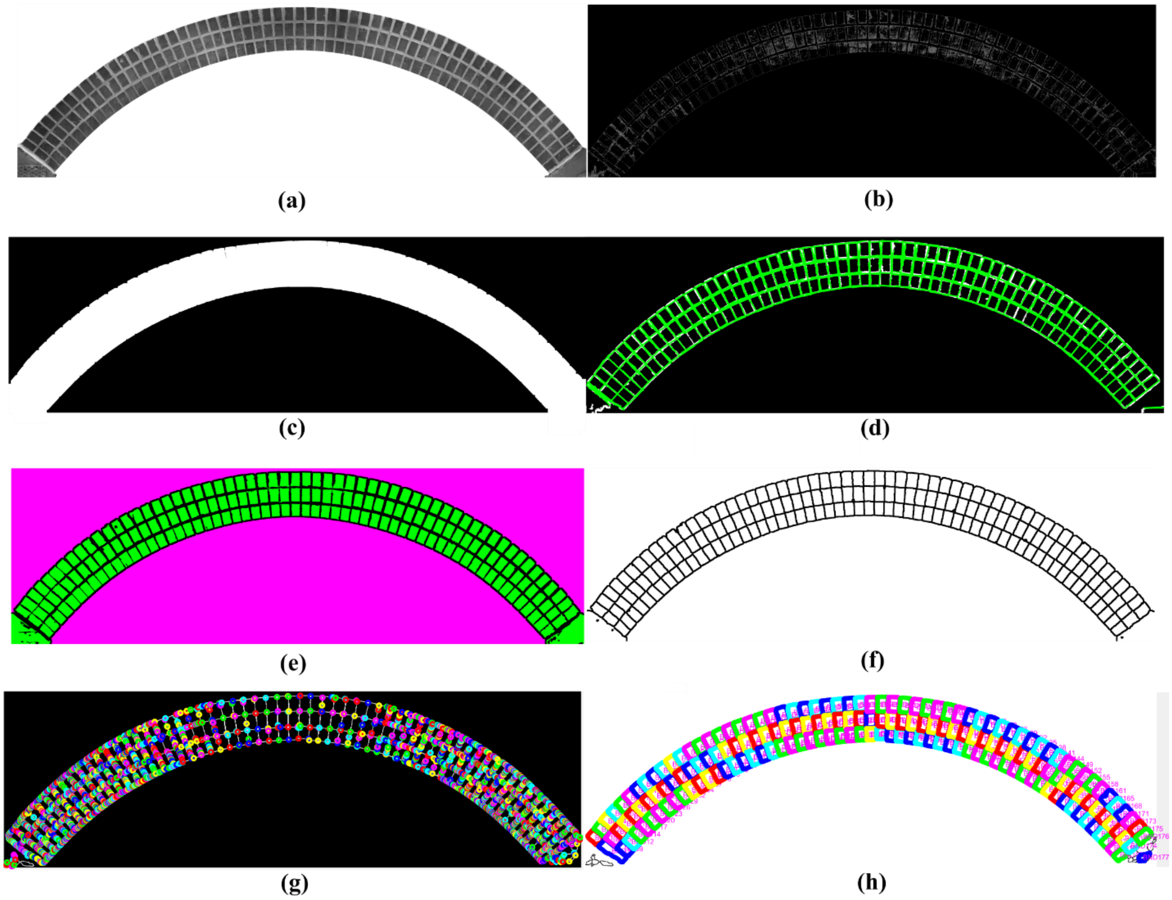

(a) (b) (c) (d) (e) (f) (g) (h)

Figure 5-11: Block segmentation with image processing on a real-world structure. Joint image development: (a) pre-processing of the greyscale image; (b) edge detection; (c) the mask creation; (d) joint line detection; (e) joint lines and block border; and f) joint line homogenisation and joint image development with watershed segmentation (watershed function). Block vertex extraction: (g) desired block vertices; and permissible block vertices.

From this Chapter's findings and the pilot study of Figure 5-11, it is evident that the employment of "*semi-automated Image2DEM*" paves the way for the automated structural analysis of real-world regular masonry structures using discrete element modelling.

Chapter 6. A novel approach for the semi-automated numerical modelling of rubble masonry "*Cloud2DEM*"[8]

In the previous Chapters, the image-based and point-based approaches were employed which regarded the numerical modelling of regular masonry structures. However, as discussed in Section 2.4, for the case of rubble masonry, where the blocks are not distinguishable and the geometry of the structure is complex, on cloud-based approaches are advantageous.

Until the present moment, the most common assessment methods of rubble masonry have been the continuum FEM models (i.e. smeared crack models) and FELA, which as demonstrated, are oversimplified. Of the few studies that exist for rubble within the DEM until now have been carried out with a simplified geometry. For instance, de Felice (2011) developed pioneering 2D DEM models of rubble masonry walls. Varying morphologies were developed according to a previous investigation characterizing rubble masonry, to evaluate their out-of-plane structural capacity. It was shown that the morphology of the stones in the wall significantly influenced the stiffness, load-bearing and deformation capacity of the walls. In a further study, 2D models base were developed (Pulatsu *et al.*, 2016) with the DEM were used to investigate the out-of-plane capacity of rubble masonry walls with different cross-sections The investigation found a notable drop in ultimate load-bearing capacity of rubble masonry walls due to internal cavities and irregularly shaped blocks, as compared to panels of regular-shaped blocks. Shrive (2015) developed 2D models to study the structural stability of a historical rubble fortification. Geometric models of the rubble masonry were comprised of circular and polygonal elements randomly assembled. From studies such as the above, the DEM's potential

[8] The study in this Chapter has been published in the following research papers:

Kassotakis, N., Sarhosis, V., Mills, J., D'Altri, A., Miranda, S. and Castellazzi, G. (2018). From point clouds to geometry generation for the detailed micro-modelling of masonry structures. Conference: 10th IMC - International Masonry Conference, Politecnico di Milano - International Masonry Society - IMS. Milan, Italy. pp. 1-14

Kassotakis, N., Sarhosis, V., Riveiro, B., Conde, B., Mills, J., D'Altri, A., Miranda, S. and Castellazzi, G. (2020c). Three-dimensional discrete element modelling of rubble masonry structures from dense point clouds. Automation in Construction, Doi: 10.1016/j.autcon.2020.103365 is unequivocally demonstrated for capturing the complex structural behaviour of rubble masonry. However, since very detailed metrical geometric information is essential, the DEM is not commonly employed.

This Chapter evaluates the potential of employing voxelization for the semi-automated and geometrically-accurate structural analysis of rubble masonry. To pursue this aim, the cloud-based approach of the methodological framework is implemented on a full-scale rubble masonry structure, according to Figure 6-1. The Chapter commences with the description of the selected case study and then proceeds to implement the methodological framework in a stage-wise fashion. Firstly, the structural surveying with a TLS survey of the structure is detailed from a previous study (Prizeman *et al.*, 2017). Next, geometric models are developed of a variable block size and orientation. Finally, numerical model development and structural analysis are carried out. Concerning the structural analysis, for the course voxel model of a 50 cm voxel size, an initiatory study was carried out to investigate the effect of: a) joint strength properties; b) the geometric uncertainty is investigated on the structural behaviour. Also, the effect of voxel size was investigated on the structural behaviour, numerical model properties and simulation times. Furthermore, for the 30 cm voxel models, the effect of voxel orientation is also investigated on the structural capacity.

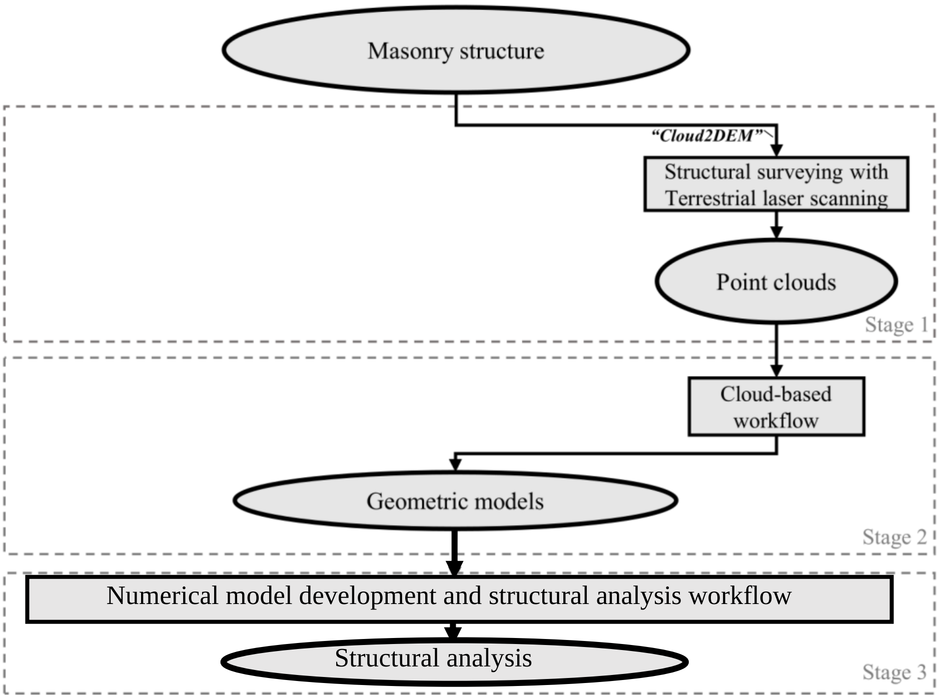


Figure 6-1: Implementation of the methodological framework in Chapter 6. The cloud-based approach "Cloud2DEM".

6.1 Case study two: The southwest leaning tower of Caerphilly Castle[9]

The case study which was used to evaluate the proposed framework is the leaning tower of the Caerphilly Castle, located in South Wales, the UK. Constructed in the 13th Century, Caerphilly is the second largest castle in the UK and one of the largest in Europe (Renn, 2002). The southwest tower shown in Figure 6-3. is 17 m tall and 9 m in diameter (Renn, 2002). It is reported to have been leaning for several centuries and stands at a current angle of approximately 10 degrees off vertical. The tower was constructed of rubble masonry, with a rough texture and indefinable joints. The most probable cause of leaning of the Caerphilly tower is attributed to the lack of foundation strength and stiffness which was induced by dewatering in the 18th Century.

[9] The structural surveying data herein presented was captured by a previous investigation of Prizeman *et al.* (2017)

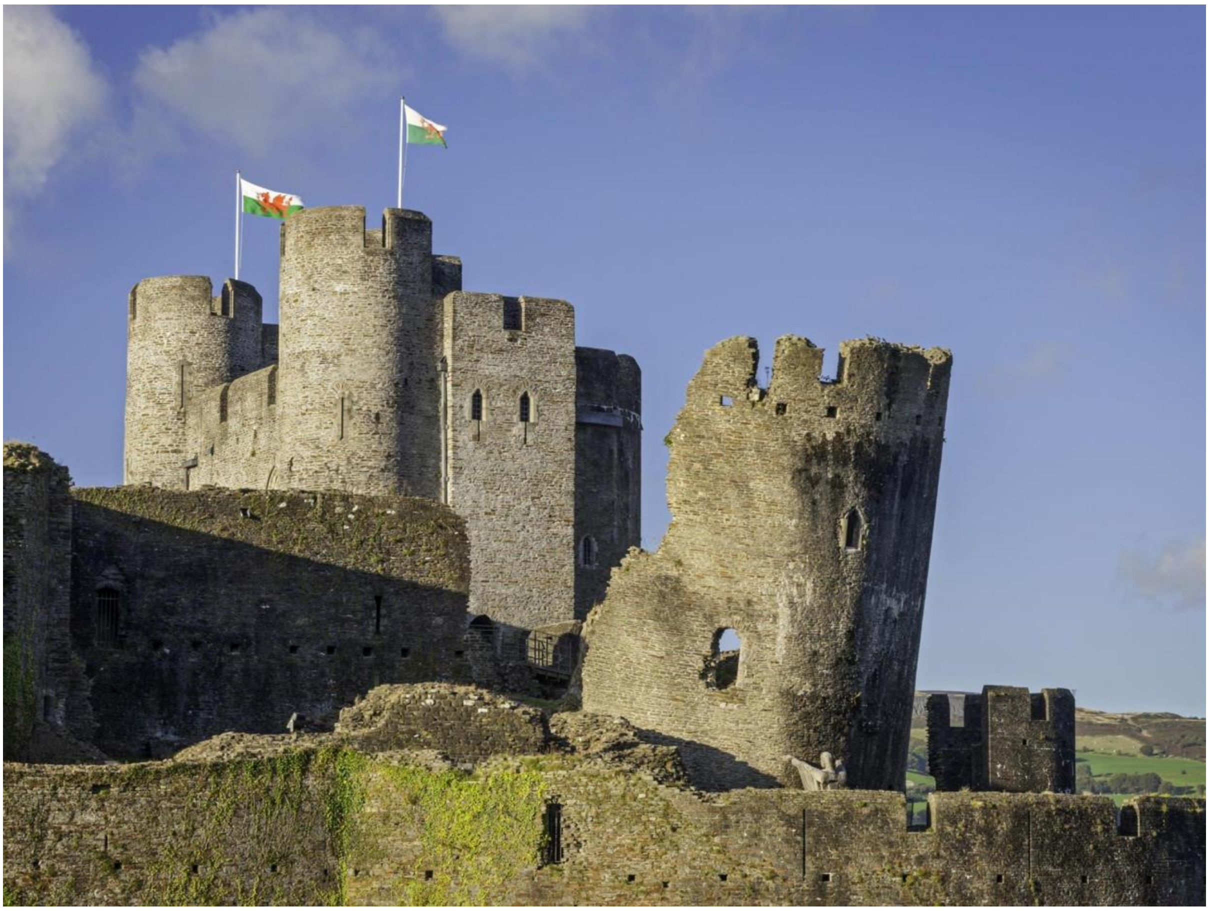


Figure 6-2: Caerphilly Castle (Renn, 2002). View of the face of the southeast leaning tower.

Over the last two decades, the structural analysis of leaning towers such as Caerphilly has attracted the attention of many researchers (Hambly, 1985; Abruzzese *et al.*, 2009; Marchi *et al.*, 2011; Milani *et al.*, 2017). For instance, early analytical approaches to investigate the safety of leaning towers (Heyman, 1992), have provided information on the critical inclination angle of leaning towers, however, rely upon oversimplified material assumptions (rigid masonry without tensile strength and regular geometries). Such approaches cannot be applied to the present case study since the tower is highly irregular in shape, with openings, voids, and a non-rectangular base. Recently, FEM and LA have been successfully applied to perform structural analysis on leaning towers with complex geometry (Castellazzi *et al.*, 2017; Prizeman *et al.*, 2017; D'Altri *et al.*, 2018b). However, as found in Section 2.2, the FEM cannot always accurately describe the discontinuous nature of rubble masonry (Dejong, 2009), while the LA can, but still relies upon simplified assumptions and is not able to provide information about the in-service condition of the structure under consideration.

6.2 Stage 1: Structural surveying

The following paragraphs details the structural surveying carried out with a terrestrial laser scanner as part of a previous investigation (Prizeman *et al.*, 2017), with a similar workflow to that of 3.1.3.

6.2.1 Structural surveying with TLS

Figure 6-5b shows the dense point cloud obtained from a survey to document the structural health condition of the tower (Prizeman *et al.*, 2017). In this instance, a FARO Focus 3D ×130 terrestrial laser scanner was used to acquire 27 scans of the entire castle. The characteristics of the scan were (Prizeman *et al.*, 2017):

- Scan resolution of 1/5 of 28.2 MPts with 4x quality;
- Point distance of 7.67mm/10m;
- Mean registration errors of 6.6 mm;
- Maximum error of 15.5 mm;
- A minimum overlap of 12.4%;
- Registration was carried out with FARO Scene software version 5.3.

Cleaning and cropping of the point cloud were carried out to remove noise and irrelevant points (e.g. non-structural elements such as vegetation, etc.), with *CloudCompare* according to Section 3.1.3. This was an important task to ensure that the points relating to the structure only, were being considered in structural analysis. Figure 6-3e-f shows the process of segmenting the tower from the rest of the using *CloudCompare* (in form of a mesh).

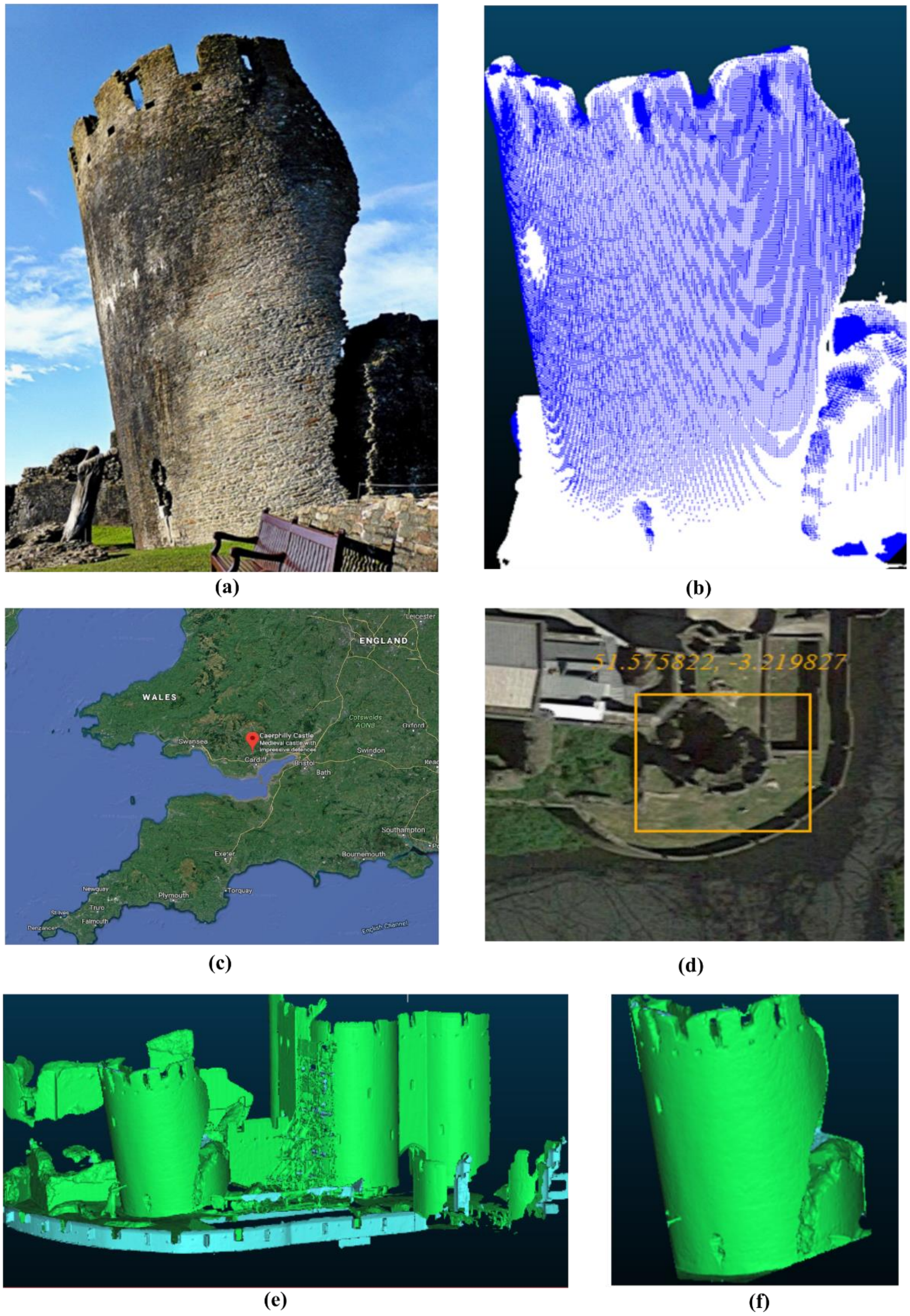


Figure 6-3: Caerphilly Castle: (a) view of the southwest leaning tower; (b) dense point cloud of the leaning tower; (c) location within Google Earth; (d) satellite image view within Google Earth; (e) mesh and (f) cropped mesh.

6.3 Stage 2: Geometric model development

The following paragraphs report the results of the implementation of the cloud-based geometric model development workflows the TLS point cloud. Various geometric models of Caerphilly tower were developed, such as those shown in Figure 6-4 with voxel sizes of 30, 40 and 50 cm. It is noteworthy that the geometric models will be herein interchangeably termed voxel models due to their construction being of voxels.

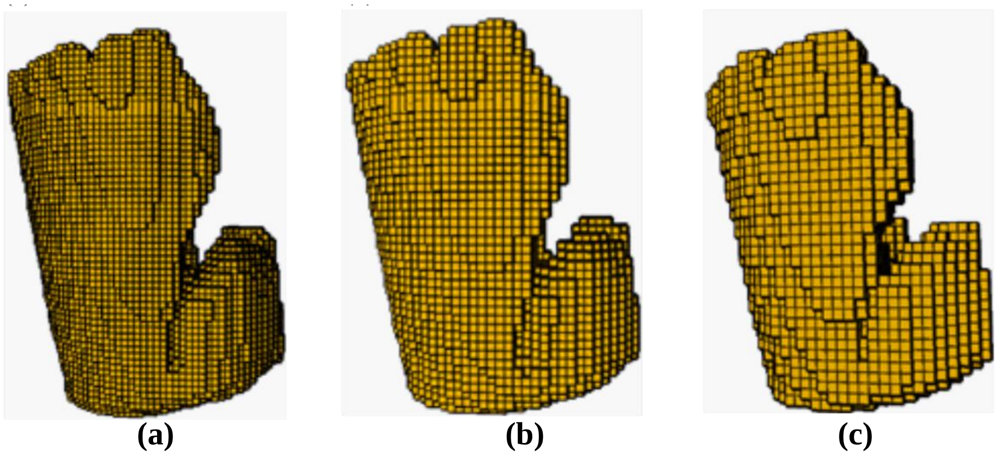


Figure 6-4: Voxel models of the tower developed using voxel sizes of: (a) 30 cm; (b) 40 cm; and (c) 50 cm

Though another three models with a voxel size of 25, 20 and 10 cm were additionally developed, structural analysis was not undertaken since they were found to be computationally unmanageable with the available computational resources. This inability to simulate smaller voxel size models was a result of the unmanageable number of free-moving blocks and total contacts. Also, as will be demonstrated further on, all the developed voxel models necessitated model refinement due to the existence of significant geometric errors. Moreover, with the 30 cm voxel size (which showed the best compromise between structural capacity estimation, geometric accuracy and computational time), a further two voxel models were developed with a voxel orientation equal to 30 and 60 degrees.

6.3.1 Geometric model development with the cloud-based workflow

Point cloud voxelization, void filling and geometric model development

The workflow commenced with the dense point cloud of Figure 6-5a with a calculated point surface density, equal to 855 points/m^2 (within *CloudCompare* according to Section 3.1.3). After the process of voxelization (shown in Figure 6-5b), the point surface density of the voxelized point cloud was significantly reduced compared to the initial dense point cloud. Figure 6-5c shows the instance of the point surface density of the 50 cm voxel model 15 points/m^2. Based upon experimentation with the above dense point cloud, the smallest voxel size permissible for the above dense point was equal to 1 cm.

After the voxelization, the filling was carried out, according to the raster image approach of Section 3.2.3. Figure 6-5d-e shows the raster images of the empty and filled voxelized point clouds for a horizontal section of the tower. This section was at an arbitrary height equal to 3.75 m. It must be noted that in Figure 6-5d-e, the active pixels or active voxels are shown in white colour. To fill the whole domain, the raster image corresponding to each of the voxelized point cloud heights was morphologically opened and closed. All these procedures were automated and incorporated into the *Matlab* script A.3.1. Finally, the geometric model (i.e. voxel model) is developed based on the filled point cloud. Indicatively, Figure 6-5f shows a voxel model with a voxel size of 50 cm, consisting of 9,407 blocks.

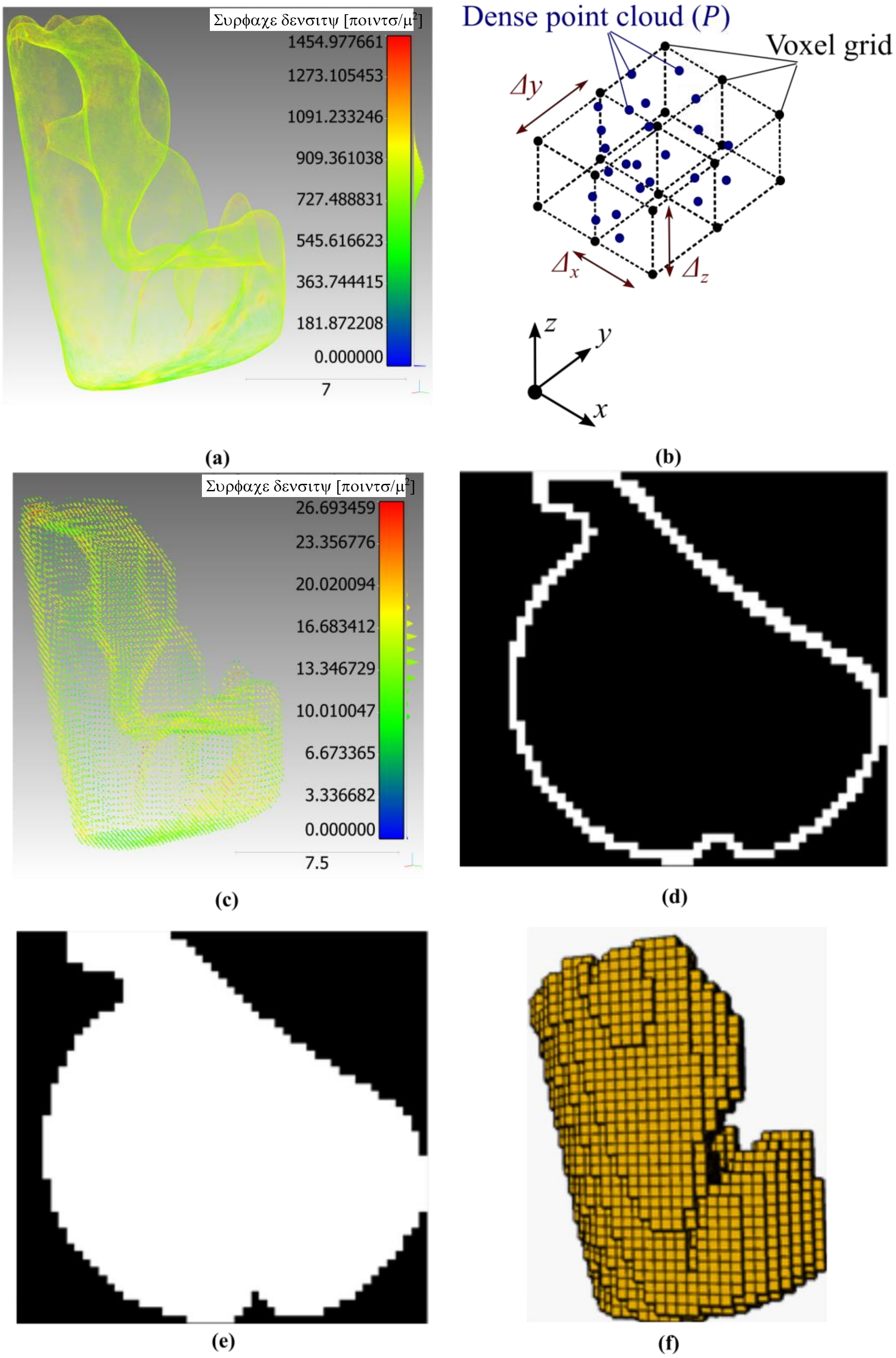


Figure 6-5: Voxelization: (a) dense point cloud; (b) voxelization process; (c) empty voxelized point cloud. Void filling: (d) empty; and (e) filled raster image for z equal to 3.75 m of the voxelized point cloud. 50 cm voxel model (f)

Geometrical assessment, and numerical model refinement

The geometrical assessment of the voxel models was carried out according to Section 3.2.3 by comparison of their geometrical properties with those of the reference mesh (as shown in Figure 6-6a). The specific reference mesh was constructed using the Poisson surface reconstruction algorithm of the dense point cloud. Also, the cloud-to-mesh distance of the mesh with 50 cm voxel model is indicatively shown in Figure 6-6b. Finally, the volumetric error, cloud-to-mesh error and the coefficient for adjusting the voxel dimension, (VAC) (calculated from (3.24)) are reported in the forthcoming paragraphs for the voxel models of varying size.

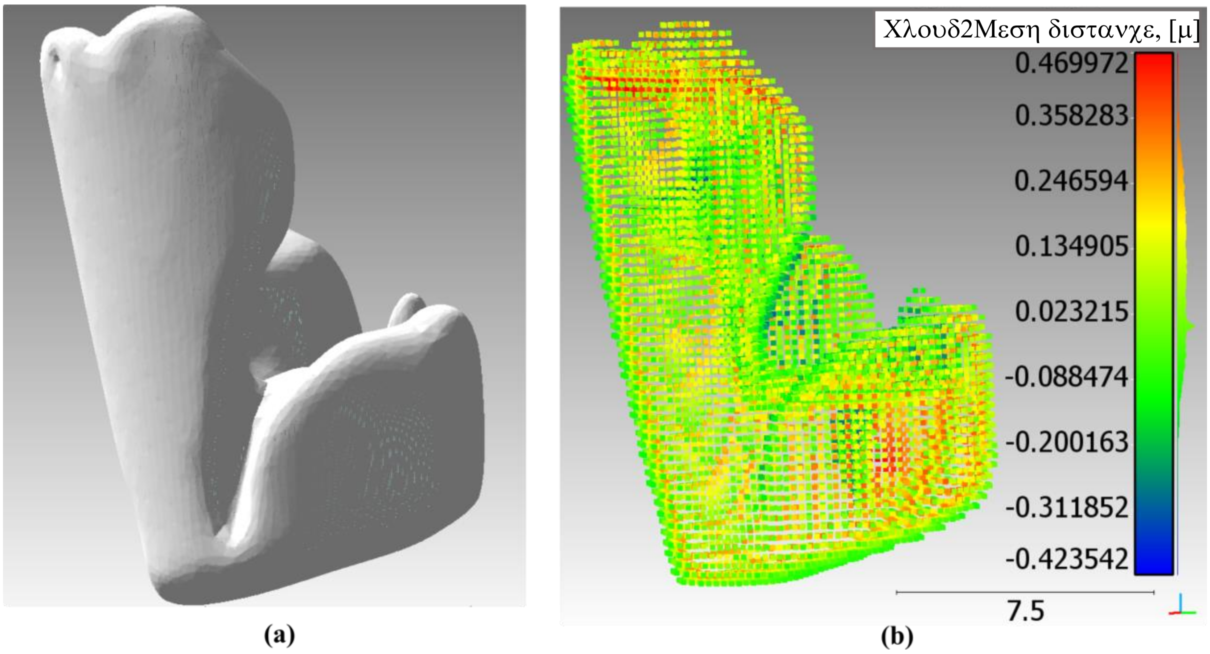


Figure 6-6: Geometric assessment: (a) watertight mesh; and (b) cloud-to-mesh distance of the 50 cm voxel model with the watertight mesh (in meters).

Table 6-1 reports the volumetric differences and cloud-to-mesh distances of all the developed voxel models. From Table 6-1, it is clear that the voxel size influences the geometrical accuracy of the voxel models. Based upon relative error with the reference watertight mesh, the volumetric accuracy exponentially increased for the decrease in voxel size. It was found that is for voxel sizes below and equal to 25 cm (that correspond to an error of equal to 5.3%) voxel models without refinement could be considered acceptable. All voxel sizes above 25 cm needed numerical model refinement. The difference in mean cloud-to-mesh distance of the voxel models and refined voxel models shows that even though the models are volumetrically accurate, the model refinement can induce error to the cross-sections of the structure.

Table 6-1: Geometrical assessment of the voxel models.

	Voxel Size	*VAC*	Volume (m^3)	Volumetric error (%)	Mean absolute C2M distance (cm)	St. dev. of absolute C2M distance (cm)
Voxel models	0.50 m	1	1028	19.50	0.1	20.4
	0.40 m	1	980	13.95	0	16.4
	0.30 m	1	933	8.46	0.1	12.1
	0.25 m	1	911	5.88	0.2	10.1
	0.20 m	1	886	2.98	0.1	8
	0.10 m	1	836	-2.80	0	3.7
Refined voxel models	0.50 m	0.943	862	0.19	8.2	27.3
	0.40 m	0.958	862	0.21	11.5	20.6
	0.30 m	0.974	862	-0.04	9	13.9
	0.25 m	0.981	860	-0.08	6.2	11.2
	0.20 m	0.990	859	-0.15	3.3	8.4
	0.10 m	1.009	859	0.19	3.5	4.8

6.4 Stage 3: Structural analysis with the discrete element method

The following paragraphs detail the numerical model development with *3DEC* according to Section 3.3.2. Then the subsequent structural analysis follows employing the various voxel models developed. Firstly, for the 0.5 voxel models which were the most computationally manageable, an initiatory study was carried out to determine the effect of the joint strength and geometric uncertainty on the structural behaviour. Thereafter, a sensitivity study was carried out on of the effect block size (for voxel sizes of 50, 40 and 30 cm). Finally, for the 30 cm voxel model which showed the best compromise between structural capacity, geometric error and simulation times, the effect of voxel orientation was also investigated on the structural capacity.

6.4.1 Numerical model development and structural analysis with 3DEC

Numerical model development

For the development of the numerical model, representative material properties for low strength masonry were adopted from previous research studies, as presented in Table 6-2. Blocks were modelled as rigid elements having a density equal to 1900 Kg/m^3. The frictional resistance between blocks was equal to 25 degrees while the joint tensile and cohesive resistance at the joints ranged from 0.25 to 0.35 MPa to represent old and deteriorated low bond strength masonry.

Table 6-2: Mechanical properties of the zero-thickness interface in the 50 cm voxel model of Caerphilly tower.

Parameter	Symbol	Unit	Model Values
Joint normal stiffness	K_n	GPa/m	20
Joint shear stiffness	K_s	GPa/m	15
Joint cohesive strength	C	MPa	0.25-0.35
Joint tensile strength	T	MPa	0.25-0.35
Joint friction	φ	°	25

A simplified assumption of planar support was taken to ensure a structural failure solely attributed to material degradation (i.e. not support failure). Since the original base of the voxel models was not flat, an additional base was added. The height of the base was determined by examining the lowest course of bricks within the tower's cloud, prior to voxelization, whilst the overall process of base addition was carried out fully automatically within the voxelization algorithm by extending the dimensionless point cloud in the z-axis. Figure 6-7a-c shows the original voxel model, the additional base and the final numerical model used for structural analysis.

After adding the base, a level plane, Πo was defined as the boundary of fixed and free-moving blocks. This is shown in Figure 6-7d (at the lowest course of the blocks of the original numerical model) and defines the boundary between the fixed and free-moving blocks. Below this level, blocks were fixed against movement in all directions (the dark grey zone shown in Figure 6-7e), while above this level the blocks were considered to represent the rubble masonry and were free to move (the silver zone shown in Figure 6-7e). By adding this additional base and ensuring planar support, failure of the structure is by material degradation only, and not support failure.

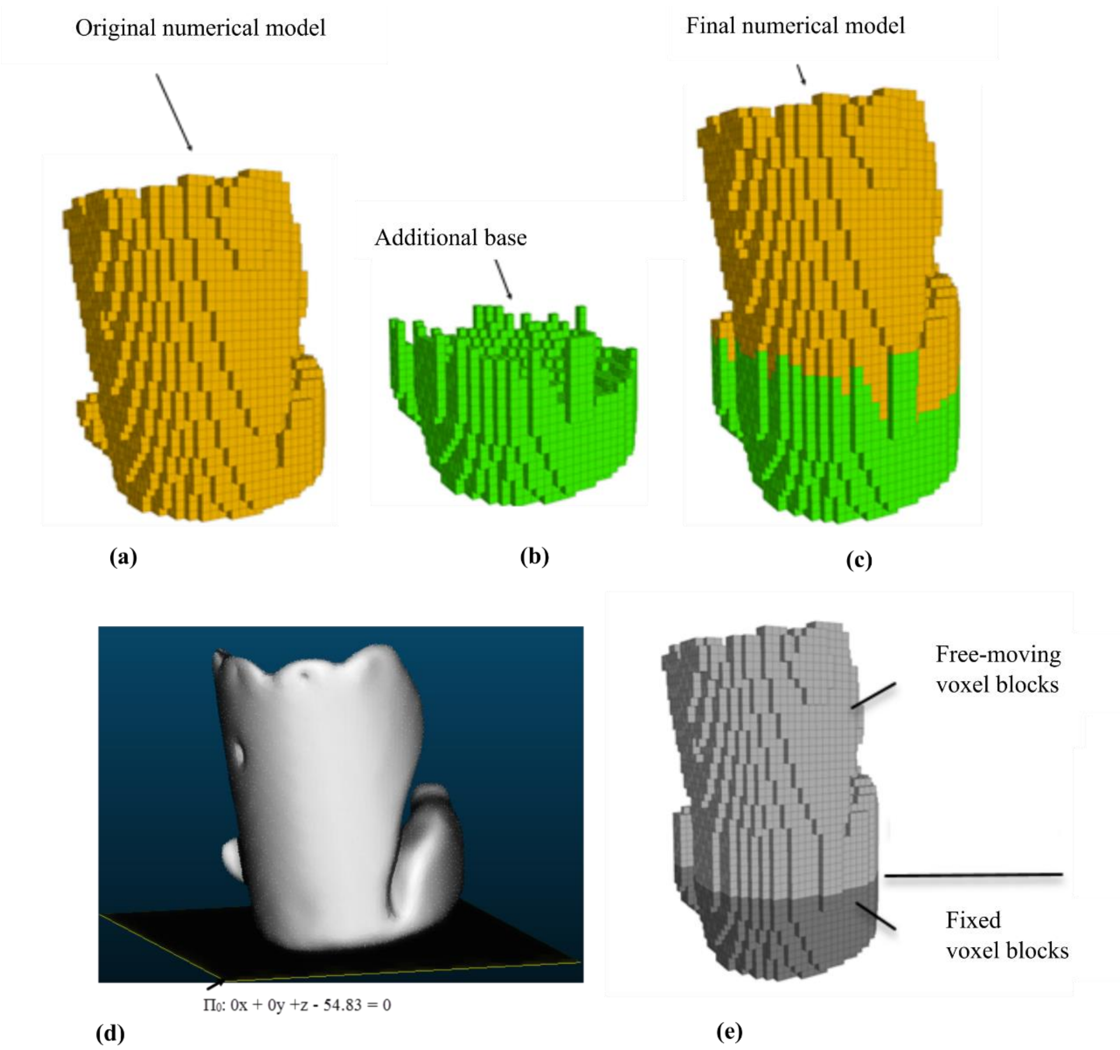


Figure 6-7: View of the numerical model of the tower developed using DEM: (a) original numerical model derived from point clouds; (b) base added to assist with the numerical simulations; (c) final model developed (note that green colour refers to an additional base while gold colour relates to the original 50 cm voxel model). Boundary conditions of tower visualised in the (d) point cloud, and (e) voxel model (i.e. geometric model).

In the numerical model, a tilt plane analysis was performed to quantify the maximum rotation angle (θ_t) of the tilt plane, according to Section 3.3.2. The maximum value of this angle measured was effectively the measure of structural capacity. In particular, the rotational angle was estimated by applying a horizontal acceleration of equal to $\lambda_h \times g$ and altering the vertical acceleration of gravity from g to a magnitude of $\lambda_v \times g$. The horizontal and vertical collapse multipliers λ_h and λ_v were obtained from the equations (3.64) and (3.65) of Chapter 3. Figure 6-8a shows a view of the tower with the gravitational acceleration components annotated. Figure 6-8b shows the plan of the tower base and azimuth (ψ) of theoretical rotation (i.e. horizontal direction in which the rotation takes place). The quadrant between the x and y-axis has previously been found to contain the range of azimuths of the structure’s existent inclination (D'Altri *et al.*, 2018b). To optimise the employed computational resources, structural analysis was only carried out for this quadrant (i.e. only for ψ equal to 0, 15, 30, 45, 60, 75 and 90 degrees).

Structural behaviour indices

Starting from a value of θ_t equal to 0 (no inclination), the theoretical inclination angle θ_t was increased incrementally. During, the simulation, the inclination angle multiplier λ_h (corresponding to θ_t) was also recorded. The critical inclination angle $\theta_{t,max}$ and multiplier $\lambda_{h,max}$ were employed to assess the load-bearing capacity of the structure, corresponding to the inclination angle multiplier at which the structure could not arrive at equilibrium at the end of a given loading cycle (corresponding to $\theta_{t,max}$). This was calculated by monitoring both the total unbalanced force of the model and the so-called inclination angle multiplier-displacement curves of strategically selected monitored points. The monitored points at Points A, B and C, shown in Figure 6-8c were strategically selected, being situated: a) on the azimuth of theoretical rotation of ψ equal to 60 °; and b) at various heights (top, mid-height and bottom). This selection of the monitored points was to ensure reliable information about the structure's behaviour was provided for global and local failure, in the principal direction of loading. It is noteworthy that further than the critical inclination angle multiplier $\lambda_{h,max}$, the critical horizontal displacements $U^A_{h,max}$, $U^B_{h,max}$ and $U^C_{h,max}$ of the monitored points A, B and C were employed as a metric of quantifying the tower's deformation capacity.

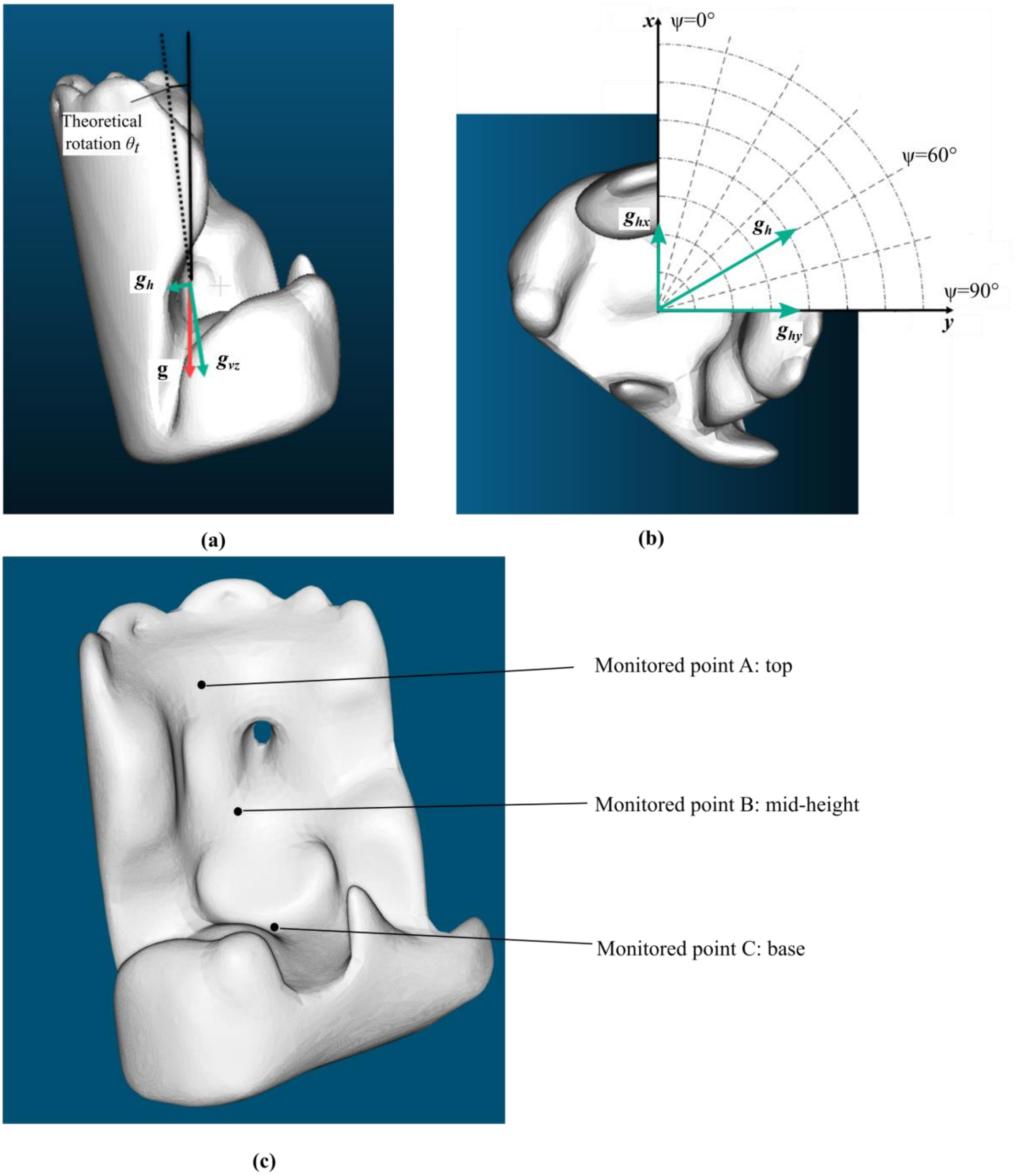


Figure 6-8: The tower with gravitational acceleration components annotated (the green vertices denote the gravitational acceleration components for a theoretical inclination angle of θ_t): (a) view; and (b) plan of the tower base with the azimuth of inclination (ψ). Monitored points A, B and C at the top, mid-height, and base of the tower (c).

6.4.2 *Influence of the joint tensile and cohesive strength on the structural capacity of the tower*

For the 50 cm voxel model, the influence of the joint strength properties (cohesive and tensile strength) upon the structural capacity was investigated. Twenty-one simulations were performed in which the joint tensile and cohesive strength ranged between 0.25 and 0.35 MPa. Figure 6-9 shows the relationship between the critical inclination angle $(\theta_{t,max})$ and azimuth angle of the theoretical rotation for all the varying inelastic joint properties for the refined voxelized 50 cm voxel model. From the analyses of the results, it was shown that an increase in the inelastic joint parameter increases the rotational angle.

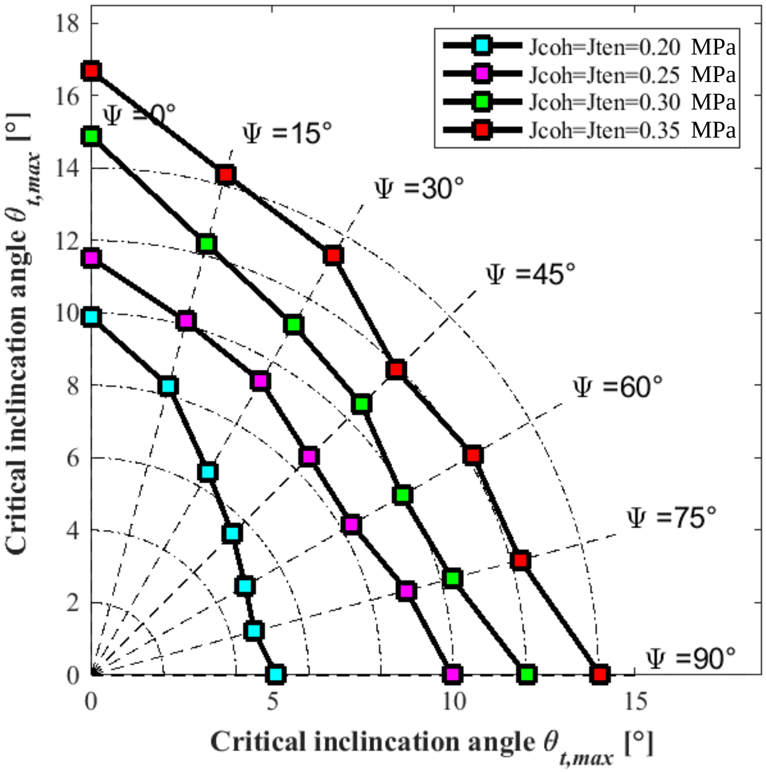


Figure 6-9: Influence of joint cohesive (symbolised c) and tensile strength (symbolised T) on the critical inclination angle ($\theta_{t,max}$) of the refined voxelized 50 cm voxel model.

The failure modes observed in all numerical simulations was the same. Indicatively, the failure mode for the 50 cm voxel model for an azimuth of theoretical rotation ψ equal to 60° is shown in Figure 6-10. In the tower, cracks initially developed at the junction of the remaining up-right body and base of the tower, gradually progressing towards the base, leading to collapse as the tower rotated and separated from its base. Similar failure modes were observed in a previous study (D'Altri *et al.*, 2018b).

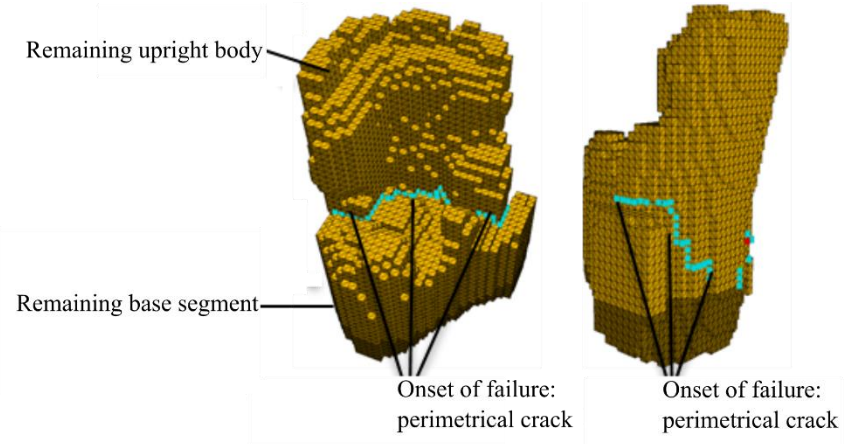


Figure 6-10: Failure mode of 50 cm voxel model (azimuth of theoretical rotation ψ equal to 60°), the blue markers denoting joint tensile failure.

6.4.3 Influence of the geometrical uncertainty on the critical inclination angle

To assess the influence of geometrical uncertainty, twenty-one numerical analyses were executed with a 50 cm voxel model, yet without refinement as plotted in Figure 6-11. Moreover, in Figure 6-12, the normalised uncertainty in the structural capacity for the voxel models without refinement various values of joint strength. The failure modes were the same as those of observed in Figure 6-10. From Figure 6-12, two observations regarding the sensitivity of the structural capacity on geometrical accuracy can be made: a) the structural behaviour uncertainty was significant (12.5%) due to the given geometric uncertainty; and b) the structural behaviour uncertainty varied for the azimuth of loading which suggested that the influence of geometrical uncertainty was not equally distributed.

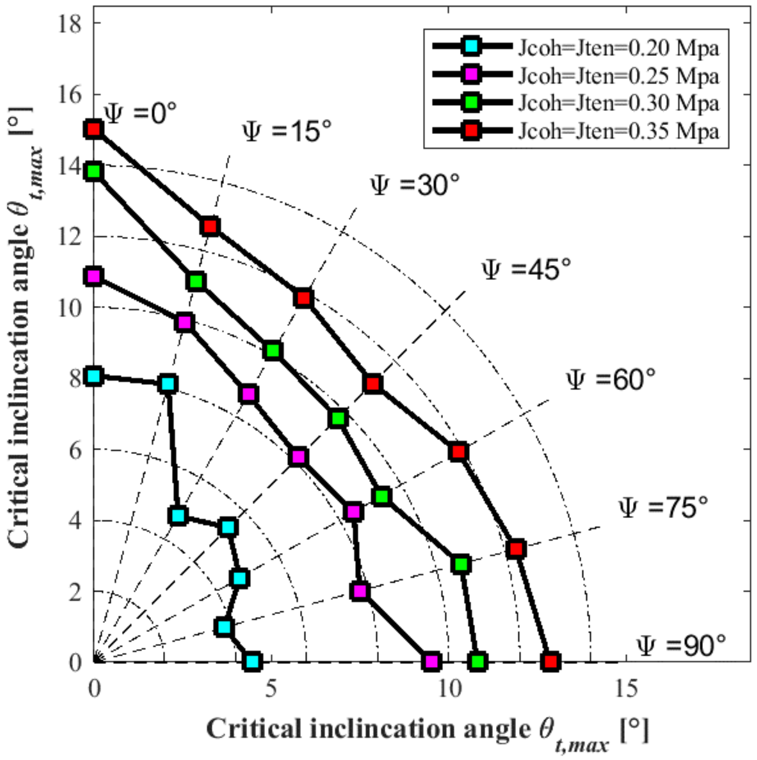


Figure 6-11: Influence of joint cohesive (symbolised c) and tensile strength (symbolised T) on the critical inclination angle ($\theta_{t,max}$) of the non-refined 50 cm voxel model.

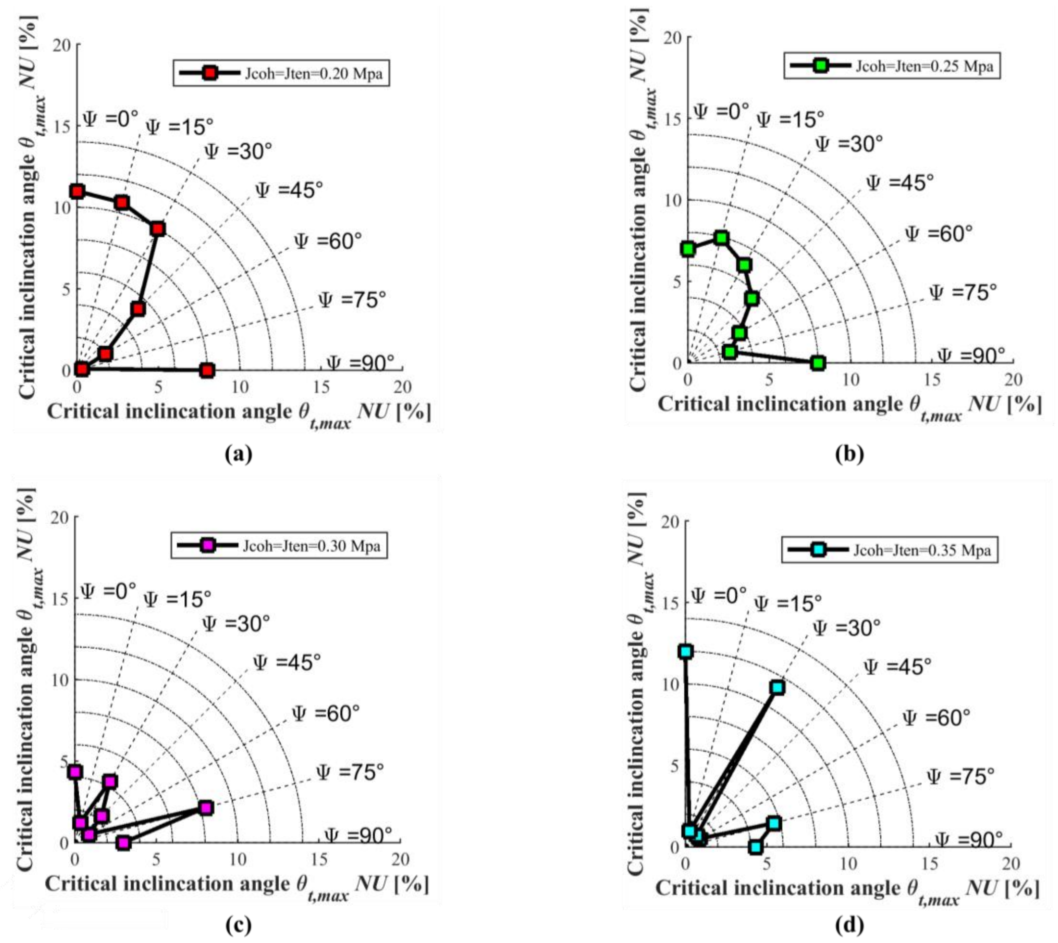


Figure 6-12: Normalised uncertainty of structural behaviour of 50 cm voxel model for joint tensile strength, T equal and joint cohesive strength, c equal to: (a) 0.20 MPa; (b) 0.25 MPa; (c) 0.30 MPa; and (d) 0.35 MPa.

6.4.4 Sensitivity study on the voxel size

Influence of the voxel size on structural capacity

The influence of the voxel size upon the structural capacity was investigated for T equal to C, equal to 0.25 MPa, which were deemed appropriate from the initiatory studies of Section 6.4.2. This entailed a further 14 numerical simulations which were carried out with the 40 cm and 30 cm voxel models. Figure 6-13 shows the relationship between the critical inclination angle $(\theta_{t,max})$ and the azimuth angle (ψ) of the theoretical rotation for the towers constructed using voxel sizes of 50, 40 and 30 cm, respectively. The relative displacement between the curves of Figure 6-13 shows that a decrease in voxel size decreases the structural capacity of the models.

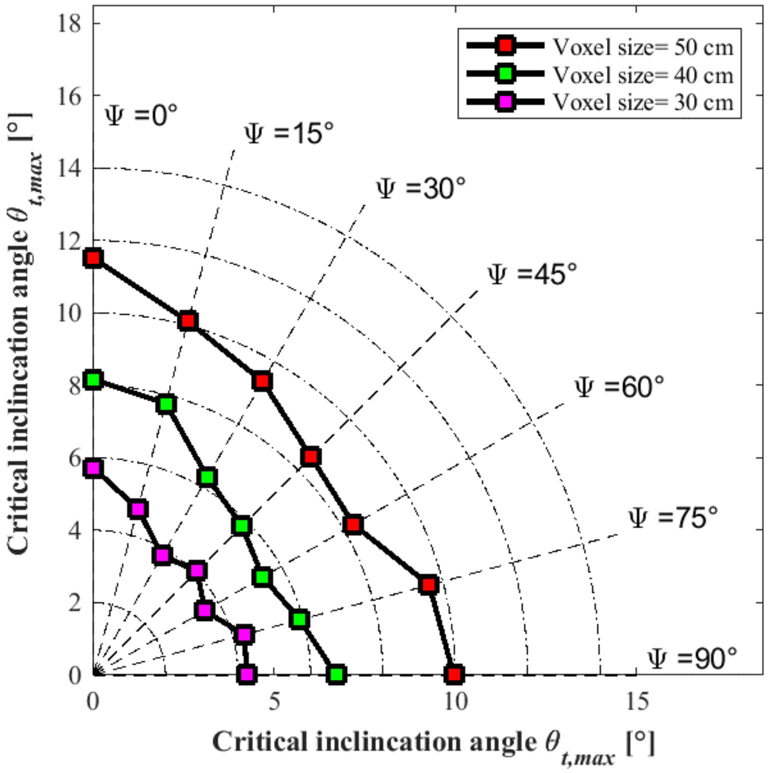


Figure 6-13: Influence of voxel size upon critical inclination angle ($\theta_{t,max}$) for joint tensile, strength, T equal to 0.25 MPa and joint cohesive strength, c equal to 0.25 MPa.

Furthermore, for an azimuth of inclination (ψ) equal to 60°, the influence of the voxel size upon the structures' load-bearing capacity, deformation capacity was investigated. Table 6-3 reports the results of the structural analysis of the 50, 40 and 30 cm voxel models for an azimuth of inclination (ψ) equal to 60°. From Table 6-3, its evident that the decrease of voxel size is associated with: a) decrease in load-bearing capacity (i.e. the critical inclination angle multiplier); b) decrease in the deformation capacity (i.e. the critical horizontal displacements of monitored points A, B and C). This is in agreement with a previous experimental investigation (Petry and Beyer, 2014)and a numerical investigation on the out-of-plane loading of masonry structures with the DEM (Godio *et al.*, 2018), that block size significantly influences the structural capacity. The reasoning behind this that effectively, the joints form planes of weakness in the structure. As the voxel size decreases, the number of joints significantly increases, resulting in a consequent reduction of the structural capacity. It is hypothesised that this reduction would become negligible for a voxel size lower than a certain threshold, yet this should be examined in further investigation. Due to computational resource limitations, it was not possible to simulate models of free moving voxels with voxel sizes smaller than 30 cm.

In Figure 6-10a-b, Figure 6-10c-d and Figure 6-10e-f respectively. The failure modes and inclination angle multiplier-displacement curves are shown for the 30, 40 and 50 cm voxel models for an azimuth of inclination (ψ) equal to 60° It's noteworthy that the blue, green and red cuboids located on the joints signify joint tensile failure, current joint slipping and past slipping respectively. Furthermore, the displacement contours of the blocks are plotted demonstrating the magnitude of displacement due to inclination. From Figure 6-10a, Figure 6-10c, Figure 6-10e, it is evident that failure modes of the models were common and consisted of perimetrical cracks developed at the junction of the remaining up-right body and base of the tower progressing towards the base. With the application of any further inclination, the remaining up-right body detached and began to rotate freely (simultaneously breaking up into pieces. Similar failure modes were observed in a previous study (D'Altri *et al.*, 2018b).

Table 6-3: Influence of voxel size on structural capacity.

Voxel size (cm)	$\lambda_{h,max}$	$U^{A}_{h,max}$ (mm)	$U^{B}_{h,max}$ (mm)	$U^{C}_{h,max}$ (mm)
50 cm	0.18	6	3.8	1.8
40 cm	0.14	5.6	3.0	1.4
30 cm	0.06	2.0	1.0	0.3

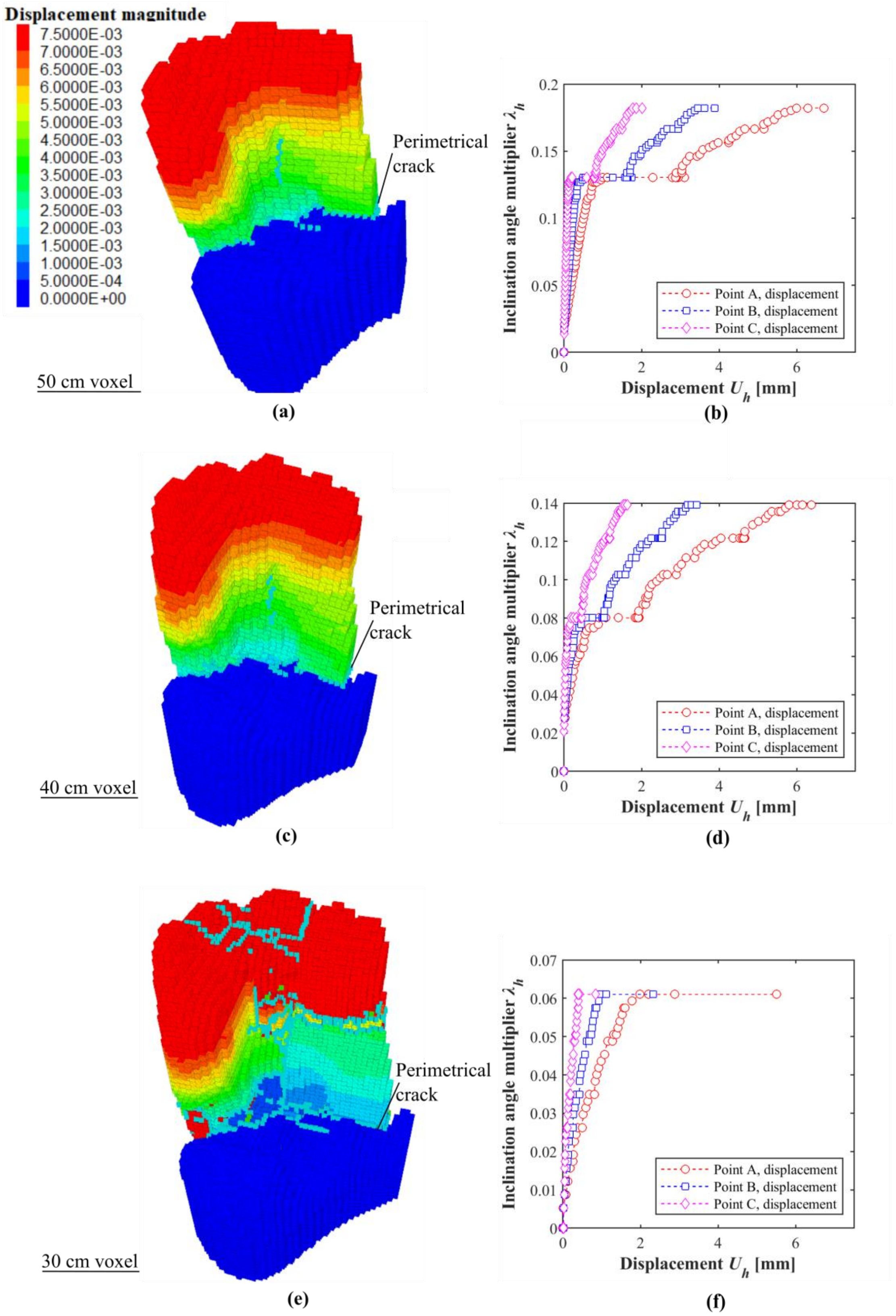


Figure 6-14: Failure modes and inclination angle multiplier-displacement curves (different scale) of: (a-b) 50 cm; (c-d) 40 cm (e-f) 30 cm voxel size models (azimuth of inclination ψ equal to 60°). The blue markers denote joint tensile failure while the displacement contour is common and in meters.

Influence of the voxel size on numerical model properties simulation times

To assess the cloud-based approach's efficiency and computational demands, model characteristics such as block and contact numbers for different voxel model sizes were compared with the times needed for model development and simulation. Table 6-4 reports the block and contact numbers of each model developed. Table 6-5 shows the computational times needed using an Intel(R) Xeon(R) CPU @ 3.00 GHz processor and 64 GB memory RAM. Model development refers to the time needed for converting the dense point cloud into a voxel model and (including adding the base) while equilibrium refers to the time needed to equilibrate the voxel model and failure refers to the time needed to load to failure from equilibrium for an azimuth of theoretical rotation ψ equal to 0. From Table 6-4 and Table 6-5 the following observations can be made: a) concerning the geometric model properties and computational times, all the quantities were inversely and exponentially proportional to the voxel size; b) the larger the size of the voxel, the faster the numerical simulation is; c) the best compromise between structural capacity estimation, geometric accuracy and computational time was represented with the 30 cm voxel size.

Table 6-4: Numerical model properties: Number of contacts and blocks of Caerphilly tower for varying voxel size.

Voxel Size	Blocks	Face-face contacts	Edge-edge contacts	Vertex-vertex contacts	Total free contacts
50 cm	13,385	37,395	71,734	46,427	155,556
40 cm	22,532	63,608	122,835	79,853	266,296
30 cm	47,827	136,636	265,964	173,849	576,449
25 cm	74,969	215,498	421,024	275,969	912,491
20 cm	111,821	846,850	1,053,624	651,740	2,552,214
10 cm	844,343	2,482,088	4,913,220	3,250,455	10,645,763

Table 6-5: Numerical model processing times.

Voxel size	Numerical model development	Equilibrium	Failure	Total time
50 cm	00:01:00	00:08:00	01:00:00	01:09:00
40 cm	00:03:00	00:15:00	04:01:00	04:18:00
30 cm	00:06:00	00:49:00	05:20:00	07:15:00
25 cm	00:11:00	-	-	-
20 cm	00:22:00	-	-	-
10 cm	01:50:00	-	-	-

6.4.5 Influence of the voxel orientation on the structural capacity of the tower

For the voxel size of 30 cm, the influence of the voxel orientation upon the structures' load-bearing capacity, deformation capacity and failure mode were investigated. Concerning Figure

6-15, the horizontal voxel principal directions were altered by rotation of the dense point cloud, before voxelization. Two simulations were performed whereby the dense point cloud was rotated by 30 and 60 degrees before voxelization around the z-axis and loaded for ψ equal to 60°. Table 6-6reports the results of the structural analysis of the models with a voxel rotation. From Table 6-6, it is evident that, in comparison with the original 30 cm voxel size model of Table 6-3, the models with a voxel rotation demonstrated: a) comparable load-bearing capacities (i.e. the critical inclination angle multiplier); and b) comparable deformation capacities (i.e. the critical horizontal displacements of monitored points A, B and C). Furthermore, the failure modes and inclination angle multiplier-displacement curves of the models with the voxel rotation are shown in Figure 6-15a-b and Figure 6-15c-d respectively. Comparing the differences between Figure 6-10e-f and Figure 6-15, it is evident that the failure modes and inclination angle multiplier-displacements curves between the models with voxel rotation and original 30 cm voxel size model show a good agreement.

Table 6-6: Influence of voxel orientation on structural capacity.

Voxel rotation	$\lambda_{h,max}$	$U^{A}_{h,max}$ (mm)	$U^{B}_{h,max}$ (mm)	$U^{C}_{h,max}$ (mm)
30°	0.06	2.0	1.0	0.4
60°	0.08	2.4	1.0	0.4

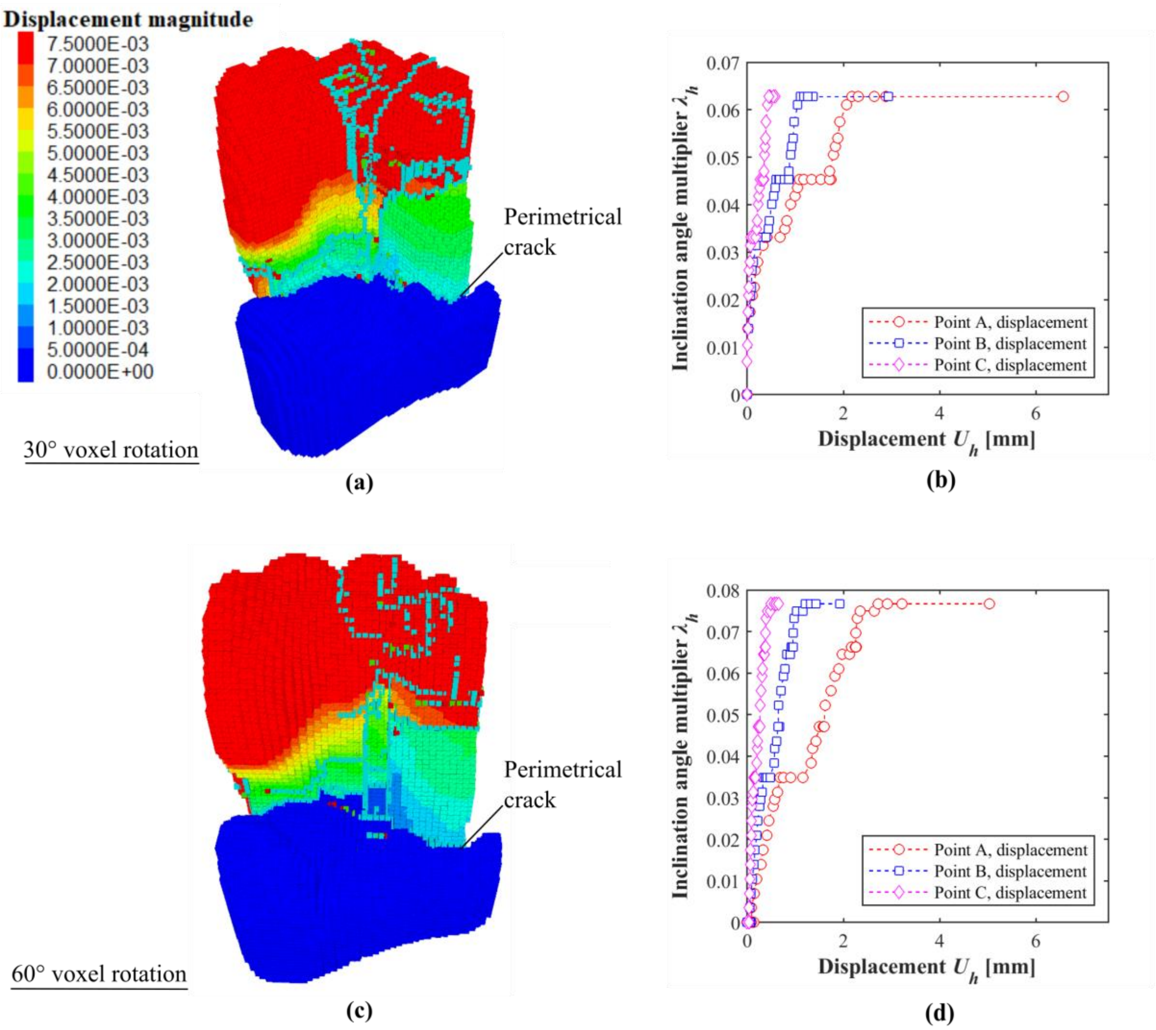


Figure 6-15: Failure modes and inclination angle multiplier-displacement curves of 30 cm voxel size with: (a-b) 30° voxel rotation; and (c-d) 60° voxel rotation. The blue markers denote joint tensile failure while the displacement contours are common and in meters.

6.5 Summary

In this Chapter, the methodological framework was implemented on a full-scale rubble masonry structure. Various voxel models (i.e. geometric models of voxels) were developed with the cloud-based-workflow employing various voxel sizes and orientations. A tilt-plane analysis was numerically simulated with the DEM. For the course voxel model of a 50 cm voxel size, an initiatory study was carried out to investigate the effect of: a) joint strength properties; b) the geometric uncertainty was investigated on the structural behaviour. Next, the effect of voxel size was investigated on the structural behaviour, numerical model properties and simulation times. Finally, for the 30 cm voxel models which were found to be the best compromise between simulation times, geometric accurate and structural capacity, the effect of voxel orientation was also investigated on the structural capacity.

For the course voxel model of 50 cm, the variation in the joint strength properties, a linear variation in the structural capacity was found. Furthermore, the geometrical uncertainty of the models was found to cause uncertainty of up to 12.5%. This finding suggests that, while developing the voxel model, particular care must be taken to employing both the correct joint strength properties and the accurate geometrical representation of the structure to ascertain the robustness of the structural analysis. Concerning the study on the effect of voxel size, as the voxel size decreases, the structural capacity and volumetric error decrease yet the computational time required to perform structural analysis increases dramatically and could lead to models that cannot be handled with standard workstations. For the case study investigated (i.e. the Caerphilly tower), for a course voxel size of 50 cm, structural analysis was carried out in a manageable computational time of 71 minutes for a geometric model with 9,000 blocks, 100,000 contacts. The best compromise between, geometric accuracy and computational time were achieved with a voxel size of 30 cm, which is also close to the size of the masonry stones observed on the structure. However, such models resulted in a less manageable computational time of approximately 238 minutes. Finally, the examination of the influence of the voxel orientation on the 30 cm model demonstrated that voxel orientation did not significantly affect the structural behaviour.

Therefore, from the above findings, the potential of the methodological framework to perform semi-automated and robust discrete element modelling of rubble masonry structures has been demonstrated. Through the employment of "*Cloud2DEM*", this study paves the way for the automated structural analysis of rubble masonry structures using discrete element modelling.

Chapter 7. Conclusions and future research

This Chapter firstly examines this research's conclusion and primary outcomes. Next, the investigation's course is evaluated by examining how its aim and objectives were addressed. Finally, the limitations are highlighted and recommendations for future research are made.

7.1 Conclusion

The research questions found in this thesis regarded the interface of structural engineering, geospatial engineering and computer vision. Firstly, this research intended to answer the question of the importance of geometrical accuracy in the structural analysis of masonry structures. Then, the potential of non-contact sensing techniques for improving the overall efficiency and robustness of high-level modelling such as the DEM was examined. Finally, the potential conjuncture of structural engineering and geospatial engineering through the employment of computer vision was assessed. Through the various approaches proposed and studies carried out, it was indeed found that geometry is an extremely important factor which must be considered by carrying out structural analysis. Then, it was demonstrated that the employment of modern geospatial techniques can be an effective manner for improving the efficiency and reliability of the structural analysis. As such, it was additionally demonstrated that low-cost, structural surveying techniques such as SfM photogrammetry can serve for accurately obtaining and reconstructing extremely complex geometries. Finally, the potential of employing computer vision was also highlighted to improve structural engineering though automatically converting the geometric data into accurate numerical models. All in all, concluding the above finding have the potential to significantly improve the structural analysis of masonry structure.

7.2 Primary outcomes

This research aimed to develop a methodological framework to improve the efficiency and robustness of the structural analysis of masonry structures using discrete element modelling. In pursuing this aim, various points were addressed and approaches were developed as summarised below.

a.) Quantifying the effect of geometric uncertainty on the robustness of structural analysis

To employ geometrically-accurate approaches of numerical modelling, a clear justification until now lacked. As found in Section 2.2.3, despite the evidence that geometric uncertainty introduces uncertainty in the structural analysis, until now, the understanding of the effect of geometric uncertainty was limited to the ultimate limit states. Furthermore, this was only for the uncertainty of invariable geometrical properties (i.e. arch span, rise) and invariable structural behaviour indices (i.e. collapse load and load position). In this investigation, for the first time, a comprehensive insight into the effect of geometric uncertainty was provided. The current state of knowledge was extended for the serviceability limit states, whilst uncertainty was also accounted for, on a block-based level (e.g. joint forces, block displacements). Through this contribution, practising engineers and researchers employing high-level numerical methods such as the DEM can be aware of the potential influence of geometric uncertainty on the robustness of the structural analysis.

b.) The manual image-based approach "manual Image2DEM"

With the manual image-based approach, the structural engineer can capture images of the masonry structures, in situ with a low-cost camera and carry out geometrically-accurate discrete element modelling of regular masonry structures. Although approaches for image-based numerical modelling exist, methodical frameworks entailing the geometrical and structural behaviour uncertainty assessment lacked. Effectively, never until now, has the process of geometric and structural behaviour assessment been formalised. Therefore, the manual image-based approach paves the groundwork for the methodical discrete element modelling of regular masonry structures. It is also envisioned that, due to its low-cost and straightforwardness, it can additionally expedite the employment of the DEM for the structural analysis of regular masonry structures. It is important to note that whilst blocks of the employed orthoimagery can be either distinguishable or non-distinguishable, it may become unmanageable for structures with many blocks (i.e. over one thousand blocks).

c.) The semi-automated image-based approach "semi-automated Image2DEM"

With the semi-automated image-based approach, the structural engineer can capture images in situ with a DSLR camera of the structure and carry out geometrically-accurate discrete element modelling of regular masonry structures in a manageable time, providing that the blocks of the orthoimage are distinguishable. Whilst computer vision techniques have been successfully used for block segmentation on many occasions, block-based modelling of regular masonry has not until now been automated. In fact, in the author's knowledge, this is the first approach in literature which permits the semi-automated discrete modelling of regular masonry structures. Through this contribution, the ground is laid for the automated discrete element modelling of regular masonry structures using discrete element modelling which would be of immense importance for structures such as masonry arch bridges. For instance, as reported in Section 2.1.2, the current most common assessment of methods of masonry arches and bridges are through FEM and LA block-based models which are either too computationally expensive or oversimplified. Through this investigation, avenues are opened for practising engineers and researchers to replace the former methods with the DEM whilst asset managers can additionally benefit from the high-level structural analysis the DEM provides. Finally, it is to be noted that this approach can be employed with other numerical modelling approaches of the block-based strategy (i.e. such as the LA block-based models, FEM block-based models, DDA, NSCD, FDEM) by simply replacing the DEM in Stage 3 of the framework.

d.) The cloud-based approach "Cloud2DEM"

With the cloud-based approach, the structural engineer can acquire a point cloud (whether in TLS or SfM photogrammetry) and carry out semi-automated geometrically-accurate discrete element modelling of rubble masonry structures. Although voxelization has been previously employed for masonry with the FEM continuum models, it has never been employed for discontinuum numerical methods such as the DEM. It is envisioned that it, through this contribution, the ground is laid for the automated discrete element modelling of rubble masonry structures, such as the tower of this study. For instance, as found in the introduction of Chapter 6, the current assessment methods of rubble masonry are mostly limited to the continuum FEM models (i.e. smeared crack models) and FELA, which as demonstrated, can be oversimplified. Through this investigation, avenues are opened for practising engineers and researchers to replace the former methods with the DEM whilst asset managers can additionally benefit from the high-level structural analysis the DEM provides.

7.3 Reviewing research objectives

As stated in Chapter 1, this investigation aimed *to develop a methodological framework to improve the efficiency and robustness of the structural analysis of masonry structures using discrete element modelling.* This aim was achieved by the accomplishment of the main objectives as such:

1.) *To evaluate the suitability of the DEM for the structural analysis of masonry structures*.

This objective was addressed in Chapter 2 through the reviewing of the state-of-the-art numerical modelling approaches in Section 2.1 and Section 2.2. Whilst the most commonly employed numerical methods are the LA and FEM, these were found to be either oversimplified or ineffective for masonry. There are emerging FEM block-based models, that however lack application and have not yet been employed for large scale masonry structures. The DEM, on the other hand, has been employed on a multitude of occasions for block-based modelling demonstrating its potential for capturing the in-service and collapse behaviour of masonry, owing to its numerical formulation. However, a major limitation of the DEM found was the fact that the majority of state-of-the-art studies employ either ad-hoc, simplified geometric models. Therefore, as a consequence of this, both the efficiency and robustness of the structural analysis is reduced. The non-employment of accurate geometric models was given to three main reasons: a) difficulty in geometric data acquisition; b) lack of a methodical framework for developing geometric models; and c) lack of comprehensive investigation that justifies the employment of accurate geometric model.

2.) *To both examine the suitability of SfM photogrammetry for rapidly providing accurate geometric data and approaches for automatically developing geometric models for the DEM from such data*.

This objective was addressed in Chapter 2. Firstly, concerning structural surveying, various geospatial techniques were reviewed in Section 2.2. Through reviewing various investigations, it was found that SfM photogrammetry can provide geometric data of a similar level of accuracy with benchmark point-based techniques such as a total station whilst at a similar resolution of benchmark non-contact sensing techniques, terrestrial laser scanner. Additionally, the SfM photogrammetry pipeline was also found to be advantageous due to its low operational costs and straightforward approach. Secondly, concerning geometric model development, various approaches were reviewed, in Section 2.4, distinguished according to the modelling strategies. Through the reviewing of various studies, the potential of image-based and cloud-based workflows was particularly highlighted to automatically develop geometric models for the DEM.

3.) *To propose and develop a methodological framework for the structural analysis of masonry structures from discrete points, orthoimages and point clouds with the discrete element method.*

This objective was addressed in Chapter 3 in which the three-stage methodological framework was proposed and is associated with research outcomes b.), c.) and d.). The first stage of the methodological framework entailed structural surveying with various geospatial techniques. It explained the numerical formulation of the SfM photogrammetry pipeline and details the steps of its employment for structural surveying. Then the bridging of geometric data from various geospatial techniques (discrete points, orthoimages and point clouds) was addressed. Finally, the numerical formulation of the DEM was explained, as were the steps for its employment detailed for the structural analysis of masonry. Three approaches stemmed from the methodological framework according to the type of geometric data employed, the point-based, image-based, and cloud-based approaches. It is noteworthy that for the study of Chapter 4, an additional approach (termed point-based approach, *"Point2DEM"*) employing discrete points was proposed added, however, this does not form part of the core investigation and is not considered a main approach.

4.) *To quantify the effect of geometric uncertainty on the robustness of the structural analysis of regular masonry structures*.

This objective was addressed in Chapter 4 and corresponds to the research outcomes a.) and b.). Specifically, with the implementation of the manual image-based approach on 25 arch specimens, geometric models were developed from SfM photogrammetry and tape measurements, respectively. The geometric and structural behaviour uncertainty was calculated between the varying approaches (and consequent varying geospatial techniques). The investigation on the effect of geometric uncertainty between geometric models from SfM photogrammetry and tape measurements showed that even seemingly minor differences in geometry can cause significant differences in the structural behaviour. Effectively, the employment of geospatial techniques of a varying accuracy (i.e. traditional geospatial techniques) led to the compromising of the robustness of the structural analysis. Therefore, geometric uncertainty should be accounted for robust structural analysis. Moreover, while most past researchers have only emphasised the geometry uncertainty of the invariable geometrical properties (e.g. arch span, rise) the geometrical uncertainty of the joints, and in specific, the joint inclination angle (i.e. joint dip) was found to significantly influence the structural behaviour. Therefore, to accurately capture structural behaviour masonry structures, accurate geometric models should be employed for geometric model development and particular attention to the geometrical accuracy of the joints should be ensured to minimize the uncertainty of the robustness of the structural analysis.

5.) *To demonstrate the framework's potential to perform semi-automated and robust discrete element modelling of regular masonry structures through the employment of image processing techniques (IPTs).*

This objective was addressed in Chapter 5 and corresponds to the research outcome c.), with the implementation of the semi-automated image-based approach on the same 25 arch specimens of Chapter 4. Effectively, the investigation of the employment of IPTs demonstrated that when high-quality orthomosaic with distinguishable blocks is employed, a semi-automated structural analysis can be carried out. Additionally, the structural analysis was also found to be robust, owing to the good agreement of both geometric model and structural behaviour with accurate geometric models of manual CAD-based block segmentation.

6.) *To demonstrate the framework's potential to perform semi-automated and robust discrete element modelling of rubble masonry structures through the employment of voxelization.*

This objective was addressed in Chapter 6 and corresponds to the research outcome d.). In specific, the cloud-based approach was implemented on a full-scale rubble structure. The potential of the employment of voxelization was demonstrated in the performing of unprecedented structural analyses with the DEM, within a manageable time, and of a highly complex rubble masonry structure. A crucial factor in the workflow was the voxel size which was directly correlated to the geometrical uncertainty, computational times, and structural capacity of the workflow.

7.4 Limitations and future research

Whilst the effectivity of this research was demonstrated in the previous Sections, limitations were also observed. The following limitations were found which can be associated with the previous contributions as such:

1.) ***Quantifying the effect of geometric uncertainty on the robustness of structural analysis***

Through the application of "*manual Image2DEM*", it was found that accurate geometric models should be employed, to ascertain the robustness of the structural analysis. However, this was only demonstrated for small-scale structures. Given that the size of masonry blocks plays a significant role in masonry (as found in Chapter 6), additional numerical and experimental investigations are scheduled to assess the validity of this investigation's findings on a large-scale structure, which however were not carried out due to limitations of time. Additionally, in this study, the loading was only quasi-static. The effect of geometric uncertainty should be investigated for the case of dynamic loading, such an earthquake. Moreover, it should be noted that the structural analysis of masonry buildings is affected by many uncertainties and characterized by issues related to the very nature of the material. Whilst the proposed framework found that provide accurate geometric models are necessary, issues such as the aforementioned should also be accounted for, to avoid leading to unrealistic conclusions. Finally, while this study examined the effect of geometric uncertainty between varying geospatial techniques, a future investigation should also examine the difference between ad-hoc and geometrically-accurate geometric models to further consolidate the evidence on the importance of geometry.

2.) Enhancing "semi-automated Image2DEM" with machine learning techniques

Through the application of the "*semi-automated Image2DEM*", the potential of employing IPTs was demonstrated for providing, accurate geometric models and permitting both semi-automated and robust discrete element modelling. However, whilst ITPs were demonstrated as effective on the specific dry-jointed masonry structure of this study, their dependency on a block-joint colour contrast means that the current approach lacks in transferability and robustness. In the current study, this limitation was overcome by employing orthoimage of a relatively small (i.e. 0.1 mm in comparison with 5 mm of (Napolitano *et al.*, 2019c)) GSD_{max}, which can be computationally unmanageable. A recommendation to overcome this such limitation is by replacing the employment of IPTs with machine learning techniques, as in Forster *et al.* (2019) which have a lesser dependency on the block-joint colour contrast. A second inherent limitation of the image-based models is discussed here, as apparent in Section 2.4.2, which is related to their two-dimensional nature. One way to overcome the limitation is through geometrical modifications within the DEM software, and specifically through the segmenting the blocks along the transverse direction into varying domains (i.e. by sequentially segmenting an image-based geometric model into segments). A second way can be through the employment of z-axis data of the orthoimagery structure.

3.) Enhancing "Cloud2DEM" with Voronois, GPR and a monitoring strategy

Through the application of the "*Cloud2DEM approach*", it was found that techniques such as voxelization (i.e. spatial enumeration) are effective for geometrically describing highly-irregular rubble structures with discontinuum methods such as the DEM. However, to increase the robustness of the structural analysis of the proposed approach, further investigation would be beneficial on the effect of the mechanical properties of the interface, the size of the block and orientation of the blocks, and computation resource optimisation techniques. Furthermore, since rubble masonry possesses voids and flaws, a future investigation should also be carried out to simulate this by removing voxels in critical locations. Additionally, due to the orthogonal nature of the voxel blocks, it is thought that the discontinuities may predispose failure in the principal directions of the blocks. To overcome these difficulties, future investigation is scheduled for employing other approaches of spatial enumeration, such as Voronoi blocks, in (Sarhosis and Lemos, 2018; Pulatsu *et al.*, 2019b; Sarhosis *et al.*, 2019) to: a) better approximate the random nature of masonry; and b) avoid the predisposition of failure in the principal directions of the voxel blocks.

Of major importance is also the definition of the internal geometry of the masonry. It is well-known that headers[10] for instance, can highly affect the structural response of multi-leaf, typically stone masonry, especially the out-of-plane behavior. Additionally, cracks, unconnected wall panels, defects are well known to affect the structural response of a historic building. At the present moment, the experienced contribution of a structural engineer is still necessary for the definition of the internal geometry (i.e. cracks, headers and defects). Future research should examine a systematic and automated definition of the internal geometry masonry structure, for example by employing GPR, as in Solla *et al.* (2012).

Finally, another interesting aspect to be incorporated into the methodological framework would be the addition of structural monitoring. Given that important heritage monuments often need to be monitored for the case that structural defects develop or if the structure approaches a limit state, the framework can be used (i.e. the structural surveying and analysis data) to monitor the development of new cracks or the movement (e.g. the rotation of the southwest leaning tower of Caerphilly Castle) and manageable time and cost-effective way. Future research could employ methodological framework (i.e. especially the structural surveying stages) for monitoring, such as the multi-temporal monitoring strategies employed for geomorphological monitoring, in the spirit of Peppa (2018).

[10] Headers are bricks or stones which lie with their greatest length at right angles to the face of a masonry panel. In case of stone masonry, a header is sometimes known as through stone, whilst the course of brick work in which all the bricks are laid as headers is known as header course.

Bibliography

Abruzzese, D., Miccoli, L. and Yuan, J. (2009). Mechanical behavior of leaning masonry Huzhu Pagoda. *Journal of Cultural Heritage* 10, pp. 480-486. Doi: 10.1016/j.culher.2009.02.004

Acary, V., Blaise, J., Drap, P., Florenzano, M., Garrec, S. and Jean, M. (1999). NSCD method applied to mechanical simulation of masonry in historical buildings using MOMA. *International Symposium WG3 on Simple methods for architectural photogrammetry*. Olinda, Brazil, Oct 1999. pp. 225-237

Addessi, D. and Sacco, E. (2016). Nonlinear analysis of masonry panels using a kinematic enriched plane state formulation. *International Journal of Solids and Structures*, 90, pp. 194-214. Doi: 10.1016/j.ijsolstr.2016.03.002

Ademovic, N. and Hadzima-Nyarko, M. (2019). Different possibilities for modelling cracked masonry structures. *Advanced Technologies, Systems, and Applications III*. Sarajevo, Bosnia and Herzegovina, June 20–23, 2019. Springer International Publishing, pp. 113-120

Agisoft (2019) *Metashape Professional version 1.5.1* [Computer program]. Available at: https://www.agisoft.com/buy/online-store/ [Accessed 19th January 2020]

Aguilar, F., Agüera, F., M.A, A. and Carvajal, F. (2005). Effects of terrain morphology, sampling density, and interpolation methods on grid dem accuracy. *Photogrammetric Engineering and Remote Sensing*, 71, pp. 805-816. Doi: 10.14358/PERS.71.7.805

Ali, S.S. and Page, A.W. (1988). Finite-element model for masonry subjected to concentrated loads. *Journal of Structural Engineering-Asce*, 114(8), pp. 1761-1784. Doi: 10.1061/(ASCE)0733-9445(1988)114:8(1761)

Altuntas, C., Pehlivanlı, M.E. and Kurban, S. (2017). Low-cost 3D imaging and measurement techniques for documentation of Meram masonry arch bridge in Turkey. *International Journal of Sensors and Sensor Networks Journal of Sensors and Sensor Networks*, 5, pp. 63-69. Doi: 10.11648/j.ijssn.20170505.11

Anthoine, A. (1995). Derivation of the in-plane elastic characteristics of masonry through homogenization theory. *International Journal of Solids and Structures*, 32(2), pp. 137-163. Doi: 10.1016/0020-7683(94)00140-R

Arias, P., Armesto, J., Di-Capua, D., Gonzalez-Drigo, R., Lorenzo, H. and Perez-Gracia, V. (2007). Digital photogrammetry, GPR and computational analysis of structural damages in a mediaeval bridge. *Engineering Failure Analysis*, 14(8), pp. 1444-1457. Doi: 10.1016/j.engfailanal.2007.02.001

Arias, P., Riveiro, B., Armesto, J. and Solla, M. (2010). Terrestrial laser scanning and non parametric methods in masonry arches inspection. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*. Newcastle upon Tyne, Uk. pp. 39-44

Asteris, P., Sarhosis, V., Mohebkhah, A., Plevris, V., Papaloizou, L., Komodromos, P. and Lemos, J. (2015).Numerical modeling of historic masonry structures, in: Panagiotis G. Asteris, V.P. (ed.) *Handbook of Research on Seismic Assessment and Rehabilitation of Historic Structures*. 1st edn., pp. 213-256.

Autodesk (2019) *Autocad 2019* (Version 23.0) [Computer program]. Available at: https://www.autodesk.com/ [Accessed 13th July 2019]

Baggio, C. and Trovalusci, P. (1998). Limit analysis for no-tension and frictional three-dimensional discrete systems. *Mechanics of Structures and Machines*, 26(3), pp. 287-304. Doi: 10.1080/08905459708945496

Baggio, C. and Trovalusci, P. (2000). Collapse behaviour of three-dimensional brick-block systems using non-linear programming. *Structural Engineering and Mechanics*, 10, pp. 181-195. Doi: 10.12989/sem.2000.10.2.181

Baltsavias, E.P. (1999). Airborne laser scanning: basic relations and formulas. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2-3), pp. 199-214. Doi: 10.1016/S0924-2716(99)00015-5

Baraldi, D., Reccia, E. and Cecchi, A. (2017). In plane loaded masonry walls: DEM and FEM/DEM models. A critical review. *Meccanica*, pp. 1-16. Doi: 10.1007/s11012-017-0704-3

Barazzetti, L., Banfi, F., Brumana, R., Gusmeroli, G., Previtali, M. and Schiantarelli, G. (2015a). Cloud-to-BIM-to-FEM: Structural simulation with accurate historic BIM from laser scans. *Simulation Modelling Practice and Theory*, 57, pp. 71-87. Doi: 10.1016/j.simpat.2015.06.004

Barazzetti, L., Giussani, A., Previtali, M. and Roncoroni, F. (2015b). Laser tracker technology for static monitoring of civil infrastructure. *Lasers in Engineering*, 32(3-4), pp. 263-294. Available at: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84941626306&partnerID=40&md5=b1a5d0815dd13bb5615f49d7bf8c8b03 [Accessed 13th July 2019]

Barrile, V., Bilotta, G., Lamari, D. and Meduri, G. (2015). Comparison between techniques for generating 3D models of cultural heritage. *Recent Advances in Mechanics, Mechatronics and Civil, Chemical and Industrial Engineering*. Zakynthos Island, Greece, July 16-20, 2015. Proceedings of the 2015 International Conference on Civil Engineering (CIVILENG 2015), pp. 140-145

Barrile, V., Bilotta, G., Meduri, G., Trovato, S. and D'Amore, E. (2016). Structural modeling of a historic castle using close range photogrammetry. *International Journal of Mathematics and Computers in Simulation*, 10, pp. 370-380. Available at: https://pdfs.semanticscholar.org/8ec1/34a69f0639928f51b0d71160983c17454349.pdf [Accessed 11th February 2020]

Barrile, V., Bilotta, G. and Nunnari, A. (2017). UAV and computer vision, detection of infrastructure losses and 3D modeling. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*. Safranbolu, Karabuk, Turkey, 11/13. pp. 135-139

Barrile, V., Candela, G. and Fotia, A. (2019). Point cloud segmentation using image processing techniques for structural analysis. *ISPRS - International Archives of the Photogrammetry, Remote Sensing And Spatial Information Sciences*. Milan, Italy, 05/04. pp. 187-193

Bassier, M., Hardy, G., Bejarano-Urrego, L., Drougkas, A., Verstrynge, E., Van Balen, K. and Vergauwen, M. (2018). Semi-automated creation of accurate FE meshes of heritage masonry walls from point cloud data. *Structural Analysis of Historical Constructions*. Cusco, Peru, 2018. Springer International Publishing, pp. 305-314

Beatini, V., Royer-Carfagni, G. and Tasora, A. (2019). A non-smooth-contact-dynamics analysis of Brunelleschi's cupola: an octagonal vault or a circular dome? *Meccanica*, 54(3), pp. 525-547. Doi: 10.1007/s11012-018-00934-9

Bentley (2020) *ContextCapture* [Computer program]. Available at: https://www.bentley.com/en/products/brands/contextcapture [Accessed 19th January 2020]

Berto, L., Saetta, A., Scotta, R. and Vitaliani, R. (2002). An orthotropic damage model for masonry structures. *International Journal for Numerical Methods in Engineering*, 55(2), pp. 127-157. Doi: 10.1002/nme.495

Binda, L., Cardani, G. and Saisi, A. (2009).A classification of structures and masonries for the adequate choice of repair, in: Groot, C. (ed.) *Workshop Repair Mortars for Historic Masonry*. 1st edn. Delft University of Technology, the Netherlands, pp. 20-34.

Bitelli, G., Castellazzi, G., D'Altri, A., Miranda, S., Lambertini, A. and Selvaggi, I. (2016). Automated voxel model from point clouds for structural analysis of cultural heritage. *ISPRS XL1 - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Prague, Czech republic, 06/16. pp. 191-197

Block, P. (2005) *Equilibrium systems: studies in masonry structure*. M.s. thesis. Massachusetts Institute of Technology [Accessed 5th December 2019]

Bobet, A., Fakhimi, A., Johnson, S., Morris, J., Tonon, F. and Yeung, M. (2009). Numerical models in discontinuous media: Review of advances for rock mechanics applications. *Journal of Geotechnical and Geoenvironmental Engineering*, 135, pp. 1547-1561. Doi: 10.1061/(ASCE)GT.1943-5606.0000133

Bosché, F., Forster, A. and Valero, E. (2015) *3D surveying technologies and applications: Point clouds and beyond*. Heriot-Watt University. [Online]. Available at: https://pureapps2.hw.ac.uk/ws/portalfiles/portal/9482712/report.pdf [Accessed 19th January 2020]

Brenner, C. (2005). Building reconstruction from images and laser scanning. *International Journal of Applied Earth Observation and Geoinformation*, 6(3), pp. 187-198. Doi: 10.1016/j.jag.2004.10.006

Bruno, N., Coisson, E., Diotri, F., Ferrari, L., Mikolajewska, S., Morra di Cella, U., Roncella, R. and Zerbi, A. (2019). History, geometry, structure: Interdisciplinary analysis of a historical bridge. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W11, pp. 317-323. Doi: 10.5194/isprs-archives-XLII-2-W11-317-2019

Bui, T.T., Limam, A., Sarhosis, V. and Hjiaj, M. (2017). Discrete element modelling of the in-plane and out-of-plane behaviour of dry-joint masonry wall constructions. *Engineering Structures*, 136, pp. 277-294. Doi: 10.1016/j.engstruct.2017.01.020

Çaktı, E., Saygılı, Ö., Lemos, J.V. and Oliveira, C.S. (2016). Discrete element modeling of a scaled masonry structure and its validation. *Engineering Structures*, 126, pp. 224-236. Doi: 10.1016/j.engstruct.2016.07.044

Castellazzi, G., Altri, M.A., Bitelli, G., Selvaggi, I. and Lambertini, A. (2015). From laser scanning to finite element analysis of complex buildings by using a semi-automatic procedure. *Sensors*, 15(8), pp. 18360–18380. Doi: 10.3390/s150818360

Castellazzi, G., D'Altri, A.M., de Miranda, S. and Ubertini, F. (2017). An innovative numerical modeling strategy for the structural analysis of historical monumental buildings. *Engineering Structures*, 132, pp. 229-248. Doi: 10.1016/j.engstruct.2016.11.032

Cavalagli, N., Cluni, F. and Gusella, V. (2013). Evaluation of a Statistically Equivalent Periodic Unit Cell for a quasi-periodic masonry. *International Journal of Solids and Structures*, 50(25), pp. 4226-4240. Doi: 10.1016/j.ijsolstr.2013.08.027

Cavicchi, A. and Gambarotta, L. (2006). Two-dimensional finite element upper bound limit analysis of masonry bridges. *Computers & Structures*, 84(31), pp. 2316-2328. Doi: 10.1016/j.compstruc.2006.08.048

Cecchi, A. and Milani, G. (2008). A kinematic FE limit analysis model for thick English bond masonry walls. *International Journal of Solids and Structures*, 45(5), pp. 1302-1331. Doi: 10.1016/j.ijsolstr.2007.09.019

Cecchi, A., Milani, G. and Tralli, A. (2007). A Reissner–Mindlin limit analysis model for out-of-plane loaded running bond masonry walls. *International Journal of Solids and Structures*, 44(5), pp. 1438-1460. Doi: 10.1016/j.ijsolstr.2006.06.033

Chen, S. (2012).Laser scanning technology for bridge monitoring, in: *Laser Scanner Technology*.

Chen, S., Laefer, D., Mangina, E., Zolanvari, I. and Byrne, J. (2019). UAV bridge inspection through evaluated 3D reconstructions. *Journal of Bridge Engineering*, 24, p. 05019001. Doi: 10.1061/(ASCE)BE.1943-5592.0001343

Chetouane, B., Dubois, F., Vinches, M. and Bohatier, C. (2005). NSCD discrete element method for modelling masonry structures. *International Journal for Numerical Methods in Engineering*, 64(1), pp. 65-94. Available at: <Go to ISI>://WOS:000231641100004 [Accessed 11th February 2020]

Chiabrando, F., Turco, M. and Rinaudo, F. (2017). Modeling the decay in an HBIM starting from 3D point clouds. A followed approach for cultural heritage knowledge. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Ottawa, Canada, 08/21. pp. 605-612

Chisari, C., Macorini, L., Amadio, C. and Izzuddin, B. (2018). Identification of mesoscale model parameters for brick-masonry. *International Journal of Solids and Structures*, 146, pp. 224-240. Doi: 10.1016/j.ijsolstr.2018.04.003

CloudCompare (2019) *CloudCompare (version 2.1), GPL software* [Computer program]. Available at: http://www.cloudcompare.org/ [Accessed 19th January 2020]

Clough, R.W.P., Joseph (1993) *Dynamics of structures*. 2nd edn. New York: McGraw-Hill Inc.,US.

Como, M. (2013) *Statics of Historic Masonry Constructions*. 1st edn. Springer Berlin Heidelberg.

Conde, B., F. Ramos, L., Oliveira, D., Riveiro, B. and Solla, M. (2017). Structural assessment of masonry arch bridges by combination of non-destructive testing techniques and three-dimensional numerical modelling: Application to Vilanova bridge. *Engineering Structures*, 148, pp. 621-638. Doi: 10.1016/j.engstruct.2017.07.011

Cundall, P.A. (1971). A computer model for simulating progressive, large - scale movement in blocky rock systems. *Proceedings of the Symposium of the International Society of Rock Mechanics*. Nancy, France, 1971. pp. 129-136

Cundall, P.A. (1987). Distinct element models of rock and soil structure. *Analytical and Computational Methods in Engineering Rock Mechanics*, pp. 129-163. Available at: https://ci.nii.ac.jp/naid/10003877143/en/ [Accessed 11th February 2020]

Cundall, P.A. and Hart, R.D. (1992). Numerical modelling of discontinua. *Engineering Computations*, 9(2), pp. 101-113. Doi: 10.1108/eb023851

Cundall, P.A. and Strack, O.D.L. (1980). A discrete numerical-model for granular assemblies - reply. *Geotechnique*, 30(3), pp. 335-336. Available at: <Go to ISI>://WOS:A1980KM60500017 [Accessed 11th February 2020]

D'Altri, A.M., de Miranda, S., Castellazzi, G. and Sarhosis, V. (2018a). A 3D detailed micro-model for the in-plane and out-of-plane numerical analysis of masonry panels. *Computers & Structures*, 206, pp. 18-30. Doi: 10.1016/j.compstruc.2018.06.007

D'Altri, A.M., Messali, F., Rots, J., Castellazzi, G. and de Miranda, S. (2019). A damaging block-based model for the analysis of the cyclic behaviour of full-scale masonry structures. *Engineering Fracture Mechanics*, 209, pp. 423-448. Doi: 10.1016/j.engfracmech.2018.11.046

D'Altri, A.M., Milani, G., Miranda, S., Castellazzi, G. and Sarhosis, V. (2018b). Stability analysis of leaning historic masonry structures. *Automation in Construction*, 92, pp. 199–213. Doi: 10.1016/j.autcon.2018.04.003

D’Altri, A.M., Sarhosis, V., Milani, G., Rots, J., Cattari, S., Lagomarsino, S., Sacco, E., Tralli, A., Castellazzi, G. and de Miranda, S. (2019). Modeling strategies for the computational analysis of unreinforced masonry structures: Review and classification. *Archives of Computational Methods in Engineering*, pp. 1-33. Doi: 10.1007/s11831-019-09351-x

Dai, F. and Lu, M. (2010). Assessing the accuracy of applying photogrammetry to take geometric measurements on building products *Journal of Construction Engineering and Management*, 136(2), pp. 242-250. Doi: 10.1061/(ASCE)CO.1943-7862.0000114

de Felice, G. (2011). Out-of-plane seismic capacity of masonry depending on wall section morphology *International Journal of Architectural Heritage*, 5(4-5), pp. 466-482. Doi: 10.1080/15583058.2010.530339

Dejong, M. (2009) *Seismic assessment strategies for masonry structures*. Ph.D. thesis. Massachusetts Institute of Technology [Online]. Available at: https://dspace.mit.edu/handle/1721.1/49538 [Accessed 19th January 2020]

Del Piero, G. (1989). Constitutive equation and compatibility of the external loads for linear elastic masonry-like materials. *Meccanica*, 24(3), pp. 150-162. Doi: 10.1007/BF01559418

Deseilligny, M. and Clery, I. (2011). APERO, an open-source bundle adjustment software for automatic calibration and orientation of set of images. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Trento, Italy, 2-4 March 2011. pp. 1-8

Dialer, C. (1992). A distinct element approach for the deformation behaviour of shear stressed masonry panels. *Proceedings of the 6th Canadian masonry symposium*. Saskatchewan, Canada, 15-17 June. pp. 765-776

Dietrich, J. (2016). Riverscape mapping with helicopter-based structure-from-motion photogrammetry. *Geomorphology*, 252, pp. 144–157. Doi: 10.1016/j.geomorph.2015.05.008

Doherty, K., Griffith, M., Lam, N. and Wilson, J. (2002). Displacement-based seismic analysis for out-of-plane bending of unreinforced masonry walls. *Earthquake Engineering & Structural Dynamics*, 31, pp. 833-850. Doi: 10.1002/eqe.126

Dubois, F. and Jean, M. (2003). LMGC90 une plateforme de développement dédiée à la modélisation des problèmes d’interaction. *Actes du Sixieme Colloque National en Calcul des Structures*. France, france. pp. 111-118

Eltner, A., Kaiser, A., Castillo, C., Rock, G., Neugirg, F. and Abellan, A. (2016). Image-based surface reconstruction in geomorphometry - merits, limits and developments. *Earth Surface Dynamics*, 4, pp. 359-389. Doi: 10.5194/esurf-4-359-2016

Eltner, A. and Schneider, D. (2015). Analysis of different methods for 3d reconstruction of natural surfaces from parallel-axes UAV images. *The Photogrammetric Record*, 30, pp. 279-299. Doi: 10.1111/phor.12115

ENSG, I.a. (2020) *MicMac* [Computer program]. Available at: https://micmac.ensg.eu/index.php/Accueil [Accessed 19th January 2020]

Estler, W.T., Edmundson, K.L., Peggs, G.N. and Parker, D.H. (2002). Large-scale metrology - An update. *CIRP Annals - Manufacturing Technology*, 51(2), pp. 587-609. Doi: 10.1016/S0007-8506(07)61702-8

Ferris, M.C. and Tin-Loi, F. (2001). Limit analysis of frictional block assemblies as a mathematical program with complementarity constraints. *International Journal of Mechanical Sciences*, 43(1), pp. 209-224. Doi: 10.1016/S0020-7403(99)00111-3

Fishwick, R.J. (1996) *Limit analysis of rigid block structures*. Ph.D. thesis. University of Portsmouth [Online]. Available at: https://ethos.bl.uk/OrderDetails.do?uin=uk.bl.ethos.310412 [Accessed 5th December 2019]

Fonstad, M., Dietrich, J., Courville, B., Jensen, J. and Carbonneau, P. (2013). Topographic structure from motion: a new development in photogrammetric measurement. *Earth Surface Processes and Landforms*, 38, pp. 421-430. Doi: 10.1002/esp.3366

Forgács, T., Rendes, S., Ádány, S. and Sarhosis, V. (2019). Mechanical role of spandrel walls on the capacity of masonry arch bridges. *9th International Conference on Arch Bridges*. Cham, Switzerland, 2-4 October 2019. Springer International Publishing, pp. 221-229

Forgacs, T., Sarhosis, V. and Bagi, K. (2017). Minimum thickness of semi-circular skewed masonry arches. *Engineering Structures*, 140, pp. 317-336. Doi: 10.1016/j.engstruct.2017.02.036

Forgács, T., Sarhosis, V. and Bagi, K. (2018). Influence of construction method on the load bearing capacity of skew masonry arches. *Engineering Structures*, 168, pp. 612-627. Doi: 10.1016/j.engstruct.2018.05.005

Forster, A., Valero, E. and Bosché, F. (2019). Automated defect detection and classification in ashlar masonry walls using machine learning. *Automation in Construction*, 106, p. 102846. Doi: 10.1016/j.autcon.2019.102846

Furukawa, Y. and Ponce, J. (2007). Accurate, dense, and robust multi-view stereopsis. *IEEE Conference on Computer Vision and Pattern Recognition*. Minneapolis, MN, USA, 17-22 June 2007 pp. 1-10

Gambarotta, L. and Lagomarsino, S. (1997). Damage models for the seismic response of brick masonry shear walls. Part I: The mortar joint model and its applications. *Earthquake Engineering & Structural Dynamics*, 26, pp. 423-439. Doi: 10.1002/(SICI)1096-9845(199704)26:4<423::AID-EQE650>3.0.CO;2-#

Gáspár, O., Sipos, A.A. and Sajtos, I. (2018). Effect of stereotomy on the lower bound value of minimum thickness of semi-circular masonry arches. *International Journal of Architectural Heritage*, 12(6), pp. 899-921. Doi: 10.1080/15583058.2017.1422572

Goda, K., Kiyota, T., Pokhrel, R.M., Chiaro, G., Katagiri, T., Sharma, K. and Wilkinson, S. (2015). The 2015 Gorkha Nepal earthquake: Insights from earthquake damage survey. *Frontiers in Built Environment*, 1 Doi: 10.3389/fbuil.2015.00008

Godio, M. (2016) *Modelling of masonry walls under seismic loadings*. Ph.D. thesis. Université Paris-Est [Online]. Available at: https://pastel.archives-ouvertes.fr/tel-01305508 [Accessed 5th December 2019]

Godio, M., Stefanou, I. and Sab, K. (2018). Effects of the dilatancy of joints and of the size of the building blocks on the mechanical behavior of masonry structures. *Meccanica*, 53(7), pp. 1629-1643. Doi: 10.1007/s11012-017-0688-z

Godio, M., Stefanou, I., Sab, K., Sulem, J. and Sakji, S. (2017). A limit analysis approach based on Cosserat continuum for the evaluation of the in-plane strength of discrete media: Application to masonry. *European Journal of Mechanics - A/Solids*, 66, pp. 168-192. Doi: 10.1016/j.euromechsol.2017.06.011

Golparvar-Fard, M., Bohn, J., Teizer, J., Savarese, S. and Peña-Mora, F. (2011). Evaluation of image-based modeling and laser scanning accuracy for emerging automated performance monitoring techniques. *Automation in Construction*, 20(8), pp. 1143-1155. Doi: 10.1016/j.autcon.2011.04.016

Griffith, M., Vaculik, J., Lam, N., Wilson, J. and Lumantarna, E. (2007). Cyclic testing of unreinforced walls in two-way bending. *Earthquake Engineering & Structural Dynamics*, 36, pp. 801-821. Doi: 10.1002/eqe.654

Haala, N. and Rothermel, M. (2012). Dense multi-stereo matching for high quality digital elevation models. *Photogrammetrie - Fernerkundung - Geoinformation*, 2012, pp. 331-343. Doi: 10.1127/1432-8364/2012/0121

Hacıefendioğlu, K. and Maraş, E.E. (2016). Photogrammetry in documentation and ambient vibration test of historical masonry minarets. *Experimental Techniques*, 40(6), pp. 1527-1537. Doi: 10.1007/s40799-016-0137-2

Hambly, E.C. (1985). Soil buckling and the leaning instability of tall structures. *The Structural Engineer*, 63, pp. 77-85. Available at: https://www.istructe.org/sitefiles/handlers/DownloadFile.ashx?productId=8009 [Accessed 19th January 2020]

Harwin, S., Lucieer, A. and Osborn, J. (2015). The impact of the calibration method on the accuracy of point clouds derived using unmanned aerial vehicle multi-view stereopsis. *Remote Sensing*, 7, pp. 11933-11953. Doi: 10.3390/rs70911933

Hendry, E. (2001). Masonry walls: materials and construction. *Construction and Building Materials*, 15, pp. 323-330. Doi: 10.1016/S0950-0618(01)00019-8

Heyman, J. (1969). The safety of masonry arches. *International Journal of Mechanical Sciences*, 11(4), pp. 363-385. Doi: 10.1016/0020-7403(69)90070-8

Heyman, J. (1992). Leaning towers. *Meccanica*, 27(3), pp. 153-159. Doi: 10.1007/BF00430041

Heyman, J. (1997) *The stone skeleton: Structural engineering of masonry architecture*. 2nd (reprint edition) edn. Cambridge University Press.

Hinks, T. (2011) *Geometric processing techniques for urban aerial laser scan data*. Ph.D. thesis. University College Dublin [Online]. Available at:

https://tommyhinks.files.wordpress.com/2012/02/thinks-thesis-single.pdf [Accessed 5th December 2019]

Hinks, T., Carr, H., Linh, T.H. and Laefer, D.F. (2012). Point cloud data conversion into solid models via point-based voxelization. *Journal of Surveying Engineering*, 139(2), pp. 72-83. Doi: 10.1061/(Asce)Su.1943-5428.0000097

Hirschmüller, H. (2008). Hirschmeller, H.: Stereo processing by semiglobal matching and mutual information. IEEE PAMI 30(2), 328-341. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30, pp. 328-41. Doi: 10.1109/TPAMI.2007.1166

Historic England (2017) *Photogrammetric applications for cultural heritage. Guidance for good practise.* 1st edn. Swindon. Historic England.

Hofstetter, G., Feist, C., Lehar, H., Theiner, Y., Valentini, B. and Winkler, B. (2011).Plasticity based crack models and applications, in: Hofstetter, G. and Meschke, G. (eds.) *Numerical Modeling of Concrete Cracking*. Vienna: Springer Vienna, pp. 161-219.

Housner, G.W. (1963). The behavior of inverted pendulum structures during earthquakes. *Bulletin of the Seismological Society of America*, 2(53), pp. 403-417. Doi: 10.1017/CBO9781107415324.004

Iman Zolanvari, S.M. and Laefer, D.F. (2016). Slicing method for curved façade and window extraction from point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 119, pp. 334-346. Doi: 10.1016/j.isprsjprs.2016.06.011

Itasca (2019a) *ITASCA. 3DEC 5.2 – Universal distinct element code manual. Theory and background. Mineapolis.* [Computer program]. Available at: www.itasca.com [Accessed 19th January 2020]

Itasca (2019b) *ITASCA. PFC –Particle flow code manual. Theory and background. Mineapolis.* [Computer program]. Available at: www.itasca.com [Accessed 19th January 2020]

Itasca (2019c) *ITASCA. UDEC – Universal distinct element code manual. Theory and background. Mineapolis.* [Computer program]. Available at: www.itasca.com [Accessed 19th January 2020]

James, M. and Robson, S. (2012). Straightforward reconstruction of 3D surfaces and topography with a camera: Accuracy and geoscience application. *Journal of Geophysical Research*, 117, p. F03017. Doi: 10.1029/2011JF002289

James, M.R., Robson, S., d'Oleire-Oltmanns, S. and Niethammer, U. (2017). Optimising UAV topographic surveys processed with structure-from-motion: Ground control quality, quantity and bundle adjustment. *Geomorphology*, 280, pp. 51-66. Doi: 10.1016/j.geomorph.2016.11.021

Jiang, K. and Esaki, T. (2002). Quantitative evaluation of stability changes in historical stone bridges in Kagoshima, Japan, by weathering. *Engineering Geology*, 63, pp. 83-91. Doi: 10.1016/S0013-7952(01)00071-0

Jing, L. (2003). A review of techniques, advances and outstanding issues in numerical modelling for rock mechanics and rock engineering. *International Journal of Rock Mechanics and Mining Sciences*, 40(3), pp. 283-353. Doi: 10.1016/S1365-1609(03)00013-3

Jirásek, M. (2011).Damage and smeared crack models, in: Hofstetter, G. and Meschke, G. (eds.) *Numerical Modeling of Concrete Cracking*. Vienna: Springer Vienna, pp. 1-49.

Kazhdan, M. and Hoppe, H. (2013). Screened poisson surface reconstruction. *ACM Transactions on Graphics*, 32(3), pp. 1-13. Doi: 10.1145/2487228.2487237

Kim, M.K., Wang, Q. and Li, H. (2019). Non-contact sensing based geometric quality assessment of buildings and civil structures: A review. *Automation in Construction*, 100, pp. 163-179. Doi: 10.1016/j.autcon.2019.01.002

Korumaz, M., Betti, M., Conti, A., Tucci, G., Bartoli, G., Bonora, V., Korumaz, A.G. and Fiorini, L. (2017). An integrated Terrestrial Laser Scanner (TLS), Deviation Analysis (DA) and Finite Element (FE) approach for health assessment of historical structures. A minaret case study. *Engineering Structures*, 153, pp. 224-238. Doi: 10.1016/j.engstruct.2017.10.026

Leica (2019) *Leica Cyclone ® version 9.1 P40 terrestrial laser scanner factory software* [Computer program]. Available at: https://leica-geosystems.com/blog-content/2014/leica-cyclone-9 [Accessed 19th January 2020]

Lemos, J. (1995).Assessment of the ultimate load of a masonry arch using discrete elements, in: *Journal of Computer Methods in Structural Masonry*. Swansea, UK: Books and Journals International, pp. 294-302.

Lemos, J.V. (2007). Discrete element modeling of masonry structures. *International Journal of Architectural Heritage*, 1(2), pp. 190-213. Doi: 10.1080/15583050601176868

Leonetti, L., Greco, F., Trovalusci, P., Luciano, R. and Masiani, R. (2018). A multiscale damage analysis of periodic composites using a couple-stress/Cauchy multidomain model: Application to masonry structures. *Composites Part B: Engineering*, 141, pp. 50-59. Doi: 10.1016/j.compositesb.2017.12.025

LimitState (2019) *Ring ® Version 3.0* [Computer program]. Available at: http://limitstate.com/ring [Accessed 19th January 2020]

Linh, T.H. and Laefer, D.F. (2013). Validating computational models from laser scanning data for historic facades. *Journal of Testing and Evaluation*, 41(3), pp. 481-496. Doi: 10.1520/Jte20120243

Linh, T.H. and Laefer, D.F. (2014). Octree-based, automatic building facade generation from LiDAR data. *Computer-Aided Design*, 53, pp. 46-61. Doi: 10.1016/j.cad.2014.03.001

Linh, T.H., Laefer, D.F., Hinks, T. and Carr, H. (2012). Flying voxel method with delaunay triangulation criterion for facade/feature detection for computation. *Journal of Computing in Civil Engineering*, 26(6), pp. 691-707. Doi: 10.1061/(Asce)Cp.1943-5487.0000188

Lisjak, A. and Grasselli, G. (2014). A review of discrete modeling techniques for fracturing processes in discontinuous rock masses. *Journal of Rock Mechanics and Geotechnical Engineering*, 6(4), pp. 301-314. Doi: 10.1016/j.jrmge.2013.12.007

Liu, S.G., Li, Z.J., Zhang, H., Wu, W., Zhong, G.H. and Lou, S. (2018). A 3-D DDA damage analysis of brick masonry buildings under the impact of boulders in mountainous areas. *Journal of Mountain Science*, 15(3), pp. 657-671. Doi: 10.1007/s11629-017-4453-5

Livesley, R.K. (1978). Limit analysis of structures formed from rigid blocks. *International Journal for Numerical Methods in Engineering*, 12(12), pp. 1853-1871. Doi: 10.1002/nme.1620121207

Lopez, J., Oller, S., Oñate, E. and Lubliner, J. (1999). A homogeneous constitutive model for masonry. *International Journal for Numerical Methods in Engineering*, 46(10), pp. 1651-1671. Doi: 10.1002/(SICI)1097-0207(19991210)46:10<1651::AID-NME718>3.0.CO;2-2

Lotfi, H.R. and Shing, P.B. (1991). An appraisal of smeared crack models for masonry shear wall analysis. *Computers & Structures*, 41(3), pp. 413-425. Doi: 10.1016/0045-7949(91)90134-8

Lotfi, H.R. and Shing, P.B. (1994). Interface model applied to fracture of masonry structures. *Journal of Structural Engineering*, 120(1), pp. 63-80. Doi: 10.1061/(ASCE)0733-9445(1994)120:1(63)

Lourenco, P. (1996) *Computational strategies for masonry structures*. Ph.D. thesis. Delft University of Technology [Online]. Available at: http://www.hms.civil.uminho.pt/arq/fich/1996_PhD _PBLourenco.pdf [Accessed 5th December 2019]

Lourenco, P. (2002). Computations on historic masonry structures. *Progress in Structural Engineering and Materials*, 4, pp. 301-319. Doi: 10.1002/pse.120

Lourenco, P.B. and Rots, J.G. (1997). Multisurface interface model for analysis of masonry structures. *Journal of Engineering Mechanics-Asce*, 123(7), pp. 660-668. Doi: 10.1061/(ASCE)0733-9399(1997)123:7(660)

Lowe, D.G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), pp. 91-110. Doi: 10.1023/B:VISI.0000029664.99615.94

Lubowiecka, I., Arias, P., Riveiro, B. and Solla, M. (2011). Multidisciplinary approach to the assessment of historic structures based on the case of a masonry bridge in Galicia (Spain). *Computers & Structures*, 89, pp. 1615-1627. Doi: 10.1016/j.compstruc.2011.04.016

Macorini, L. and Izzuddin, B.A. (2011). A non-linear interface element for 3D mesoscale analysis of brick-masonry structures. *International Journal for Numerical Methods in Engineering*, 85(12), pp. 1584-1608. Doi: 10.1002/nme.3046

Makris, N. and Alexakis, H. (2013). The effect of stereotomy on the shape of the thrust-line and the minimum thickness of semicircular masonry arches. *Archive of Applied Mechanics*, 83(10), pp. 1511-1533. Doi: 10.1007/s00419-013-0763-4

Malomo, D., DeJong, M.J. and Penna, A. (2019). Distinct element modelling of the in-plane cyclic response of URM walls subjected to shear-compression. *Earthquake Engineering & Structural Dynamics*, 48(12), pp. 1322-1344. Doi: 10.1002/eqe.3178

Marchi, M., Butterfield, R., Gottardi, G. and Lancellotta, R. (2011). Stability and strength analysis of leaning towers. *Géotechnique*, 61(12), pp. 1069-1079. Doi: 10.1680/geot.9.P.054

Mathworks (2019) *Matlab* (Version 9.6.0.1174912 (R2019a) Update 5) [Computer program]. Available at: https://www.mathworks.com/products/matlab.html [Accessed 19th January 2020]

Meschini, A., Leoni, G., Petrucci, E., Sicuranza, F., Zona, A., Piattoni, Q., Dezi, L. and Dall'Asta, A. (2015). An integrated Survey Experience for Assessing the Seismic Vulnerability of Senigallia’s Fortress (Italy): Documentation for Conservation and FEM Modeling. *Digital Heritage 2015*. Granada, Spain, 28 Sept.-2 Oct. 2015. pp. 21-28

Milani, G. (2008). 3D upper bound limit analysis of multi-leaf masonry walls. *International Journal of Mechanical Sciences*, 50(4), pp. 817-836. Doi: 10.1016/j.ijmecsci.2007.11.003

Milani, G. (2011). Simple lower bound limit analysis homogenization model for in- and out-of-plane loaded masonry walls. *Construction and Building Materials*, 25(12), pp. 4426-4443. Doi: 10.1016/j.conbuildmat.2011.01.012

Milani, G., Beyer, K. and Dazio, A. (2009). Upper bound limit analysis of meso-mechanical spandrel models for the pushover analysis of 2D masonry frames. *Engineering Structures*, 31(11), pp. 2696-2710. Doi: 10.1016/j.engstruct.2009.06.015

Milani, G., Casolo, S., Naliato, A. and Tralli, A. (2012). Seismic assessment of a medieval masonry tower in northern Italy by limit, nonlinear static, and full dynamic analyse. *International Journal of Architectural Heritage*, 6(5), pp. 489-524. Doi: 10.1080/15583058.2011.588987

Milani, G., Lourenço, P. and Tralli, A. (2006a). Homogenization approach for the limit analysis of out-of-plane loaded masonry walls. *Journal of Structural Engineering*, 132(10), pp. 1650-1663. Doi: 10.1061/(ASCE)0733-9445(2006)132:10(1650)

Milani, G., Lourenço, P.B. and Tralli, A. (2006b). Homogenised limit analysis of masonry walls, Part I: Failure surfaces. *Computers & Structures*, 84(3), pp. 166-180. Doi: 10.1016/j.compstruc.2005.09.005

Milani, G., Shehu, R. and Valente, M. (2017). Role of inclination in the seismic vulnerability of bell towers: FE models and simplified approaches. *Bulletin of Earthquake Engineering*, 15(4), pp. 1707-1737. Doi: 10.1007/s10518-016-0043-0

Mills, J. and Barber, D. (2004). Geomatics techniques for structural surveying. *Journal of Surveying Engineering*, 57, pp. 56-64 Doi: 10.1061/(ASCE)0733-9453(2004)130:2(56)

Moreau, J.J. (1988). Free boundaries and non-smooth solutions to some field-equations - variational characterization through the transport method. *Lecture Notes in Control and Information Sciences*, 100, pp. 235-264. Available at: <Go to ISI>://WOS:A1988M644400013 [Accessed 23th April 2020]

Morer, P., de Arteaga, I., Armesto, J. and Arias, P. (2011). Comparative structural analyses of masonry bridges: An application to the Cernadela Bridge. *Journal of Cultural Heritage*, 12(3), pp. 300-309. Doi: 10.1016/j.culher.2011.01.006

Morer, P., de Arteaga, I. and Ortueta, A. (2013). A low-cost photogrammetric methodology to obtain geometrical data of masonry arch bridges. *Journal of Architectural Conservation*, 19(3), pp. 246-264. Doi: 10.1080/13556207.2013.869974

Munjiza, A., Owen, D.R.J. and Bicanic, N. (1995). A combined finite-discrete element method in transient dynamics of fracturing solids. *Engineering Computations*, 12(2), pp. 145-174. Doi: 10.1108/02644409510799532

Napolitano, R. and Glisic, B. (2019). Understanding the function of bonding courses in masonry construction: An investigation with mixed numerical methods. *Journal of Cultural Heritage*, 39, pp. 120-129. Doi: 10.1016/j.culher.2019.03.007

Napolitano, R., Hess, M., Coe-Scharff, R. and Glisic, B. (2019a).Numerical modeling of crack propagation in masonry structures, in: Aguilar, R., Torrealva, D., Moreira, S., Pando, M.A. and Ramos, L.F. (eds.) *Structural Analysis of Historical Constructions*. Cham, Switzerland: Springer International Publishing, pp. 826-834.

Napolitano, R., Hess, M. and Glisic, B. (2019b). Quantifying the differences in documentation and modeling levels for building pathology and diagnostics. *Archives of Computational Methods in Engineering*, pp. 1-18. Doi: 10.1007/s11831-019-09350-y

Napolitano, R.K., Hess, M. and Glisic, B. (2019c). The foundation walls of the baptistery Di San Giovanni: A combination of laser scanning and finite-distinct element modeling to ascertain damage origins. *International Journal of Architectural Heritage*, 13(7), pp. 1180-1193. Doi: 10.1080/15583058.2019.1582726

Nikolić, D. (2017). Thrust line analysis and the minimum thickness of pointed masonry arches. *Acta Mechanica*, 228(6), pp. 2219-2236. Doi: 10.1007/s00707-017-1823-6

Olofsson, I., Elfgren, L., Bell, B., Paulsson, B., Niederleithinger, E., Sandager Jensen, J., Feltrin, G., Täljsten, B., Cremona, C., Kiviluoma, R. and Bien, J. (2005). Assessment of european railway bridges for future traffic demands and longer lives – ec project "sustainable bridges". *Structure and Infrastructure Engineering*, 1(2), pp. 93-100. Doi: 10.1080/15732470412331289396

Olsen, M. and Kayen, R. (2012). Post-earthquake and tsunami 3D laser scanning forensic investigations. *ASCE Sixth Congress on Forensic Engineering*. San Francisco, June 2012. pp. 477-486

Orduña, A. and Lourenço, P.B. (2005). Three-dimensional limit analysis of rigid blocks assemblages. Part I: Torsion failure on frictional interfaces and limit analysis formulation. *International Journal of Solids and Structures*, 42(18), pp. 5140-5160. Doi: 10.1016/j.ijsolstr.2005.02.010

Oses, N., Dornaika, F. and Moujahid, A. (2014). Image-based delineation and classification of built heritage masonry. *Remote Sensing*, 6(3), pp. 782-789. Doi: 10.3390/rs6031863

Otter, J.R.H., Cassell, A.C., Hobbs, R.E. and Poisson, J. (1966). Dynamic relaxation. *Proceedings of the Institution of Civil Engineers*, 35(4), pp. 633-656. Doi: 10.1680/iicep.1966.8604

Owen, R.D., Perić, D., Petrinic, N. and James, J.P. (1998) *Finite/discrete element models for assessment and repair of masonry structures*. [Online]. Available at: https://cintec.com/wp-content/uploads/2015/04/Finite_discrete-Elements.pdf [Accessed 19th January 2020]

Page, A.W. (1978). Finite-element model for masonry. *Journal of the Structural Division-Asce*, 104(8), pp. 1267-1285. Available at: <Go to ISI>://WOS:A1978FK58500007 [Accessed 23th April 2020]

Pelà, L., Cervera, M., Oller, S. and Chiumenti, M. (2014). A localized mapped damage model for orthotropic materials. *Engineering Fracture Mechanics*, 124-125, pp. 196-216. Doi: 10.1016/j.engfracmech.2014.04.027

Pelà, L., Cervera, M. and Roca, P. (2011). Continuum damage model for orthotropic materials: Application to masonry. *Computer Methods in Applied Mechanics and Engineering*, 200(9), pp. 917-930. Doi: 10.1016/j.cma.2010.11.010

Pelà, L., Cervera, M. and Roca, P. (2013). An orthotropic damage model for the analysis of masonry structures. *Construction and Building Materials*, 41, pp. 957-967. Doi: 10.1016/j.conbuildmat.2012.07.014

Pepe, M., Fregonese, I. and Crocetto, N. (2019). Application of the SfM approach in building 3D models of masonry bridges using nadir, oblique and zenith images obtained from UAV aerial platform. *IJCIET*, Volume 10, (3), pp. 1626 - 1638. Available at: http://d.researchbib.com/f/9nq3q3YzyuMJ1yYzAioF9ALKA0MKWOMT1cov9IpTkiLJETo2kxMKViFHcQFHIHKmRjKmNmKmR2ZF9WFxAWEIEsZGOsZQAsZGLkYaOxMt.pdf [Accessed 23th April 2020]

Peppa, M.V. (2018) *Morphology-based landslide monitoring with an unmanned aerial vehicle*. Ph.D. thesis. University of Newcastle [Accessed 5th December 2019]

Perez-Aparicio, J.L., Bravo, R. and Ortiz, P. (2013). Refined element discontinuous numerical analysis of dry-contact masonry arches. *Engineering Structures*, 48, pp. 578-587. Doi: 10.1016/j.engstruct.2012.09.027

Petracca, M., Pelà, L., Rossi, R., Zaghi, S., Camata, G. and Spacone, E. (2017). Micro-scale continuous and discrete numerical models for nonlinear analysis of masonry shear walls. *Construction and Building Materials*, 149, pp. 296–314. Doi: 10.1016/j.conbuildmat.2017.05.130

Petry, S. and Beyer, K. (2014). Scaling unreinforced masonry for reduced-scale seismic testing. *Bulletin of Earthquake Engineering*, 12(6), pp. 2557-2581. Doi: 10.1007/s10518-014-9605-1

PhotoModelerTechnologies (2019) *PhotoModeler Premium* (Version 2020.0.0) [Computer program]. Available at: https://www.photomodeler.com/resources/faq/# [Accessed 19th January 2020]

Pieraccini, M., Dei, D., Betti, M., Bartoli, G., Tucci, G. and Guardini, N. (2014). Dynamic identification of historic masonry towers through an expeditious and no-contact approach: Application to the "Torre del Mangia" in Siena (Italy). *Journal of Cultural Heritage*, 15(3), pp. 275-282. Doi: 10.1016/j.culher.2013.07.006

Pietruszczak, S. and Niu, X. (1992). A mathematical description of macroscopic behaviour of brick masonry. *International Journal of Solids and Structures*, 29(5), pp. 531-546. Doi: 10.1016/0020-7683(92)90052-U

Portioli, F., Casapulla, C., Gilbert, M. and Lucrezia, C. (2014). Limit analysis of 3D masonry block structures with non-associative frictional joints using cone programming. *Computers & Structures*, 143, pp. 108-121. Doi: 10.1016/j.compstruc.2014.07.010

Prizeman, O.E.C., Sarhosis, V., D'Alri, A.M., Whitman, C.J. and Muratore, G. (2017). Modelling from the past: The leaning southwest tower of caerphilly castle 1539-2015. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*. Ottawa, Canada. pp. 221-227

Psycharis, I.N., Fragiadakis, M. and Stefanou, I. (2013). Seismic reliability assessment of classical columns subjected to near-fault ground motions. *Earthquake Engineering & Structural Dynamics*, 42(14), pp. 2061-2079. Doi: 10.1002/eqe.2312

Psycharis, I.N., Lemos, J.V., Papastamatiou, D.Y., Zambas, C. and Papantonopoulos, C. (2003). Numerical study of the seismic behaviour of a part of the Parthenon Pronaos. *Earthquake Engineering & Structural Dynamics*, 32(13), pp. 2063-2084. Doi: 10.1002/eqe.315

Pulatsu, B., Bretas, E. and Lourenco, P. (2016). Discrete element modeling of masonry structures: Validation and application. *Earthquakes and Structures*, 11, pp. 563-582. Doi: 10.12989/eas.2016.11.4.563

Pulatsu, B., Erdogmus, E. and Bretas, E. (2018). Parametric study on masonry arches using 2d discrete-element modeling. *Journal of Architectural Engineering*, 24(2), p. 04018005 Doi: 10.1061/(Asce)Ae.1943-5568.0000305

Pulatsu, B., Erdogmus, E. and Lourenco, P.B. (2019a). Comparison of in-plane and out-of-plane failure modes of masonry arch bridges using discontinuum analysis. *Engineering Structures*, 178, pp. 24-36. Doi: 10.1016/j.engstruct.2018.10.016

Pulatsu, B., Erdogmus, E., Lourenco, P.B. and Quey, R. (2019b). Simulation of uniaxial tensile behavior of quasi-brittle materials using softening contact models in DEM. *International Journal of Fracture*, 217(1-2), pp. 105-125. Doi: 10.1007/s10704-019-00373-x

Pulatsu, B., Sarhosis, V., Bretas, E.M., Nikitas, N. and Lourenco, P.B. (2017). Non-linear static behaviour of ancient free-standing stone columns. *Proceedings of the Institution of Civil Engineers-Structures and Buildings*, 170(6), pp. 406-418. Doi: 10.1680/jstbu.16.00071

Radu Bogdan, R., Blodow, N., Marton, Z., Soos, A. and Beetz, M. (2007). Towards 3D object maps for autonomous household robots. *IEEE/RSJ International Conference on Intelligent Robots and Systems*. San Diego, California, USA, 29 Oct.-2 Nov. 2007. pp. 3191-3198

Rafiee, A., Vinches, M. and Bohatier, C. (2008). Application of the NSCD method to analyse the dynamic behaviour of stone arched structures. *International Journal of Solids and Structures*, 45(25-26), pp. 6269-6283. Doi: 10.1016/j.ijsolstr.2008.07.034

Remondino, F., Spera, M.G., Nocerino, E., Menna, F. and Nex, F. (2014). State of the art in high density image matching. *The Photogrammetric Record*, 29(146), pp. 144-166. Doi: 10.1111/phor.12063

Renn, D. (2002) *Caerphilly Castle, Cardiff, UK*. 1st edn. Cadw.

Reyes, E., Gálvez, J.C., Casati, M.J., Cendón, D.A., Sancho, J.M. and Planas, J. (2009). An embedded cohesive crack model for finite element analysis of brickwork masonry fracture. *Engineering Fracture Mechanics*, 76(12), pp. 1930-1944. Doi: 10.1016/j.engfracmech.2009.05.002

Riveiro, B., Caamaño, J.C., Arias, P. and Sanz, E. (2011). Photogrammetric 3D modelling and mechanical analysis of masonry arches: An approach based on a discontinuous model of voussoirs. *Automation in Construction*, 20(4), pp. 380-388. Doi: 10.1016/j.autcon.2010.11.008

Riveiro, B., Cabaleiro, M., Conde, B., Soilan, M. and Sanchez-Rodriguez, A. (2020).Laser scanning data for inverse problems in structural engineering, in: Riveiro, B. and Lindenbergh, R. (eds.) *Laser Scanning: An Emerging Technology in Structural Engineering*. pp. 215-225.

Riveiro, B., Lourenço, P.B., Oliveira, D.V., González-Jorge, H. and Arias, P. (2016). Automatic morphologic analysis of quasi-periodic masonry walls from LiDAR. *Computer-Aided Civil and Infrastructure Engineering*, 31(4), pp. 305-319. Doi: 10.1111/mice.12145

Riveiro, B., Solla, M., de Arteaga, I., Arias, P. and Morer, P. (2013). A novel approach to evaluate masonry arch stability on the basis of limit analysis theory and non-destructive geometric characterization. *Automation in Construction*, 31, pp. 140-148. Doi: 10.1016/j.autcon.2012.11.035

Roca, P., Cervera, M., Gariup, G. and Pelà, L. (2010). Structural analysis of masonry historical constructions. Classical and advanced approaches. *Archives of Computational Methods in Engineering*, 17, pp. 299-325. Doi: 10.1007/s11831-010-9046-1

Rolin, R., Antaluca, E., Batoz, J.L., Lamarque, F. and Lejeune, M. (2019). From point cloud data to structural analysis through a geometrical hbim-oriented model. *Journal of Computational Cultural Heritage*, 12(2), pp. 1-26. Doi: 10.1145/3242901

Rusinkiewicz, S. and Levoy, M. (2001). Efficient variants of the ICP algorithm. *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*. Quebec City, Canada, 28 May-1 June 2001. pp. 145-152

Sánchez-Aparicio, L., Villarino, A., García Gago, J.M. and González-Aguilera, D. (2016). Photogrammetric, geometrical, and numerical strategies to evaluate initial and current conditions in historical constructions: A test case in the church of san lorenzo (Zamora, Spain). *Remote Sensing*, 8(1), p. 60. Doi: 10.3390/rs8010060

Sánchez-Aparicio, L.J., Riveiro, B., González-Aguilera, D. and Ramos, L.F. (2014). The combination of geomatic approaches and operational modal analysis to improve calibration of finite element models: A case of study in Saint Torcato Church (Guimarães, Portugal). *Construction and Building Materials*, 70, pp. 118-129. Doi: 10.1016/j.conbuildmat.2014.07.106

Sarhosis, V., Asteris, P., Wang, T., Hu, W. and Han, Y. (2016a). On the stability of colonnade structural systems under static and dynamic loading conditions. *Bulletin of Earthquake Engineering*, 14(4), pp. 1131-1152. Doi: 10.1007/s10518-016-9881-z

Sarhosis, V., Asteris, P.G., Mohebkhah, A., Xiao, J. and Wang, T. (2016b). Three dimensional modelling of ancient colonnade structural systems subjected to harmonic and seismic loading. *Structural Engineering and Mechanics*, 60(4), pp. 633-653. Doi: 10.12989/sem.2016.60.4.633

Sarhosis, V., Bagi, K., Lemos, A.R. and Milani, G. (2016c) *Computational modeling of masonry structures using the discrete element method*. 1st edn. Hershey, PA, USA: IGI Global.

Sarhosis, V., Bagi, K., Lemos, J. and Milani, G. (eds.) (2016d) *Computational Modeling of Masonry Structures Using the Discrete Element Method*. Hershey, PA, USA: IGI Global.

Sarhosis, V., De Santis, S. and de Felice, G. (2016e). A review of experimental investigations and assessment methods for masonry arch bridges. *Structure and Infrastructure Engineering*, 12(11), pp. 1439-1464. Doi: 10.1080/15732479.2015.1136655

Sarhosis, V., Forgacs, T. and Lemos, J. (2020). Stochastic strength prediction of masonry structures: a methodological approach or a way forward? *RILEM Technical Letters*, 4(0), pp. 122-129. Doi: 10.21809/rilemtechlett.2019.100

Sarhosis, V., Forgács, T. and Lemos, J.V. (2019). Modelling backfill in masonry arch bridges: A DEM approach. *9th International Conference on Arch Bridges*. Cham, Switzerland, 2-4 October 2019. Springer International Publishing, pp. 178-184

Sarhosis, V., Garrity, S.W. and Sheng, Y. (2015). Influence of brick-mortar interface on the mechanical behaviour of low bond strength masonry brickwork lintels. *Engineering Structures*, 88, pp. 1-11. Doi: 10.1016/j.engstruct.2014.12.014

Sarhosis, V. and Lemos, J.V. (2018). A detailed micro-modelling approach for the structural analysis of masonry assemblages. *Computers & Structures*, 206, pp. 66-81. Doi: 10.1016/j.compstruc.2018.06.003

Sarhosis, V. and Sheng, Y. (2014). Identification of material parameters for low bond strength masonry. *Engineering Structures*, 60, pp. 100-110. Doi: 10.1016/j.engstruct.2013.12.013

Schonberger, J.L.F., Jan-Michael (2020) *Colmap* [Computer program]. Available at: https://colmap.github.io/ [Accessed 19th January 2020]

Selvaggi, I., Dellapasqua, M., Franci, F., Spangher, A., Visintini, D. and Bitelli, G. (2018). 3D comparison towards a comprehensive analysis of a building in cultural heritage. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Riva del Garda, Italy. Copernicus Publications, pp. 1061-1066

Serpieri, R., Albarella, M. and Sacco, E. (2017). A 3D microstructured cohesive–frictional interface model and its rational calibration for the analysis of masonry panels. *International Journal of Solids and Structures*, 122-123, pp. 110-127. Doi: 10.1016/j.ijsolstr.2017.06.006

Shen, Y., Lindenbergh, R. and Wang, J. (2016). Change analysis in structural laser scanning point clouds: The baseline method. *Sensors*, 17(1), p. 26. Doi: 10.3390/s17010026

Shen, Y.Q., Wang, J.G., Wei, S.F., Zheng, D.H. and Ferreira, V.G. (2019). Accurate extraction of brick shapes in masonry walls from dense terrestrial laser scanning point cloud. *Measurement*, 146, pp. 254-267. Doi: 10.1016/j.measurement.2019.05.086

Shi, G.H. and Goodman, R.E. (1985). Two-dimensional discontinuous deformation analysis. *International Journal for Numerical and Analytical Methods in Geomechanics*, 9(6), pp. 541-556. Doi: 10.1002/nag.1610090604

Shi, G.H. and Goodman, R.E. (1989). Generalization of two-dimensional discontinuous deformation analysis for forward modeling. *International Journal for Numerical and Analytical Methods in Geomechanics*, 13(4), pp. 359-380. Doi: 10.1002/nag.1610130403

Shrive, N. (2015). Discrete element modeling of stone masonry walls with varying core conditions: Prince of Wales fort case study. *International Journal of Architectural Heritage*, 9(5), pp. 564-580. Doi: 10.1080/15583058.2013.819135

Simulia Inc. (2017) *Abaqus ® Version 6.17* [Computer program]. Available at: https://www.3ds.com/products-services/simulia/ [Accessed 19th January 2020]

Sithole, G. (2008). Detection of bricks in a masonry wall. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. XXXVII*. Beijing. pp. 154-165

Šmilauer, V., Gladky, A., Kozicki, J., Modenese, C. and Stránský, J. (2010) *Yade, Using and Programming. In Yade Documentation* [Computer program]. Available at: http://yade-dem.org/doc [Accessed 19th January 2020]

Smoljanovic, H., Nikolic, Z. and Zivaljic, N. (2015). A finite-discrete element model for dry stone masonry structures strengthened with steel clamps and bolts. *Engineering Structures*, 90, pp. 117-129. Doi: 10.1016/j.engstruct.2015.02.004

Smoljanovic, H., Zivaljic, N. and Nikolic, Z. (2013a). A combined finite-discrete element analysis of dry stone masonry structures. *Engineering Structures*, 52, pp. 89-100. Doi: 10.1016/j.engstruct.2013.02.010

Smoljanovic, H., Zivaljic, N. and Nikolic, Z. (2013b). Overview of the methods for the modelling of historical masonry structures. *Gradevinar*, 65(7), pp. 603-618. Doi: 10.14256/JCE.890.2013

Smoljanovic, H., Zivaljic, N., Nikolic, Z. and Munjiza, A. (2018). Numerical analysis of 3D dry-stone masonry structures by combined finite-discrete element method. *International Journal of Solids and Structures*, 136, pp. 150-167. Doi: 10.1016/j.ijsolstr.2017.12.012

Snavely, K.N. (2008) *Scene reconstruction and visualization from internet photo collections*. Ph.D. thesis. University of Washington [Online]. Available at: https://www.cs.cornell.edu/~snavely/publications/thesis/thesis.pdf [Accessed 5th December 2019]

Solla, M., Caamaño, J.C., Riveiro, B. and Arias, P. (2012). A novel methodology for the structural assessment of stone arches based on geometric data by integration of photogrammetry and ground-penetrating radar. *Engineering Structures*, 35, pp. 296-306. Doi: 10.1016/j.engstruct.2011.11.004

Soni, A., Robson, S. and Gleeson, B. (2015). Structural monitoring for the rail industry using conventional survey, laser scanning and photogrammetry. *Applied Geomatics*, 7(2), pp. 123-138. Doi: 10.1007/s12518-015-0156-1

Stavroulaki, M.E., Riveiro, B., Drosopoulos, G.A., Solla, M., Koutsianitis, P. and Stavroulakis, G.E. (2016). Modelling and strength evaluation of masonry bridges using terrestrial photogrammetry and finite elements. *Advances in Engineering Software*, 101, pp. 136-148. Doi: 10.1016/j.advengsoft.2015.12.007

Stefanou, I., Fragiadakis, M. and Psycharis, I. (2015).Seismic reliability assessment of classical columns subjected to near source ground motions, in, pp. 61-82.

Stockdale, G., Sarhosis, V. and Milani, G. (2018). Increase in seismic resistance for a dry joint masonry arch subjected to hinge control. *10th International Masonry Conference*. Milan, Italy. pp. 968-981

Sutcliffe, D.J., Yu, H.S. and Page, A.W. (2001). Lower bound limit analysis of unreinforced masonry shear walls. *Computers & Structures*, 79(14), pp. 1295-1312. Doi: 10.1016/S0045-7949(01)00024-4

Szakály, F., Hortobágyi, Z. and Bagi, K. (2016). Discrete element analysis of the shear resistance of planar walls with different bond patterns. *The Open Construction and Building Technology Journal*, 10, pp. 220-232. Doi: 10.2174/1874836801610010220

Tang, P., Akinci, B. and Garrett, J. (2007) *Laser scanning for bridge inspection and management*. [Online]. Available at: www.doi.org/10.2749/222137807796120283 [Accessed 19th January 2020]

Tavafi, E., Mohebkhah, A. and Sarhosis, V. (2019). Seismic behavior of the cube of Zoroaster tower using the discrete element method. *International Journal of Architectural Heritage*, pp. 1-17. Doi: 10.1080/15583058.2019.1650135

Teza, G., Pesci, A. and Ninfo, A. (2016). Morphological analysis for architectural applications: Comparison between laser scanning and structure-from-motion photogrammetry. *Journal of Surveying Engineering*, 142(3), p. 04016004. Doi: 10.1061/(ASCE)SU.1943-5428.0000172

Thavalingam, A., Bicanic, N., Robinson, J.I. and Ponniah, D.A. (2001). Computational framework for discontinuous modelling of masonry arch bridges. *Computers & Structures*, 79(19), pp. 1821-1830. Doi: 10.1016/S0045-7949(01)00102-X

The highways agency (2001). *The assessment of highway bridges and structures - BA16/97*. [Online]. Available at: https://www.thenbs.com/PublicationIndex/documents/details?Pub=HA&DocID=254890# [Accessed]

TNO DIANA BV. Delft, N. (2020) *Diana ® Version 10.4* [Computer program]. Available at: https://dianafea.com/diana-downloads [Accessed 19th January 2020]

Tobiasz, A., Markiewicz, J., Lapinski, S., Nikel, J., Kot, P. and Muradov, M. (2019). Review of methods for documentation, management, and sustainability of cultural heritage. Case study: museum of king jan iii's palace at wilanów. *Sustainability (Switzerland)*, 11(24) Doi: 10.3390/su11247046

Truong-Hong, L. and Laefer, D.F. (2014). Octree-based, automatic building façade generation from LiDAR data. *Computer-Aided Design*, 53, pp. 46-61. Doi: 10.1016/j.cad.2014.03.001

Tsilimantou, E., Delegou, E., Ioannidis, C. and Moropoulou, A. (2016). Geoinformation techniques for the 3D visualisation of historic buildings and representation of a building's pathology. *4th International*

Conference on Remote Sensing and Geoinformation of The Environment. Paphos, Cyprus. SPIE, pp. 1-9

Tubaldi, E., Macorini, L. and Izzuddin, B.A. (2019). Identification of critical mechanical parameters for advanced analysis of masonry arch bridges. *Structure and Infrastructure Engineering*, pp. 328-345. Doi: 10.1080/15732479.2019.1655071

Tucci, G. and Guardini, N. (2014). Rilevo e modellazione 3D a supporto dell'analisi strutturale: un approccio metodologico e sostenibile per il patrimonio architettonico (Italian). *Conference: Atti 18 Conferenza ASITA*. Florence, Italy. pp. 1197-1209

Valero, E., Bosche, F. and Forster, A. (2018). Automatic segmentation of 3D point clouds of rubble masonry walls, and its application to building surveying, repair and maintenance. *Automation in Construction*, 96, pp. 29-39. Doi: 10.1016/j.autcon.2018.08.018

Van der Pluijm, R. (1999) *Out-of-plane Bending of Masonry: Behaviour and Strength*. 1st edn. Technische Universiteit Eindhoven.

Vanin, F., Zaganelli, D., Penna, A. and Beyer, K. (2017). Estimates for the stiffness, strength and drift capacity of stone masonry walls based on 123 quasi-static cyclic tests reported in the literature. *Bulletin of Earthquake Engineering*, 15(12), pp. 5435-5479. Doi: 10.1007/s10518-017-0188-5

Vatan, M. and Arun, G. (2005). Using photogrammetric data for establishing 3D finite element model of a masonry aqueduct. *CIPA 2005 XX International Symposium*. Torino, Italy, 26 September – 01 October, 2005. pp. 44-48

Vincenzi, L., Bassoli, E., Ponsi, F., Castagnetti, C. and Mancini, F. (2019). Dynamic monitoring and evaluation of bell ringing effects for the structural assessment of a masonry bell tower. *Journal of Civil Structural Health Monitoring*, pp. 439–458. Doi: 10.1007/s13349-019-00344-9

Volk, R., Stengel, J. and Schultmann, F. (2014). Building information modeling (BIM) for existing buildings — Literature review and future needs. *Automation in Construction*, 38, pp. 109-127. Doi: 10.1016/j.autcon.2013.10.023

Volodine, T. (2007) *Point cloud processing using linear algebra and graph theory*. Ph.D. thesis. Celestijnenlaan 200A, B-3001 Leuven, Belgium [Online]. Available at: http://www.cs.kuleuven.be/publicaties/doctoraten/tw/TW2007_05.pdf [Accessed 19th January 2020]

Vosselman, G. and Maas, H.-G. (2014) *Airborne and terrestrial laser scanning*. Illustrated, reprint edn. Whittles Publishing.

Westoby, M.J., Brasington, J., Glasser, N.F., Hambrey, M.J. and Reynolds, J.M. (2012). 'Structure-from-Motion' photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, 179, pp. 300-314. Doi: 10.1016/j.geomorph.2012.08.021

Willis, A., Sui, Y., Galor, K. and Sanders, D. (2010). Estimating gothic facade architecture from imagery. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*. San Francisco, CA, USA, 13-18 June 2010. pp. 43-48

Wischounig, L. (2020) *dxf2coord 2.0* (Version 2.0) [Computer program]. Available at: https://www.mathworks.com/matlabcentral/fileexchange/28791-dxf2coord-2-0 [Accessed 19th January 2020]

Wolf, P.R., Dewitt, B.A. and Wilkinson, B.E. (2014) *Elements of photogrammetry with applications in GIS*. 4th edn. New York: McGraw-Hill Education.

Woodget, A., Carbonneau, P., Visser, F. and Maddock, I. (2015). Quantifying submerged fluvial topography using hyperspatial resolution UAS imagery and structure from motion photogrammetry. *Earth Surface Processes and Landforms*, 40, pp. 47-64. Doi: 10.1002/esp.3613

Yang, H. and Xu, X. (2019). Multi-sensor technology for B-spline modelling and deformation analysis of composite structures. *Composite Structures*, 224 Doi: 10.1016/j.compstruct.2019.111000

Ye, C., Acikgoz, S., Pendrigh, S., Riley, E. and DeJong, M.J. (2018). Mapping deformations and inferring movements of masonry arch bridges using point cloud data. *Engineering Structures*, 173, pp. 530-545. Doi: 10.1016/j.engstruct.2018.06.094

Zhang, S., Hofmann, M. and Beyer, K. (2018a). A 2D typology generator for historical masonry elements. *Construction and Building Materials*, 184, pp. 440-453. Doi: 10.1016/j.conbuildmat.2018.06.085

Zhang, Y. (2013) *Image-based geometric modeling and mesh generation*. 1st edn. Springer Netherlands.

Zhang, Y., Macorini, L. and Izzuddin, B. (2016). Mesoscale partitioned analysis of brick-masonry arches. *Engineering Structures*, 124, pp. 142–166. Doi: 10.1016/j.engstruct.2016.05.046

Zhang, Y., Macorini, L. and Izzuddin, B. (2017). Numerical investigation of arches in brick-masonry bridges. *Structure and Infrastructure Engineering*, pp. 1-19. Doi: 10.1080/15732479.2017.1324883

Zhang, Y., Tubaldi, E., Macorini, L. and Izzuddin, B. (2018b). Mesoscale partitioned modelling of masonry bridges allowing for arch-backfill interaction. *Construction and Building Materials*, 173, pp. 820-842. Doi: 10.1016/j.conbuildmat.2018.03.272

Zhu, H.P., Zhou, Z.Y., Yang, R.Y. and Yu, A.B. (2007). Discrete particle simulation of particulate systems: Theoretical developments. *Chemical Engineering Science*, 62(13), pp. 3378-3396. Doi: 10.1016/j.ces.2006.12.089

Zhu, H.P., Zhou, Z.Y., Yang, R.Y. and Yu, A.B. (2008). Discrete particle simulation of particulate systems: A review of major applications and findings. *Chemical Engineering Science*, 63(23), pp. 5728-5770. Doi: 10.1016/j.ces.2008.08.006

Zucchini, A. and Lourenço, P.B. (2002). A micro-mechanical model for the homogenisation of masonry. *International Journal of Solids and Structures*, 39(12), pp. 3233-3255. Doi: 10.1016/S0020-7683(02)00230-5

Appendix A *Matlab* and *FISH* scripts

A.1 Quantifying the effect of geometric uncertainty on the robustness of structural analysis

A.1.1 Block vertex extraction of “Point2DEM” Matlab script

```
hold on

for looper=1:lwpolylines(end,1,:,:)

ranger=find(lwpolylines(:,1,:,:)==[looper])

s(looper).Area =
polyarea(lwpolylines(ranger,2,:,:),lwpolylines(ranger,3,:,:))

s(looper).Boundary_DownsampleCH=[lwpolylines(ranger,2,:,:),lwpolylines(rang
er,3,:,:)]

polyin=polyshape(s(looper).Boundary_DownsampleCH)

[s(looper).Centroid_x,s(looper).Centroid_y] = centroid(polyin)

plot(lwpolylines(ranger,2,:,:),lwpolylines(ranger,3,:,:),'-r')

end




%Filter out areas less/larger than threshold

for looper=1:size(s,2)

Area(looper,:)=s(looper).Area

end



[~, idx] = unique(round([s.Area].',5), 'rows', 'stable');  %stable optional
if you don't care about the order.

 Sunique = s(idx)



 %Sort out blocks by x coordinates

[x,region_table]=sort([Sunique.Centroid_x]);

Sregion=Sunique(region_table);

 %Number blocks
```

```
for looper=1:size(Sregion,2)

Sunique(looper).Region=region_table(looper)

end



for
looper=1:size(Sregion,2)text(Sunique(looper).Centroid_x,Sunique(looper).Cen
troid_y,horzcat('Region',mat2str(Sunique(looper).Region)))

end
```

A.1.2 Block development of "Point2DEM" Matlab script

```
as_built_x=reference_x*scale_factor

as_built_y=reference_y*scale_factor

dx1=as_built_x-idealised_x

dy1=as_built_y-idealised_y

 clearvars -except Project_name s Sregion lwpolylines Threshold_Area
Scale_Factor region_table width1 width2 dx1 dy1 scale_factor
Assessed_simulation_group Reference_simulation_group legend_ref legend_ass

NumericalModelTitle=horzcat('Point2DEM',Project_name,'.txt')

for i=1:size(Sregion,2)

 if size(Sregion(i).Boundary_DownsampleCH)~=0 &
Sregion(i).Area<=Threshold_Area % Building upon phase2 convergence

 Extremes1=unique(Sregion(i).Boundary_DownsampleCH,'rows','stable')

  Extremes1(:,1)=Extremes1(:,1)*scale_factor-dx1

 Extremes1(:,2)=Extremes1(:,2)*scale_factor*1-dy1

 s1(i).ExtremesFinal=Extremes1

 idextr2=[1,2]

 Extremes(idextr2,:)=Extremes1(idextr2,:)

idextrema_null=[3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,
25,26,27,28,29,30]

 Extremes(idextrema_null,:)=0



 if size(Extremes1,1)>2

 idextr3=[1,2,3]



 Extremes(idextr3,:)=Extremes1(idextr3,:)

 end

 if size(Extremes1,1)>3

 idextr4=[1,2,3,4]

 Extremes(idextr4,:)=Extremes1(idextr4,:)

 end
```

```
if size(Extremes1,1)>4

idextr5=[1,2,3,4,5]

Extremes(idextr5,:)=Extremes1(idextr5,:)

end

if size(Extremes1,1)>5

idextr6=[1,2,3,4,5,6]

Extremes(idextr6,:)=Extremes1(idextr6,:)

end

if size(Extremes1,1)>6

idextr7=[1,2,3,4,5,6,7]

Extremes(idextr7,:)=Extremes1(idextr7,:)

end

if size(Extremes1,1)>7

idextr8=[1,2,3,4,5,6,7,8]

Extremes(idextr8,:)=Extremes1(idextr8,:)

end

if size(Extremes1,1)>8

idextr9=[1,2,3,4,5,6,7,8,9]

Extremes(idextr9,:)=Extremes1(idextr9,:)

end

if size(Extremes1,1)>9

idextr10=[1,2,3,4,5,6,7,8,9,10]

Extremes(idextr10,:)=Extremes1(idextr10,:)

end

if size(Extremes1,1)>10

idextr11=[1,2,3,4,5,6,7,8,9,10,11]

Extremes(idextr11,:)=Extremes1(idextr11,:)

end

if size(Extremes1,1)>11
```

```
idextr12=[1,2,3,4,5,6,7,8,9,10,11,12]

Extremes(idextr12,:)=Extremes1(idextr12,:)

end

if size(Extremes1,1)>12

idextr13=[1,2,3,4,5,6,7,8,9,10,11,12,13]

Extremes(idextr13,:)=Extremes1(idextr13,:)

end

if size(Extremes1,1)>13

idextr14=[1,2,3,4,5,6,7,8,9,10,11,12,13,14]

Extremes(idextr14,:)=Extremes1(idextr14,:)

end

if size(Extremes1,1)>14

idextr15=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15]

Extremes(idextr15,:)=Extremes1(idextr15,:)

end

if size(Extremes1,1)>15

idextr16=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16]

Extremes(idextr16,:)=Extremes1(idextr16,:)

end

if size(Extremes1,1)>16

idextr17=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17]

Extremes(idextr17,:)=Extremes1(idextr17,:)

end

if size(Extremes1,1)>17

idextr18=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18]

Extremes(idextr18,:)=Extremes1(idextr18,:)

end

if size(Extremes1,1)>18

idextr19=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19]
```

```
 Extremes(idextr19,:)=Extremes1(idextr19,:)

 end

 if size(Extremes1,1)>19

 idextr20=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]

 Extremes(idextr20,:)=Extremes1(idextr20,:)

 end


idextr21=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21]

idextr22=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22]

idextr23=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23]

idextr24=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23,
24]

idextr25=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23,
24, 25]

idextr26=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23,
24, 25, 26]

idextr27=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23,
24, 25, 26, 27]

idextr28=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23,
24, 25, 26, 27 ,28]

idextr29=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23,
24, 25, 26, 27, 28, 29]

idextr30=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22, 23,
24, 25, 26, 27, 28, 29, 30]



 if size(Extremes1,1)>20

 Extremes(idextr21,:)=Extremes1(idextr21,:)

 end


 if size(Extremes1,1)>21

 Extremes(idextr22,:)=Extremes1(idextr22,:)

 end
```

```
if size(Extremes1,1)>22
Extremes(idextr23,:)=Extremes1(idextr23,:)
end

if size(Extremes1,1)>23
Extremes(idextr24,:)=Extremes1(idextr24,:)
end

if size(Extremes1,1)>24
Extremes(idextr25,:)=Extremes1(idextr25,:)
end

if size(Extremes1,1)>25
Extremes(idextr26,:)=Extremes1(idextr26,:)
end

if size(Extremes1,1)>26
Extremes(idextr27,:)=Extremes1(idextr27,:)
end

if size(Extremes1,1)>27
Extremes(idextr28,:)=Extremes1(idextr28,:)
end

if size(Extremes1,1)>28
Extremes(idextr29,:)=Extremes1(idextr29,:)
end
```

```
  if size(Extremes1,1)>29

 Extremes(idextr30,:)=Extremes1(idextr30,:)

 end



 block_uni(i,:)=[Extremes(1,:) Extremes(2,:) Extremes(3,:) Extremes(4,:)
Extremes(5,:) Extremes(6,:) Extremes(7,:) Extremes(8,:) Extremes(9,:)
Extremes(10,:) Extremes(11,:) Extremes(12,:) Extremes(13,:) Extremes(14,:)
Extremes(15,:) Extremes(16,:) Extremes(17,:) Extremes(18,:) Extremes(19,:)
Extremes(20,:) Extremes(21,:) Extremes(22,:) Extremes(23,:) Extremes(24,:)
Extremes(25,:) Extremes(26,:) Extremes(27,:) Extremes(28,:) Extremes(29,:)
Extremes(30,:) ]



 %region(i,:)=Sregion(i).Region

 end

end

block_uni=unique(block_uni,'rows','stable')




clear a1 a2 a3 xc yc zc bb1 bb2 bb3 sss



a1=['polyhedron prism a%'];

a2=['%b%'];

a3=[','];



a4=['region%'];

%region markers



%face a

idx=[1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43,45,47,49,
51,53,55,57,59,1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43
,45,47,49,51,53,55,57,59]

xc=block_uni(:,idx)
```

```
idy=[2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50
,52,54,56,58,60,
2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50,52,5
4,56,58,60]

yc=block_uni(:,idy)


zed_a=width1 % user defined depth

zed_b=width2 % user defined depth


for i=1:size(block_uni(:,1))


idza=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,
27,28,29,30]

idzb=[30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,
53,54,55,56,57,58,59,60]

zc(i,idza)=zed_a

zc(i,idzb)=zed_b

region(i,:)=i

end


b1=xc

b2=zc

b3=yc

b4=region


bb1 = num2cell(b1)

bb2 = num2cell(b2)

bb3 = num2cell(b3)

bb4 = num2str(b4)


aa1=repmat(a1,size(block_uni(:,2)),1)

aa2=repmat(a2,size(block_uni(:,2)),1)
```

```
aa4=repmat(a4,size(block_uni(:,2)),1)



sss = strcat(bb1,bb2,bb3)

A = cell2mat(sss)

A1=A(:,1:90)

A2=A(:,90:180)



B1 =  num2str(A1)

B2 =  num2str(A2)

BB1=strcat(aa1,B1)

BB2=strcat(aa2,B2,aa4,bb4)


firstrow='new'

secondrow='plot block'

BBFIN=strcat(BB1,BB2)


print = BBFIN  % Write this to file.

fid = fopen(NumericalModelTitle,'wt');

for ii = 1:size(A,1)

fprintf(fid,'%c',print(ii,:));

fprintf(fid,'\n');

end

fclose(fid)


%%%Replace %


Str = fileread(NumericalModelTitle);
```

```
Str2 = strrep(Str, '%', ' ');
FID = fopen(NumericalModelTitle, 'w');
if FID < 0, error('Cannot open file'); end
fwrite(FID, Str2, 'uchar');
fclose(FID);

%%%Replace 0 0 1 %
replacement11=horzcat('0 ',mat2str(width1),' 0')
replacement12=horzcat('0',mat2str(width2),' 0')

Str = fileread(NumericalModelTitle);
Str2 = strrep(Str, replacement11, ' ');
FID = fopen(NumericalModelTitle, 'w');
if FID < 0, error('Cannot open file'); end
fwrite(FID, Str2, 'uchar');
fclose(FID);

Str = fileread(NumericalModelTitle);
Str2 = strrep(Str, replacement12, ' ');
FID = fopen(NumericalModelTitle, 'w');
if FID < 0, error('Cannot open file'); end
fwrite(FID, Str2, 'uchar');
fclose(FID);

replacementb=horzcat(mat2str(width2),' 0 b 0')

Str = fileread(NumericalModelTitle);
Str2 = strrep(Str,replacementb, ' b ');
FID = fopen(NumericalModelTitle, 'w');
```

```
if FID < 0, error('Cannot open file'); end

fwrite(FID, Str2, 'uchar');

fclose(FID);
```

A.1.3 Structural behaviour indices FISH script

```
def CrackDetection
;Contact Failure Detector

Contacts@Slip_State1=0
Contacts@Elastic_State2=0
Contacts@Slipped_State3=0 ; All tensile failure contacts

Ring1_Slipped_31=0 ;ring 1 tensile failure contacts


Percent_Ring1_Slipped_31=0
Percent_Ring2_Slipped_32=0
Percent_Interring_Slipped_33=0

cid=contact_head
loop while cid 0
scid=c_cx(cid)
loop while scid # 0
if cx_state(scid)==1
Contacts@Slip_State1=Contacts@Slip_State1+1
cx_extra(scid)=Contacts@Slip_State1
Joints_Elastic_1(Contacts@Slip_State1,1)=c_group(cid)
endif
if cx_state(scid)==2
Contacts@Elastic_State2=Contacts@Elastic_State2+1
cx_ispare(scid)=Contacts@Elastic_State2

Joints_@_Slip_2(Contacts@Elastic_State2,1)=c_group(cid)
```

```
endif


if cx_state(scid)==3

Contacts@Slipped_State3=Contacts@Slipped_State3+1

c_extra(cid)=Contacts@Slipped_State3

Joints_Slipped_3(Contacts@Slipped_State3,1)=c_group(cid)

;;;Save hinges

if Contacts@Slipped_State3>1 & lamda #0;& crit_dispc> 2e-5

index1=Contacts@Slipped_State3

index2=Contacts@Slipped_State3-1

if Joints_Slipped_3(index1,1)# Joints_Slipped_3(index2,1)& Contact_Marker[lper,c_group(cid)]==0

Contact_Marker[lper,c_group(cid)]=1

hinge=hinge+1

Load_Per_Hinge(lper,hinge)=lamda

savedstate_name_hinge=Parameter+'SAV_DRY_JOINTED_ARCH__'+string(hinge)+'_hinge'+'_Looper'+string
(looper)+'.3dsav'

command

save @savedstate_name_hinge

endcommand

endif

endif

;;;Save first crack



if Contacts@Slipped_State3=1 & Hinge_set_table(looper,3)==int(c_group(cid)); crit_dispc> 2e-6

if save_crack=0

hinge=1

Load_First_Crack(lper,1)=lamda

Load_Per_Hinge(lper,hinge)=lamda

Contact_First_Crack(lper,1)=c_group(cid)
```

```
Stiffness(lper,1)=lamda/crit_dispc

command

;call MULTRING_FORCES_PER_REGION.3ddat

save @savedstate_name_firstcrack

;call DRYJOINTED_FORCES_PER_REGION.3ddat

endcommand

save_crack=1

endif

endif

endif


scid=cx_next(scid)

endloop


cid=c_next(cid)

endloop




if hinge#0

if hinge_marker[hinge]==0

if hinge==1 | hinge==2 | hinge==3 | hinge==4

hinge_marker[hinge]=1

Hinge_Load[hinge]=lamda

index_registration_force_registration=hinge ; This index pritns forces according to hinge-formation

;index_registration_force_registration=round(crit_dispc*10000000)/5; This records forces ever 0.5 of a
milimeter in displacement

;%if index_registration_force_registration==10| index_registration_force_registration==20|
index_registration_force_registration==30| index_registration_force_registration==40|
index_registration_force_registration==50 | index_registration_force_registration==60|
index_registration_force_registration==70 | index_registration_force_registration==80|
index_registration_force_registration==90 | index_registration_force_registration==100
```

```
;Force_Per_Cycle(index_registration_force_registration,lper)=lamda
command
call DRYJOINTED_FORCES_PER_REGION.3ddat
endcommand
endif
endif
endif
end
@CrackDetection

def Contact_Forces_RingI
;Force_Cycle(index_registration_force_registration)=lamda ; registration of force at cycle
loop master(1,26)
ci=contact_head

loop while ci#0
 ;if c_ispare(ci)==int(master)
 ;if c_extra(ci,2)==master
 if master==int(c_group(ci))
  scid=c_cx(ci)
  loop while scid # 0

SForce_RingI(master,1)=max(abs(SForce_RingI(master,1)),sqrt(cx_xsforce(scid)^2+cx_zsforce(scid)^2))
SForce_RingI(master,2)=master
SForce_RingI(master,3)=lamda

SDispl_RingI(master,1)=max(abs(SDispl_RingI(master,1)),sqrt(cx_xsdis(scid)^2+cx_zsdis(scid)^2))
SDispl_RingI(master,2)=master
SDispl_RingI(master,3)=lamda


NForce_RingI(master,1)=max(abs(cx_nforce(scid)),NForce_RingI(master,1))

NForce_RingI(master,2)=master

NForce_RingI(master,3)=lamda


NDispl_RingI(master,1)=max(abs(NDispl_RingI(master,1)),cx_ndis(scid))

NDispl_RingI(master,2)=master

NDispl_RingI(master,3)=lamda

scid=cx_next(scid)

  endloop

  endif

ci=c_next(ci)

endloop

endloop

end

@Contact_Forces_RingI
```

A.1.4 Geometric model development from tape measurements FISH script

```
;Tape measurement voussoirs



polyhedron prism a 0.001,0,0.0471974 0.010137,0.0000,0.0880243 0.0649418,0.0000,0.0744158
0.0584794,0.0000,0.0374769 b 0.001,0.0989,0.0471974 0.010137,0.0989,0.0880243
0.0649418,0.0989,0.0744158 0.0584794,0.0989,0.0374769

polyhedron prism a 0.010137,0.0000,0.0880243 0.0240769,0.0000,0.131012 0.0765092,0.0000,0.110087
0.0649418,0.0000,0.0744158 b 0.010137,0.0989,0.0880243 0.0240769,0.0989,0.131012
0.0765092,0.0989,0.110087 0.0649418,0.0989,0.0744158

polyhedron prism a 0.0240769,0.0000,0.131012 0.0431957,0.0000,0.171014 0.0929148,0.0000,0.144412
0.0765092,0.0000,0.110087 b 0.0240769,0.0989,0.131012 0.0431957,0.0989,0.171014
0.0929148,0.0989,0.144412 0.0765092,0.0989,0.110087

polyhedron prism a 0.0431957,0.0000,0.171014 0.0658802,0.0000,0.209141 0.1123,0.0000,0.176993
0.0929148,0.0000,0.144412 b 0.0431957,0.0989,0.171014 0.0658802,0.0989,0.209141 0.1123,0.0989,0.176993
0.0929148,0.0989,0.144412

polyhedron prism a 0.0658802,0.0000,0.209141 0.0942037,0.0000,0.244549 0.135666,0.0000,0.206204
0.1123,0.0000,0.176993 b 0.0658802,0.0989,0.209141 0.0942037,0.0989,0.244549 0.135666,0.0989,0.206204
0.1123,0.0989,0.176993

polyhedron prism a 0.0942037,0.0000,0.244549 0.126788,0.0000,0.275074 0.163503,0.0000,0.232283
0.135666,0.0000,0.206204 b 0.0942037,0.0989,0.244549 0.126788,0.0989,0.275074 0.163503,0.0989,0.232283
0.135666,0.0989,0.206204

polyhedron prism a 0.126788,0.0000,0.275074 0.161619,0.0000,0.302414 0.193509,0.0000,0.255835
0.163503,0.0000,0.232283 b 0.126788,0.0989,0.275074 0.161619,0.0989,0.302414 0.193509,0.0989,0.255835
0.163503,0.0989,0.232283

polyhedron prism a 0.161619,0.0000,0.302414 0.199707,0.0000,0.324997 0.226586,0.0000,0.275447
0.193509,0.0000,0.255835 b 0.161619,0.0989,0.302414 0.199707,0.0989,0.324997 0.226586,0.0989,0.275447
0.193509,0.0989,0.255835

polyhedron prism a 0.199707,0.0000,0.324997 0.240233,0.0000,0.345021 0.261063,0.0000,0.292482
0.226586,0.0000,0.275447 b 0.199707,0.0989,0.324997 0.240233,0.0989,0.345021 0.261063,0.0989,0.292482
0.226586,0.0989,0.275447

polyhedron prism a 0.240233,0.0000,0.345021 0.282239,0.0000,0.358677 0.296679,0.0000,0.304218
0.261063,0.0000,0.292482 b 0.240233,0.0989,0.345021 0.282239,0.0989,0.358677 0.296679,0.0989,0.304218
0.261063,0.0989,0.292482

polyhedron prism a 0.282239,0.0000,0.358677 0.324864,0.0000,0.368358 0.334312,0.0000,0.312739
0.296679,0.0000,0.304218 b 0.282239,0.0989,0.358677 0.324864,0.0989,0.368358 0.334312,0.0989,0.312739
0.296679,0.0989,0.304218

polyhedron prism a 0.324864,0.0000,0.368358 0.368294,0.0000,0.373296 0.371572,0.0000,0.316976
0.334312,0.0000,0.312739 b 0.324864,0.0989,0.368358 0.368294,0.0989,0.373296 0.371572,0.0989,0.316976
0.334312,0.0989,0.312739

polyhedron prism a 0.368294,0.0000,0.373296 0.412436,0.0000,0.373431 0.409072,0.0000,0.317091
0.371572,0.0000,0.316976 b 0.368294,0.0989,0.373296 0.412436,0.0989,0.373431 0.409072,0.0989,0.317091
0.371572,0.0989,0.316976
```

```
polyhedron prism a 0.412436,0.0000,0.373431 0.456578,0.0000,0.368004 0.446291,0.0000,0.312514
0.409072,0.0000,0.317091 b 0.412436,0.0989,0.373431 0.456578,0.0989,0.368004 0.446291,0.0989,0.312514
0.409072,0.0989,0.317091

polyhedron prism a 0.456578,0.0000,0.368004 0.499314,0.0000,0.357352 0.482979,0.0000,0.30337
0.446291,0.0000,0.312514 b 0.456578,0.0989,0.368004 0.499314,0.0989,0.357352 0.482979,0.0989,0.30337
0.446291,0.0989,0.312514

polyhedron prism a 0.499314,0.0000,0.357352 0.540472,0.0000,0.342635 0.518836,0.0000,0.290548
0.482979,0.0000,0.30337 b 0.499314,0.0989,0.357352 0.540472,0.0989,0.342635 0.518836,0.0989,0.290548
0.482979,0.0989,0.30337

polyhedron prism a 0.540472,0.0000,0.342635 0.580419,0.0000,0.323635 0.552946,0.0000,0.274325
0.518836,0.0000,0.290548 b 0.540472,0.0989,0.342635 0.580419,0.0989,0.323635 0.552946,0.0989,0.274325
0.518836,0.0989,0.290548

polyhedron prism a 0.580419,0.0000,0.323635 0.617597,0.0000,0.299664 0.584598,0.0000,0.253917
0.552946,0.0000,0.274325 b 0.580419,0.0989,0.323635 0.617597,0.0989,0.299664 0.584598,0.0989,0.253917
0.552946,0.0989,0.274325

polyhedron prism a 0.617597,0.0000,0.299664 0.651664,0.0000,0.272278 0.613951,0.0000,0.230321
0.584598,0.0000,0.253917 b 0.617597,0.0989,0.299664 0.651664,0.0989,0.272278 0.613951,0.0989,0.230321
0.584598,0.0989,0.253917

polyhedron prism a 0.651664,0.0000,0.272278 0.682514,0.0000,0.241314 0.640418,0.0000,0.203755
0.613951,0.0000,0.230321 b 0.651664,0.0989,0.272278 0.682514,0.0989,0.241314 0.640418,0.0989,0.203755
0.613951,0.0989,0.230321

polyhedron prism a 0.682514,0.0000,0.241314 0.709775,0.0000,0.207146 0.663806,0.0000,0.174442
0.640418,0.0000,0.203755 b 0.682514,0.0989,0.241314 0.709775,0.0989,0.207146 0.663806,0.0989,0.174442
0.640418,0.0989,0.203755

polyhedron prism a 0.709775,0.0000,0.207146 0.733115,0.0000,0.170189 0.68383,0.0000,0.142736
0.663806,0.0000,0.174442 b 0.709775,0.0989,0.207146 0.733115,0.0989,0.170189 0.68383,0.0989,0.142736
0.663806,0.0989,0.174442

polyhedron prism a 0.733115,0.0000,0.170189 0.752251,0.0000,0.130891 0.700248,0.0000,0.109021
0.68383,0.0000,0.142736 b 0.733115,0.0989,0.170189 0.752251,0.0989,0.130891 0.700248,0.0989,0.109021
0.68383,0.0989,0.142736

polyhedron prism a 0.752251,0.0000,0.130891 0.766953,0.0000,0.0897273 0.71286,0.0000,0.0737053
0.700248,0.0000,0.109021 b 0.752251,0.0989,0.130891 0.766953,0.0989,0.0897273 0.71286,0.0989,0.0737053
0.700248,0.0989,0.109021

polyhedron prism a 0.766953,0.0000,0.0897273 0.777041,0.0000,0.0471974 0.721515,0.0000,0.0372178
0.71286,0.0000,0.0737053 b 0.766953,0.0989,0.0897273 0.777041,0.0989,0.0471974
0.721515,0.0989,0.0372178 0.71286,0.0989,0.0737053



def Voxel_Marker



bi=block_head

bn=0
```

```
rn=0

loop while bi#0

rn=rn+1

rrn=27-rn

bn=bn+1 ; number blocks

;bc=b_next(bi)

xi=b_x(bi)

yi=b_y(bi)

zi=b_z(bi)


command

hide

seek bl @bi

mark reg @rrn

mark sreg @rrn

group block @rrn

endcommand

b_extra(bi)=rrn

bi=b_next(bi)

;bv=b_vol(bi)+bv

endloop


bnend=bn

;bv_ol=bv

end

@Voxel_Marker

hide
```

```
;Fixed Voussoris 1 and 27

polyhedron prism a 0.001,0,0.0471974 0.001,0.0000,0 0.0584794,0.0000,0 0.0584794,0.0000,0.0374769 b
0.001,0.0989,0.0471974 0.001,0.0989,0 0.0584794,0.0989,0 0.0584794,0.0989,0.0374769

mark reg 1

group block 1

hide



polyhedron prism a 0.777041,0.0000,0 0.777041,0.0000,0.0471974 0.721515,0.0000,0.0372178
0.721515,0.0000,0 b 0.777041,0.0989,0 0.777041,0.0989,0.0471974 0.721515,0.0989,0.0372178
0.721515,0.0989,0



mark reg 27

group block 27

fix reg 1 27

seek



def Contact_Marker



ci=contact_head

bn=0

loop while ci#0

rn=rn+1



if c_b1(ci)#c_b2(ci)



marker=min(b_extra(c_b1(ci)),b_extra(c_b2(ci)))



if marker==0 & c_x(ci)>0.35

marker=26

endif

if marker==0 & c_x(ci)<0.35
```

```
marker=1

endif


c_ispare(ci)=marker

c_group(ci)=marker

c_extra(ci,2)=marker

endif

ci=c_next(ci)


endloop


end

@Contact_Marker


;plot block colorby group

;plot joint colorby cgroup type blockface

seek

join on


Def Hinge_set_maker




Hinge_set_table[1,1]=1

Hinge_set_table[1,2]=8

Hinge_set_table[1,3]=17

Hinge_set_table[1,4]=26


Hinge_set_table[2,1]=1

Hinge_set_table[2,2]=8
```

```
Hinge_set_table[2,3]=17
Hinge_set_table[2,4]=25

Hinge_set_table[3,1]=1
Hinge_set_table[3,2]=8
Hinge_set_table[3,3]=16
Hinge_set_table[3,4]=24

Hinge_set_table[4,1]=1
Hinge_set_table[4,2]=8
Hinge_set_table[4,3]=16
Hinge_set_table[4,4]=23

Hinge_set_table[5,1]=1
Hinge_set_table[5,2]=8
Hinge_set_table[5,3]=16
Hinge_set_table[5,4]=22

Hinge_set_table[6,1]=2
Hinge_set_table[6,2]=8
Hinge_set_table[6,3]=17
Hinge_set_table[6,4]=22

Hinge_set_table[7,1]=2
Hinge_set_table[7,2]=9
Hinge_set_table[7,3]=17
Hinge_set_table[7,4]=23

Hinge_set_table[8,1]=2
```

```
Hinge_set_table[8,2]=9
Hinge_set_table[8,3]=17
Hinge_set_table[8,4]=24

Hinge_set_table[9,1]=2
Hinge_set_table[9,2]=9
Hinge_set_table[9,3]=17
Hinge_set_table[9,4]=25

Hinge_set_table[10,1]=2
Hinge_set_table[10,2]=9
Hinge_set_table[10,3]=18
Hinge_set_table[10,4]=26


Hinge_set_table[11,1]=3
Hinge_set_table[11,2]=10
Hinge_set_table[11,3]=18
Hinge_set_table[11,4]=26


Hinge_set_table[12,1]=3
Hinge_set_table[12,2]=10
Hinge_set_table[12,3]=18
Hinge_set_table[12,4]=25

Hinge_set_table[13,1]=3
Hinge_set_table[13,2]=10
Hinge_set_table[13,3]=17
```

```
Hinge_set_table[13,4]=24

Hinge_set_table[14,1]=3
Hinge_set_table[14,2]=9
Hinge_set_table[14,3]=17
Hinge_set_table[14,4]=23

Hinge_set_table[15,1]=3
Hinge_set_table[15,2]=9
Hinge_set_table[15,3]=18
Hinge_set_table[15,4]=22

Hinge_set_table[16,1]=4
Hinge_set_table[16,2]=10
Hinge_set_table[16,3]=18
Hinge_set_table[16,4]=22

Hinge_set_table[17,1]=4
Hinge_set_table[17,2]=10
Hinge_set_table[17,3]=19
Hinge_set_table[17,4]=23

Hinge_set_table[18,1]=4
Hinge_set_table[18,2]=10
Hinge_set_table[18,3]=19
Hinge_set_table[18,4]=24

Hinge_set_table[19,1]=4
Hinge_set_table[19,2]=11
```

```
Hinge_set_table[19,3]=19
Hinge_set_table[19,4]=25

Hinge_set_table[20,1]=4
Hinge_set_table[20,2]=11
Hinge_set_table[20,3]=19
Hinge_set_table[20,4]=26


Hinge_set_table[21,1]=5
Hinge_set_table[21,2]=12
Hinge_set_table[21,3]=20
Hinge_set_table[21,4]=26

Hinge_set_table[22,1]=5
Hinge_set_table[22,2]=11
Hinge_set_table[22,3]=20
Hinge_set_table[22,4]=25

Hinge_set_table[23,1]=5
Hinge_set_table[23,2]=11
Hinge_set_table[23,3]=20
Hinge_set_table[23,4]=24

Hinge_set_table[24,1]=5
Hinge_set_table[24,2]=11
Hinge_set_table[24,3]=19
Hinge_set_table[24,4]=23
```

```
Hinge_set_table[25,1]=5
Hinge_set_table[25,2]=11
Hinge_set_table[25,3]=19
Hinge_set_table[25,4]=22

loop HingeLooper(1,4)
HingeGroup=Hinge_set_table[looper,HingeLooper]
HingeGroup_anterior=HingeGroup-1
HingeGroup_posterior=HingeGroup+1
if HingeGroup==1
command
join off range jgroup @HingeGroup
join on range jgroup @HingeGroup_posterior
endcommand
endif

if HingeGroup>1 & HingeGroup<26
command
join off range jgroup @HingeGroup
join on range jgroup @HingeGroup_anterior
join on range jgroup @HingeGroup_posterior
endcommand
endif

if HingeGroup==26
command
join off range jgroup @HingeGroup
join on range jgroup @HingeGroup_anterior
endcommand
```

```
endif

endloop


command

fix reg 1 27

endcommand

;

end

@Hinge_set_maker
```

A.2 A novel approach for the semi-automated numerical modelling of regular masonry *“semi-automated Image2DEM”*

A.2.1 Joint image development “semi-automated Image2DEM” Matlab script

```
clear all, close all

close all

%%% Stage I: Image preprocessing (CHAT LEE'S PREPROCESSING)

Project_Name='Orthophoto_DSLR_0.1mm_Preprocessed'

Format='.png'

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

%(a) Read image and make greyscale

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

img=imread(horzcat(Project_Name,Format))%
img=imread('Orthophoto_DSLR_0.1mm_Cropped2.png')%


%Greyscale

greyimg=rgb2gray(img);

%imshow(greyimg)

imshow(greyimg)

imwrite(greyimg,'Gray_scale.png')

%export_fig 'Gray_scale' -native -tiff




%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

%(b)Region of interest filler, outliers and histogram

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

var1=1 % User Pre-Processing Defined Value 5 predefined vlaeus

var2=1 %
```

```
%Remove Outlier

xoutlier = roifill(greyimg,imdilate(greyimg<var1,true(var2)));

%imshow(xoutlier)

%export_fig 'Outlier' -native -tiff

imwrite(xoutlier,'Outlier.png')


%Apply Median filter

medimg=medfilt2(xoutlier);

%imshow(medimg)

%export_fig 'Median_Filter' -native -tiff

imwrite(medimg,'Median_Filter.png')


%Apply Low Pass filter

lpimg=imgaussfilt3(medimg);

%imshow(lpimg)

%export_fig 'Low_Pass_Filter' -native -tiff

imwrite(lpimg,'Low_Pass_Filter.png')


%Equalise Histogram

eqimg=histeq(medimg);

%imshow(eqimg)

%export_fig 'Equalzied_Histogram' -native -tiff

imwrite(eqimg,'Equalzied_Histogram.png')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

%(c) Edge Detection, Morphologically processing

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

smoothfactor = 0.750 ; % User Edge Detection Defined Value  - Increase
reduces detected edges 1.2 predefined values (Masonry_Panel 0.5,
[~, threshold] = edge(xoutlier, 'sobel');
edgeimg = edge(xoutlier,'sobel', threshold*smoothfactor);
%imshow(edgeimg)
imwrite(edgeimg,'Edge_Detection_Initial.png')
%export_fig 'Edge_Detection_Initial' -native -tiff

%Morphologically process edge-detected image
se90 = strel('line', 15, 90);
se0 = strel('line', 15, 0);

BWsdil = imdilate(edgeimg, [se90 se0]);
%imshow(BWsdil)
seD = strel('diamond',5);
BWfinal = imerode(BWsdil,seD);
imshow(BWfinal)
%export_fig 'Edge_Detection_Final' -native -tiff
imwrite(BWfinal,'Edge_Detection_Final.png')



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
%(d) Masks and hough transformation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

%Small mask for hough

mask= im2bw(eqimg,0.9);%PLAY WITH 0.32 VALUE FOR RESULTS (small-white; big-
black)
seD = strel('diamond',1);
mask=imerode(mask,seD);
```

```
%imshow(mask)

%export_fig 'Small_mask' -native -tiff

imwrite(mask,'Mask_Small.png')


%Large mask for boundaries of numerical model

mask2= im2bw(eqimg,0.9);%PLAY WITH 0.32 VALUE FOR RESULTS (small-white;
big-black)

seD = strel('diamond',10);

mask2=imerode(mask2,seD);

imshow(~mask2)

%export_fig 'Large_mask' -native -tiff

imwrite(mask2,'Mask_Large.png')

%%%


%Find joints with hough/ avoids noise

[H,T,R] = hough(BWfinal)

P  = houghpeaks(H,1000,'threshold',ceil(0.01*max(H(:))));

lines = houghlines(BWfinal,T,R,P,'FillGap',10,'MinLength',500);


figure, imshow(mask), hold on

max_len = 0;

for k = 1:length(lines)

xy = [lines(k).point1; lines(k).point2];


% Plot beginnings and ends of lines

% Determine the lengths of line segment

lines(k).len = norm(lines(k).point1 - lines(k).point2);


plot(xy(:,1),xy(:,2),'LineWidth',2,'Color','white');
```

```
%plot(xy(1,1),xy(1,2),'x','LineWidth',2,'Color','yellow');
%plot(xy(2,1),xy(2,2),'x','LineWidth',2,'Color','red');
end

%Save plot of houghimage
export_fig 'Hough_image' -native -tiff

%Fuse houghimage with big mask
I=imfuse(~imread('Hough_image.tif'),mask2)
%imshow(I)
imwrite(I,'Fused_Hough_and_Mask.tif')

%Rough joints
Rough_Joint_Image=im2bw(I,0.25)
%imshow(Rough_Joint_Image)
%export_fig 'Rough_joint_image' -native -tiff
imwrite(Rough_Joint_Image,'Rough_Joint_Image.png')

%Size filter/ Get rid of small areas
L=bwlabeln(Rough_Joint_Image,8)
%imshow(L)
S=regionprops(L,'area')
Blocks=ismember(L,find([S.Area]>= 200000))
%imshow(Blocks)
imwrite(Blocks,'Blocks_clean.png')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
%(e) Watershed to homogenize joint width/ smooth joint
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%


D = bwdist(Blocks);
%imshow(D)
DL = watershed(D);
%imshow(DL)

bgm = DL == 0;
%imshow(bgm)
Smooth_Joint_Image=bgm

sel=strel('disk',5)
Smooth_Joint_Image=imdilate(Smooth_Joint_Image,sel)
%imshow(Smooth_Joint_Image)

imshow(Smooth_Joint_Image)
imwrite(~Smooth_Joint_Image,'Smooth_Joint_Image.png')

title('Watershed Ridge Lines to Homogenize Joint Width)')
```

A.2.2 Block vertex extraction of “semi-automated Image2DEM” Matlab script

```
%%%%Stage I: Joint-image pre-processing

if Image_read==1

piedras  = ~imread(input_image)

else

Image2Joint_IM2DEM

piedras = BW7

end



%Image Cropping

% imshow(piedras),title 'Mark corners of the cropping Window (from
down_left to up_right)'

% [xi1, zi1] = ginput (2)

%

% ci=xi1(1)

% cf=xi1(2)

% ff=zi1(1)

% fi=zi1(2)

%

% piedras=piedras(fi:ff,ci:cf);

%Image Resizing

if Image_resize==1

piedras=imresize(piedras,0.25)

end

%Image Binarisation

imshow(piedras)

BW1=im2bw(piedras,0.70)



if Image_Segmentation==1
```

```
%resing only for cropping

BW1=imresize(BW1,4)

figure, imshow(BW1),title 'Mark polygon of image to be masked'

hold on

x=PolyDraw

mask = roipoly(BW1,x(1,:),x(2,:));

flipmask=flip(mask)

imshow(flipmask)

im3 = BW1;

BW1(~flipmask) = 0;

imshow(BW1)

BW1=imresize(BW1,0.250)


BW1(:,size(BW1,2))=1

BW1(size(BW1,1),:)=1


BW1(:,size(BW1,2+1))=1

BW1(size(BW1,1)+1,:)=1


BW1(:,1)=1

BW1(1,:)=1


BW1(:,2)=1

BW1(2,:)=1

end


%Image Morphological Processing

if Morph==1

if Mortar_Dilation~=0
```

```
BW2=BW1

sel=strel('line',Mortar_Dilation,90)

BW2=imdilate(BW2,sel)


sel=strel('line',Mortar_Dilation,0)

BW2=imdilate(BW2,sel)

BW3=~BW2

end

if Block_Erosion~=0

sel= strel('rectangle',[Block_Erosion,Block_Erosion])

BW2 = imerode(BW2,sel)


sel= strel('diamond',Block_Erosion)

BW2 = imerode(BW2,sel)

BW3=~BW2

end


else

BW3=~BW1

end


imwrite(BW1,'Processed_Image.png')


%%%%Stage II: Boundary extraction


id_del=[]

s=regionprops(BW3,'Extrema','ConvexHull','Area','Centroid')

[B,L,n,A] = bwboundaries(BW3,'noholes');
```

```
for i=1:size(s,1)

s(i).BoundaryInitial=B{i}

s(i).Boundary(:,1)=s(i).BoundaryInitial(:,2)

s(i).Boundary(:,2)=s(i).BoundaryInitial(:,1)

if s(i).Area>size(BW3,1)*size(BW3,2)*0.15 |
s(i).Area<size(BW3,1)*size(BW3,2)*0.0005

id_del=horzcat(i,id_del)

end

end




s(id_del)=[]; % these will be deleted due to large area

colorstring = repmat('rgbymc',1,10^6); % these are the colors of the
elements for plotting



%ConvexHulls of blocks

figure(11);

Title211=('The calculated "ConvexHulls" of the blocks.')

imshow(BW3)

hold on

for i=1:size(s,1)

plot(s(i).ConvexHull(:,1), s(i).ConvexHull(:,2),
colorstring(i),'LineWidth', 1)

scatter(s(i).ConvexHull(:,1), s(i).ConvexHull(:,2),  colorstring(i))

text(s(i).Centroid(:,1)-10,
s(i).Centroid(:,2)+3,horzcat('CH',mat2str(i)),'color','m')

end

title(Title211)



%Boundaries of blocks

figure(12);
```

```
Title212=('The "Boundaries" of the blocks.')

imshow(BW3)

hold on

for i=1:size(s,1)

plot(s(i).Boundary(:,1), s(i).Boundary(:,2), colorstring(i),'LineWidth', 1)

scatter(s(i).Boundary(:,1), s(i).Boundary(:,2),  colorstring(i))

text(s(i).Centroid(:,1)-10, s(i).Centroid(:,2)-
3,horzcat('BND',mat2str(i)),'color','m')

end

export_fig 'Block_Boundaries' -native -tiff

title(Title212)


%%%%Stage III: Boundary dilation

%Before convergence

figure(21);

Title211='The original "Boundaries" of the blocks.'

imshow(BW3)

hold on

for i=1:size(s,1)

plot(s(i).Boundary(:,1), s(i).Boundary(:,2),'or', 'LineWidth', 1)

end

title(Title211)


Boundary_Dilation_Threshold=0.50


for i=1:size(s,1)

 for k=1:size(s(i).Boundary,1)


s(i).Sign_x(k)=(s(i).Boundary(k,1)-
s(i).Centroid(1))/abs(s(i).Boundary(k,1)-s(i).Centroid(1))
```

```
s(i).Sign_y(k)=(s(i).Boundary(k,2)-
s(i).Centroid(2))/abs(s(i).Boundary(k,2)-s(i).Centroid(2))

s(i).Boundary(k,1)= s(i).Boundary(k,1)
+Boundary_Dilation_Threshold*s(i).Sign_x(k)

s(i).Boundary(k,2)= s(i).Boundary(k,2)
+Boundary_Dilation_Threshold*s(i).Sign_y(k)

 end

end



%After convergence

%Dilated boundaries of blocks

figure(22);

Title212=horzcat('The "Boundaries" of the blocks, dilated by
',mat2str(Boundary_Dilation_Threshold),'pixels.')

imshow(BW3)

hold on

for i=1:size(s,1)

plot(s(i).Boundary(:,1), s(i).Boundary(:,2),'or', 'LineWidth', 1)

end

title(Title212)



%%%%Stage IV: Boundary dilation

%Before convergence

figure(21);

Title211='The original "Boundaries" of the blocks.'

imshow(BW3)

hold on

for i=1:size(s,1)

plot(s(i).Boundary(:,1), s(i).Boundary(:,2),'or', 'LineWidth', 1)

end

title(Title211)
```

```
Boundary_Dilation_Threshold=0.50


for i=1:size(s,1)

 for k=1:size(s(i).Boundary,1)


s(i).Sign_x(k)=(s(i).Boundary(k,1)-
s(i).Centroid(1))/abs(s(i).Boundary(k,1)-s(i).Centroid(1))

s(i).Sign_y(k)=(s(i).Boundary(k,2)-
s(i).Centroid(2))/abs(s(i).Boundary(k,2)-s(i).Centroid(2))

s(i).Boundary(k,1)= s(i).Boundary(k,1)
+Boundary_Dilation_Threshold*s(i).Sign_x(k)

s(i).Boundary(k,2)= s(i).Boundary(k,2)
+Boundary_Dilation_Threshold*s(i).Sign_y(k)

 end

end


%After convergence

%Dilated boundaries of blocks

figure(22);

Title212=horzcat('The "Boundaries" of the blocks, dilated by
',mat2str(Boundary_Dilation_Threshold),'pixels.')

imshow(BW3)

hold on

for i=1:size(s,1)

plot(s(i).Boundary(:,1), s(i).Boundary(:,2),'or', 'LineWidth', 1)

end

title(Title212)


%%%%Stage V: Boundary downsampling

%This function downsamples and plots boundaries (very time consuming if not
done!!!)
```

```
%This function downsalmples and plots ConvexHull (LINES)

figure(31);

Title211=horzcat('The downsampled "ConvexHull " to
',mat2str(downsample_number),' vertices')

imshow(BW3)

hold on

for i=1:size(s,1)

s(i).ConvexHull_Downsample =
downsample(s(i).ConvexHull,ceil(size(s(i).ConvexHull,1)/downsample_number))

plot(s(i).ConvexHull_Downsample(:,1),
s(i).ConvexHull_Downsample(:,2),'color', colorstring(i),'linewidth',1)

scatter(s(i).ConvexHull_Downsample(:,1), s(i).ConvexHull_Downsample(:,2),
colorstring(i),'linewidth',1)

end

title(Title211)



%This function downsalmples and plots ConvexHull (LINES)

figure(32);

Title212=horzcat('The downsampled "Boundary " to
',mat2str(downsample_number),' vertices')

imshow(BW3)

hold on

for i=1:size(s,1)

s(i).Boundary_Downsample =
downsample(s(i).Boundary,ceil(size(s(i).Boundary,1)/downsample_number))

plot(s(i).Boundary_Downsample(:,1), s(i).Boundary_Downsample(:,2),'color',
colorstring(i),'linewidth',1)

scatter(s(i).Boundary_Downsample(:,1), s(i).Boundary_Downsample(:,2),
colorstring(i),'linewidth',1)

end

title(Title212)

%%%%Stage VI: Node extraction
```

```
%Resulting Image Visualisation
BW = ~BW3

if Node_Dilation==1
% sel=strel('line',Node_Dilation_Threshold,0)
% BW=imdilate(BW,sel)
% sel=strel('line',Node_Dilation_Threshold,90)
% BW=imdilate(BW,sel)
sel=strel('disk',5)
BW=imdilate(BW,sel)
imshow(BW)
end

if Node_Erosion==1
% sel=strel('line',Node_Erosion_Threshold,0)
% BW=imerode(BW,sel)
% sel=strel('line',Node_Erosion_Threshold,90)
% BW=imerode(BW,sel)
sel=strel('square',6)
BW=imerode(BW,sel)
imshow(BW)
end


%Image Skeleton Extraction
skelImage = bwmorph(BW, 'skel', inf);

figure(41);
Title411=horzcat('Image skeleton extraction.')
```

```
imshow(skelImage)

export_fig 'Image_Skeleton' -native -tiff

title(Title411)

%Image Skeleton Extraction Crossings Extraction

crossings = bwmorph(skelImage, 'branchpoints');


figure(42);

Title412=horzcat('Image crossings extraction.')

imshow(crossings)

export_fig 'Image_Crossings' -native -tiff

title(Title412)

%Nodes Extraction

Nodes=regionprops(crossings,'Extrema','ConvexHull','Area','Centroid')

Nodes_Matrix=[]


figure(43);

Title413=horzcat('The "Nodes" extracted')

imshow(BW)

hold on

for i=1:size(Nodes,1)

counter=1+size(Nodes_Matrix,1)

scatter(Nodes(i).Centroid(:,1),
Nodes(i).Centroid(:,2),colorstring(i),'linewidth',3)

Nodes_Matrix(counter,:)=Nodes(i).Centroid

end

export_fig 'Nodes_Intial' -native -tiff

title(Title413)

%With dilated blocks superimposed convergence


figure(44);
```

```
Title414=horzcat('The "Nodes" extracted with superimposed "Boundaries"')

imshow(BW)

hold on

for i=1:size(Nodes,1)

counter=1+size(Nodes_Matrix,1)

scatter(Nodes(i).Centroid(:,1),
Nodes(i).Centroid(:,2),colorstring(i),'linewidth',3)

Nodes_Matrix(counter,:)=Nodes(i).Centroid

end


for i=1:size(s,1)

plot(s(i).Boundary(:,1), s(i).Boundary(:,2),'r', 'LineWidth', 1)

end

export_fig 'Nodes_Intial_Boundaries' -native -tiff

title(Title414)


%%%%Stage VII: Nodes_Unique

%Find ConvexHull_Nodes and Nodes within Threshold

counter=0


%Unigue joints on x axis


 for i=1:size(Nodes_Matrix,1)

 for j=1:size(Nodes_Matrix,1)


dx=abs(Nodes_Matrix(i,1)-Nodes_Matrix(j,1))

dy=abs(Nodes_Matrix(i,2)-Nodes_Matrix(j,2))


if dx<=Threshold_Node_Node & dy<=0

Nodes_Matrix(i,1)=(Nodes_Matrix(i,1)+Nodes_Matrix(j,1))/2
Nodes_Matrix(i,2)=(Nodes_Matrix(i,2)+Nodes_Matrix(j,2))/2

Nodes_Matrix(j,1)=(Nodes_Matrix(i,1)+Nodes_Matrix(j,1))/2

Nodes_Matrix(j,2)=(Nodes_Matrix(i,2)+Nodes_Matrix(j,2))/2

end


  end

 end

 Nodes_Matrix=unique(Nodes_Matrix,'rows')


%  %Unigue joints on y axis

%

%  for i=1:size(Nodes_Matrix,1)

%  for j=1:size(Nodes_Matrix,1)

%

%dx=abs(Nodes_Matrix(i,1)-Nodes_Matrix(j,1))

%dy=abs(Nodes_Matrix(i,2)-Nodes_Matrix(j,2))

%

%if dy<=Threshold_Node_Node & dx<=0

%Nodes_Matrix(i,1)=(Nodes_Matrix(i,1)+Nodes_Matrix(j,1))/2

%Nodes_Matrix(i,2)=(Nodes_Matrix(i,2)+Nodes_Matrix(j,2))/2

%Nodes_Matrix(j,1)=(Nodes_Matrix(i,1)+Nodes_Matrix(j,1))/2

%Nodes_Matrix(j,2)=(Nodes_Matrix(i,2)+Nodes_Matrix(j,2))/2

%end

%

%end

%  end

%


 %This makes bricks square again
 for i=1:size(Nodes_Matrix,1)

 for j=1:size(Nodes_Matrix,1)


dx=abs(Nodes_Matrix(i,1)-Nodes_Matrix(j,1))

dy=abs(Nodes_Matrix(i,2)-Nodes_Matrix(j,2))


if dy==1

Nodes_Matrix(i,2)=Nodes_Matrix(j,2)

end


  end

 end


 %No Nearest Neighbors (by 0.5 pixels)


 for i=1:size(Nodes_Matrix,1)

 for j=1:size(Nodes_Matrix,1)


dx=abs(Nodes_Matrix(i,1)-Nodes_Matrix(j,1))

dy=abs(Nodes_Matrix(i,2)-Nodes_Matrix(j,2))


if dy<=Threshold_Node_Node_Nearest_Neighbor &
dx<=Threshold_Node_Node_Nearest_Neighbor

Nodes_Matrix(i,1)=Nodes_Matrix(j,1)

Nodes_Matrix(i,2)=Nodes_Matrix(j,2)

end


  end

 end
 %This makes sure they are unique

 Nodes_Matrix=unique(Nodes_Matrix,'rows')


Nodes=[]


figure(51);

Title111=horzcat('Unique nodes with superimposed downsampled
"Boundaries".')

imshow(BW)

hold on

for i=1:size(Nodes_Matrix,1)

counter=1+size(Nodes_Matrix,1)

scatter(Nodes_Matrix(i,1), Nodes_Matrix(i,2),colorstring(i),'linewidth',3)

Nodes(i,:)=Nodes_Matrix(i,:)

end


for i=1:size(s,1)

s(i).Boundary_Downsample =
downsample(s(i).Boundary,ceil(size(s(i).Boundary,1)/downsample_number))

plot(s(i).Boundary_Downsample(:,1), s(i).Boundary_Downsample(:,2),'color',
colorstring(i),'linewidth',1)

scatter(s(i).Boundary_Downsample(:,1), s(i).Boundary_Downsample(:,2),
colorstring(i),'linewidth',1)

end

export_fig 'Nodes_Unique_Boundaries' -native -tiff

title(Title111)


%%%%Stage VI: Nodes_Boundary_Matching

%Find ConvexHull_Downsample and Nodes within Threshold

counter=0
```

```
for i=1:size(s,1)

 for k=1:size(s(i).Boundary_Downsample,1)

 for j=1:size(Nodes_Matrix,1)


dx=abs(s(i).Boundary_Downsample(k,1)-Nodes_Matrix(j,1))

dy=abs(s(i).Boundary_Downsample(k,2)-Nodes_Matrix(j,2))


if dx<=Threshold_Node_Boundary & dy<=Threshold_Node_Boundary

counter=size(s(i).Boundary_Downsample,1)+1

s(i).Boundary_Downsample(counter,1)=Nodes_Matrix(j,1)

s(i).Boundary_Downsample(counter,2)=Nodes_Matrix(j,2)

end


   end

  end

  end


%%%%Stage VII: Nodes_Boundary_Unique

%This function removes superfluosu vertices by mainting only the

%intersection of convexHull_downsample-uniqyenodes


if Plot_superimposed==1

Project_Name='Orthophoto_DSLR_0.1mm_Preprocessed'

Format='.png'

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%(a) Read image and make greyscale
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%

BW3=imread(horzcat(Project_Name,Format))%
img=imread('Orthophoto_DSLR_0.1mm_Cropped2.png')%

end



figure(61);

imshow(BW3)

hold on



for i=1:size(s,1)

  [Intersection,ia,ib] = intersect(s(i).Boundary_Downsample,
Nodes_Matrix,'rows','legacy')

  %scatter(s(i).Boundary_Downsample(:,1), s(i).Boundary_Downsample(:,2),
colorstring(i), 'LineWidth', 1)

  scatter(Intersection(:,1), Intersection(:,2),  colorstring(i),
'LineWidth', 3)

  s(i).Boundary_Downsample_Intersection=Intersection

end

%Visualisation



figure(62);

imshow(BW3)

hold on



for i=1:size(s,1)

if size(s(i).Boundary_Downsample_Intersection,1)>2

%Clear previous values

s(i).Boundary_DownsampleCH=[]

s(i).Boundary_DownsampleBD=[]

s(i).Boundary_DownsampleALPHA=[]
```

```
K =
convhull(s(i).Boundary_Downsample_Intersection(:,1),s(i).Boundary_Downsampl
e_Intersection(:,2))

s(i).Boundary_DownsampleCH(:,1)= s(i).Boundary_Downsample_Intersection(K,1)

s(i).Boundary_DownsampleCH(:,2)= s(i).Boundary_Downsample_Intersection(K,2)



plot(s(i).Boundary_Downsample_Intersection(K,1),
s(i).Boundary_Downsample_Intersection(K,2), 'color', 'r','linewidth',3)

%scatter(s(i).Boundary_Downsample_Intersection(K,1),
s(i).Boundary_Downsample_Intersection(K,2),
colorstring(i),'LineWidth',3+2*(-1)^i)



end

end

export_fig 'Discrete elements' -native -tiff

title('The ConvexHulls of the elements')



% figure(63);

% imshow(BW3)

% hold on

%

% for i=1:size(s,1)

% if size(s(i).Boundary_Downsample_Intersection,1)>2

% L =
boundary(s(i).Boundary_Downsample_Intersection(:,1),s(i).Boundary_Downsampl
e_Intersection(:,2))

% s(i).Boundary_DownsampleBD(:,1)=
s(i).Boundary_Downsample_Intersection(L,1)

% s(i).Boundary_DownsampleBD(:,2)=
s(i).Boundary_Downsample_Intersection(L,2)

%

% plot(s(i).Boundary_DownsampleBD(:,1), s(i).Boundary_DownsampleBD(:,2),
'color', colorstring(i),'linewidth',1)

% %scatter(s(i).Boundary_DownsampleBD(:,1),
s(i).Boundary_DownsampleBD(:,2),  colorstring(i))
```

```
% end

% end

% title('The Boundaries of the elements')



% figure(64);

% imshow(BW3)

% hold on

%

%

% for i=1:size(s,1)

% if size(s(i).Boundary_Downsample_Intersection,1)>2

% M =
alphaShape(s(i).Boundary_Downsample_Intersection(:,2),s(i).Boundary_Downsam
ple_Intersection(:,1))

% s(i).Boundary_DownsampleALPHA(:,1)= M.Points(:,1)

% s(i).Boundary_DownsampleALPHA(:,2)= M.Points(:,2)

% text(s(i).Centroid(:,1)-10, s(i).Centroid(:,2)-
20,horzcat('B',mat2str(i)),'color',colorstring(i))

%

% scatter(s(i).Centroid(:,1), s(i).Centroid(:,2),
colorstring(i),'LineWidth',3)

% plot(M.Points(:,2), M.Points(:,1), 'color', colorstring(i),'linewidth',1)

% %scatter(M.Points(:,2), M.Points(:,1),  colorstring(i))

% end

% end

% title('The Alpha-Shapes of the elements')
```

A.3 A novel approach for the semi-automated numerical modelling of rubble masonry "*Cloud2DEM*"

A.3.1 "Cloud2DEM" Matlab script

```
%This code basically take a dense point cloud and voxelizes, fills and
writes it into a 3DEC model


%Clean all
clear
reducloud= load('Caerphilly_Original.xyz')
clearvars -except reducloud voxel_points voxel_arch
%% Voxelization % Iniital Parameters
%Point Cloud subject to Voxlisation
cloud=reducloud
%Volume Adjustment Coefficient
Volume_Adjustment_Factor=1.00

%Nominative Voxel Grid
grid=0.50;
% Chamar o projeito
project='Caerphilly_' % Project Names

savesuffix0=horzcat(num2str(grid),'m.xyz')
savefile= horzcat(project,savesuffix0)
%Record time
time0=horzcat('time0',project)
save time0

%% Start voxelization
voxel_arch=Voxels(reducloud, grid);
```

```
%% Fill point cloud

tamano=voxel_arch.mno


base=zeros([tamano(2) tamano(1) tamano(3)]);

base(voxel_arch.indices)=1;


%imshow(base(:,:,4));

% Fill superficial holes


Hole_Fill_superficial=1 % (default value 4) increases voxels; care it
doesn't add false voxels


SE=strel('cuboid',[Hole_Fill_superficial Hole_Fill_superficial]);

relleno_suave=imclose(base,SE);


imshow(relleno_suave(:,:,15));

idx_empty=find(relleno_suave);




% Relleno por dentro- Hole_Fill_internal and Hole_Fill_superficial are the

% most imporant variable of filling. Make sure
Hole_Fill_internal/Hole_Fill_superficial> 4 and that they don't add too
many extra voxels.


Hole_Fill_internal=round(size(base,3)/4)-2 % this greatly increases
additional voxel points (default value 20)


SE=strel('cuboid',[Hole_Fill_internal Hole_Fill_internal+2]);

relleno_completo=imclose(base,SE);
```

```
imshow(relleno_completo(:,:,12));

%Sustraigo superficie exterior al relleno completo porque así creo una
%imagen que es el interior (y evito zonas que se han rellenado por error
resta=abs(relleno_completo-relleno_suave);
imshow(resta(:,:,4));

% al quitar el envoltorio exterior he particionado en varios grupos de
% voxeles conectados, así que me quedo con la clase de mayor area/volumen
CC = bwconncomp(resta,6);

numPixels = cellfun(@numel,CC.PixelIdxList);
[biggest,idx] = max(numPixels);
relleno_suave(CC.PixelIdxList{idx}) = 1; %los voxeles de la clase con más componentes conectados les doy valor 1

imshow(relleno_suave(:,:,5));

idx_fill=find(relleno_suave); %indices de píxeles activos (despues de rellenar la imagen del exterior)
% control of filling
for i=1:size(relleno_suave,3)
imshow(relleno_suave(:,:,i))
end
%% Visualizo los vóxeles en el modelo "envoltorio" y en el modelo "con relleno"
%Modelo de voxeles que se ha rellenado
dims=[tamano(2) tamano(1) tamano(3)];
[vy vx vz] = ind2sub(dims, idx_fill);
voxel_points=[vy vx vz]
```

```
%pcshow(voxel_points) %visualizo los voxeles como una nube de puntos
hold on

%compruebo con la nube del "envoltorio" del puente
[vy_e vx_e vz_e] = ind2sub(dims, idx_empty);
voxel_points_e=[vy_e vx_e vz_e]
%pcshow(voxel_points)

save ('voxel_points.xyz', 'voxel_points','-ascii')
vx=voxel_points(:,1)
vy=voxel_points(:,2)
vz=voxel_points(:,3)

%% Calculate 8 nodes for each voxel
%Nodes in voxel coordinates
nodes(:,1:3)=[vx vy vz];
nodes(:,4:6)=[(vx+1) vy vz];
nodes(:,7:9)=[(vx+1) vy (vz+1)];
nodes(:,10:12)=[vx vy (vz+1)];
nodes(:,13:15)=[vx (vy+1) vz];
nodes(:,16:18)=[(vx+1) (vy+1) vz];
nodes(:,19:21)=[(vx+1) (vy+1) (vz+1)];
nodes(:,22:24)=[vx (vy+1) (vz+1)];

%Nodes in metric units
x_idx=[1;4;7;10;13;16;19;22];
y_idx=x_idx+1;
z_idx=x_idx+2;
```

```
num_voxeles=ceil((max (cloud) - min(cloud)) / grid);

paso=(max (cloud) - min(cloud))./num_voxeles;


metric_nodes(:,x_idx)=Volume_Adjustment_Factor*nodes(:,x_idx)*paso(1);

metric_nodes(:,y_idx)=Volume_Adjustment_Factor*nodes(:,y_idx)*paso(2);

metric_nodes(:,z_idx)=Volume_Adjustment_Factor*nodes(:,z_idx)*paso(3);


empty_final_pc(:,1)=Volume_Adjustment_Factor*paso(1)*voxel_points_e(:,1)

empty_final_pc(:,2)=Volume_Adjustment_Factor*paso(2)*voxel_points_e(:,2)

empty_final_pc(:,3)=Volume_Adjustment_Factor*paso(3)*voxel_points_e(:,3)


final_pc(:,1)=Volume_Adjustment_Factor*paso(1)*voxel_points(:,1)

final_pc(:,2)=Volume_Adjustment_Factor*paso(2)*voxel_points(:,2)

final_pc(:,3)=Volume_Adjustment_Factor*paso(3)*voxel_points(:,3)


figure

hold on

%pcshow(empty_final_pc,'red')

%pcshow(final_pc,'yellow')

save('Boundary_15.xyz', 'empty_final_pc','-ascii')

save('Fill_15.xyz', 'final_pc','-ascii')


xc = metric_nodes(:,x_idx)

yc = metric_nodes(:,y_idx)

zc = metric_nodes(:,z_idx)


% para ahorrar tiempo se hace en grupos de set_step; con

% mas de 10000 se hace lento el programa
```

```
nVoxels=size(voxel_points,1)

set_step=10000

sets=round(nVoxels/set_step)+1

a1=['polyhedron prism a%'];

a2=['%b%'];

a3=[','];




for j=1:sets-1

  m=(j-1)*set_step+1

  if j < sets-1

  set_stepm=j*set_step

  elseif j==sets-1 % for the case of the last set of voxels

  set_stepm=size(voxel_points,1)

  set_step=size(voxel_points,1)-(j-1)*set_step

  end

  kmeter=1


for k=m:set_stepm % MAYBE THIS LOOP CAN BE NESTED FOR MULTIPLES OF 10000
NVOXELS


b1(kmeter,:)=[metric_nodes(k,1),metric_nodes(k,4),metric_nodes(k,7),metric_
nodes(k,10),metric_nodes(k,13),metric_nodes(k,16),metric_nodes(k,19),metric
_nodes(k,22)];

b2(kmeter,:)=[metric_nodes(k,2),metric_nodes(k,5),metric_nodes(k,8),metric_
nodes(k,11),metric_nodes(k,14),metric_nodes(k,17),metric_nodes(k,20),metric
_nodes(k,23)];

b3(kmeter,:)=[metric_nodes(k,3),metric_nodes(k,6),metric_nodes(k,9),metric_
nodes(k,12),metric_nodes(k,15),metric_nodes(k,18),metric_nodes(k,21),metric
_nodes(k,24)];


kmeter=kmeter+1

end
```

```
bb1 = num2cell(b1)
bb2 = num2cell(b2)
bb3 = num2cell(b3)

aa1=repmat(a1,set_step,1)
aa2=repmat(a2,set_step,1)
sss = strcat(bb1,bb2,bb3)
A = cell2mat(sss)
A1=A(:,1:12)
A2=A(:,13:24)


B1 =  num2str(A1)
B2 =  num2str(A2)
BB1=strcat(aa1,B1)
BB2=strcat(aa2,B2)


suffix0='_part'
filename= horzcat(project,suffix0)
suffix1='grid'
suffix2='.3ddat'
%suffix=sprintf ( '%s %i', suffix1,grid,suffix2);

fname = sprintf ( '%s %i', filename,j,suffix2);

BBFIN=strcat(BB1,BB2)
print = BBFIN  % Write this to file.
```

```
fid = fopen(fname,'wt');

for ii = 1:size(A,1)

fprintf(fid,'%c',print(ii,:));

fprintf(fid,'\n');

end

fclose(fid)


%%%Replace %


Str = fileread(fname);

Str2 = strrep(Str, '%', ' ');

FID = fopen(fname, 'w');

if FID < 0, error('Cannot open file'); end

fwrite(FID, Str2, 'uchar');

fclose(FID);


clearvars b1 b2 b3 aa1 aa2 sss A A1 A2

end


time1=horzcat('time1',project) % finish time

save time1
```

Appendix B Data and results

B.1 Quantifying the effect of geometric uncertainty on the robustness of structural analysis

Table B- 1: SfM photogrammetry vs tape measurement. Geometric differences (NU).

Aspect	Joint dip	Joint midp. x	Joint midp. z	Joint length	Block volume	Block centr. x	Block centr. z
Average	0	2	2	2	6	2	2
Minimum error	-4	0	-1	-3	1	0	-1
Maximum error	5	3	4	7	10	3	4
Standard deviation	2	1	2	3	3	1	2

Table B- 2: SfM photogrammetry vs tape measurement. Stiffness at 1st hinge formation and load multipliers at each hinge formation differences (NU).

Aspect	Serviceability limit states				Ultimate
	Stiffness	λ^{h1}	λ^{h2}	λ^{h3}	λ^{h4}
Average	14	1	4	4	6
Minimum error	2	-1	2	2	1
Maximum error	46	3	6	7	10
Standard deviation	10	1	1	1	3

Table B- 3: SfM photogrammetry vs tape measurement. Internal force differences between geometric models (NU).

Aspect	1st hinge formation				2nd hinge formation				3rd hinge formation				4th hinge formation			
	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n
Average	3	0	4	3	3	0	4	3	3	0	4	4	3	0	4	3
Minimum error	-14	-21	0	-15	-14	-21	0	-15	-14	-21	0	-15	-14	-21	0	-15
Maximum error	22	5	10	13	22	5	10	13	22	5	10	13	22	8	10	13
Standard deviation	11	7	3	9	11	7	3	9	11	7	3	9	11	8	3	10

Table B- 4: SfM photogrammetry vs tape measurement. Correlation matrix of geometrical properties with stiffness at 1st hinge formation and load multipliers at each hinge formation.

Correlation coefficient of uncertainty	Stiffness	λ^{h1}	λ^{h2}	λ^{h3}	λ^{h4}
J_1 dip	0.05	0.01	0.4	-0.05	-0.84
J_2 dip	0.52	0.14	0.45	0.45	0.06
J_3 dip	0.23	0.2	0.37	0.2	0.09
J_4 dip	-0.22	0.37	0.24	0.15	-0.01
J_1 length	0.49	-0.19	0.1	0.29	0.26
J_2 length	0.9	0	0.31	0.64	0.43
J_3 length	0.8	0.02	0.12	0.53	0.29
J_4 length	0.21	0.58	0.18	0.2	-0.05
J_1 midpoint x	0.85	-0.1	0.24	0.62	0.52
J_2 midpoint x	0.52	0.18	0.41	0.63	0.32
J_3 midpoint x	-0.76	-0.15	-0.44	-0.52	-0.04
J_4 midpoint x	-0.28	-0.4	-0.03	-0.07	0.05

J_1 midpoint z	-0.43	0.03	-0.04	-0.48	-0.47
J_2 midpoint z	0.48	-0.13	-0.17	0.36	0.74
J_3 midpoint z	0.01	0.21	0.38	0.14	0.15
J_4 midpoint z	-0.05	-0.48	-0.35	-0.25	0.01
B_1 volume	-0.22	-0.03	0.27	0.37	-0.22
B_2 volume	0.27	0.13	0.45	0.57	0.27
B_3 volume	-0.34	0.21	-0.23	-0.08	-0.34
B_1 centroid x	0.27	-0.03	0.39	0.62	0.27
B_2 centroid x	0.35	0.05	0.32	0.62	0.35
B_3 centroid x	0.08	0.52	0.3	0.45	0.08
B_1 centroid z	0.4	-0.04	0.32	0.62	0.4
B_2 centroid z	-0.21	-0.07	-0.37	-0.59	-0.21
B_3 centroid z	-0.26	-0.34	-0.32	-0.56	-0.26

Table B- 5: SfM photogrammetry vs tape measurement. Correlation matrix of geometrical properties with joint forces at each hinge formation.

Correlation	1st hinge formation				2nd hinge formation				3rd hinge formation				4th hinge formation			
	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n
J_1 dip	0.12	0.33	0.24	-0.02	0.13	0.33	0.24	0.01	0.16	0.33	0.24	0.04	0.06	0.26	0.19	0
J_2 dip	0.33	0.59	0.2	0.03	0.34	0.59	0.2	0.03	0.33	0.59	0.2	0.04	0.29	0.69	0.37	0.03
J_3 dip	-0.38	0.2	0.41	0.17	-0.38	0.2	0.41	0.15	-0.34	0.21	0.41	0.14	-0.2	0.16	0.45	0.27
J_4 dip	0.09	0.17	0.08	0.56	0.08	0.18	0.08	0.56	0.07	0.17	0.08	0.57	0.11	0.1	0.2	0.63
J_1 length	-0.29	-0.19	0.01	0.01	-0.28	-0.2	0.01	0	-0.31	-0.21	0.01	-0.02	-0.32	-0.03	0.03	0
J_2 length	-0.4	0.16	0.49	0.24	-0.39	0.16	0.49	0.2	-0.42	0.15	0.49	0.17	-0.35	0.19	0.52	0.19
J_3 length	-0.52	-0.04	0.17	0.12	-0.51	-0.04	0.17	0.09	-0.53	-0.05	0.17	0.05	-0.32	0.02	0.23	0.15
J_4 length	0.09	0.1	-0.08	0.68	0.11	0.1	-0.08	0.67	0.12	0.1	-0.08	0.68	0.31	-0.05	-0.11	0.75
J_1 midpoint x	-0.48	0.04	0.43	0.03	-0.48	0.04	0.43	-0.01	-0.51	0.03	0.43	-0.05	-0.39	0.15	0.53	0
J_2 midpoint x	0.17	0.51	0.27	0.07	0.17	0.52	0.27	0.06	0.15	0.51	0.27	0.05	0.23	0.63	0.45	0.05
J_3 midpoint x	0.29	-0.24	-0.51	-0.22	0.29	-0.25	-0.51	-0.2	0.29	-0.24	-0.51	-0.17	0.22	-0.29	-0.63	-0.18
J_4 midpoint x	0	0.06	0.07	-0.43	-0.03	0.06	0.07	-0.43	-0.05	0.06	0.07	-0.42	-0.25	0.06	0.2	-0.46
J_1 midpoint z	0.62	0.03	-0.28	-0.03	0.62	0.03	-0.28	0.01	0.65	0.04	-0.28	0.05	0.51	0.15	-0.35	0
J_2 midpoint z	-0.36	-0.08	-0.1	0.17	-0.36	-0.09	-0.1	0.14	-0.4	-0.1	-0.1	0.09	-0.3	-0.06	-0.01	0.13
J_3 midpoint z	0.31	0.29	0.64	0.03	0.31	0.3	0.64	0.04	0.31	0.3	0.64	0.04	0.29	0.37	0.7	-0.03
J_4 midpoint z	-0.18	-0.3	0.09	-0.47	-0.18	-0.31	0.09	-0.47	-0.18	-0.31	0.09	-0.48	-0.17	-0.03	-0.08	-0.52
B_1 volume	-0.56	0.19	0.43	0.1	-0.56	0.18	0.43	0.08	-0.56	0.18	0.43	0.06	-0.45	-0.01	0.42	0.07
B_2 volume	-0.16	0.26	0.64	0.11	-0.15	0.27	0.64	0.09	-0.16	0.27	0.64	0.07	-0.06	0.35	0.7	0.1
B_3 volume	-0.06	-0.1	-0.44	0.53	-0.04	-0.11	-0.44	0.53	-0.03	-0.11	-0.44	0.54	-0.03	-0.23	-0.49	0.54
B_1 centroid x	-0.38	0.21	0.54	0.15	-0.38	0.21	0.54	0.12	-0.4	0.2	0.54	0.09	-0.33	0.26	0.59	0.11
B_2 centroid x	-0.43	0.14	0.46	0.2	-0.42	0.14	0.46	0.17	-0.44	0.13	0.46	0.13	-0.31	0.18	0.52	0.18
B_3 centroid x	-0.15	0.14	0.14	0.66	-0.13	0.14	0.14	0.64	-0.13	0.14	0.14	0.63	0.08	0.06	0.18	0.71
B_1 centroid z	-0.38	0.16	0.5	0.11	-0.37	0.16	0.5	0.08	-0.39	0.16	0.5	0.04	-0.3	0.25	0.58	0.08
B_2 centroid z	0.37	-0.18	-0.47	-0.2	0.37	-0.18	-0.47	-0.17	0.38	-0.18	-0.47	-0.14	0.28	-0.23	-0.54	-0.18
B_3 centroid z	0.35	-0.09	-0.28	-0.43	0.34	-0.1	-0.28	-0.4	0.34	-0.09	-0.28	-0.37	0.06	-0.09	-0.34	-0.5

B.2 A novel approach for the semi-automated numerical modelling of regular masonry "*semi-automated Image2DEM*"

Table B- 6: manual CAD-based vs IPT-based block segmentation. Geometric differences (NU).

Aspect	Joint dip	Joint midp. x	Joint midp. z	Joint length	Block volume	Block centr. x	Block centr. z
Average	0	2	3	4	0	0	0
Minimum error	-1	0	0	1	0	0	0
Maximum error	4	4	4	7	0	0	0
Standard	1	1	1	2	0	0	0

Table B- 7: manual CAD-based vs IPT-based block segmentation. Stiffness at 1st hinge formation and load multipliers at each hinge formation differences (NU).

Aspect	Serviceability limit states				Ultimate
	Stiffness	λ^{h1}	λ^{h2}	λ^{h3}	λ^{h4}
Average	11	0	0	0	-1
Minimum error	-3	-2	-2	-1	-8
Maximum error	26	2	1	1	2
Standard deviation	8	1	1	1	3

Table B- 8: manual CAD-based vs IPT-based block segmentation. Internal force differences between geometric models (NU).

Aspect	1st hinge formation				2nd hinge formation				3rd hinge formation				4th hinge formation			
	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n
Average	10	4	2	4	10	4	2	4	9	3	2	4	9	3	2	4
Minimum error	-11	-4	-3	2	-11	-4	-3	1	-11	-4	-3	2	-11	-4	-3	2
Maximum error	70	40	12	14	70	34	12	14	63	18	12	14	63	18	12	14
Standard deviation	17	8	3	3	17	7	3	3	16	4	3	3	16	4	3	3

Table B- 9: manual CAD-based vs IPT-based block segmentation. Correlation matrix of geometrical properties with stiffness at 1st hinge formation and load multipliers at each hinge formation.

Correlation coefficient of uncertainty	Stiffness	λ^{h1}	λ^{h2}	λ^{h3}	λ^{h4}
J_1 dip	0.13	0.32	0.27	-0.11	-0.92
J_2 dip	-0.42	-0.07	-0.28	0.06	0.42
J_3 dip	0.06	0.02	-0.22	-0.14	0.39
J_4 dip	0.05	-0.23	-0.69	0.11	0.13
J_1 length	0.75	0.48	0.18	0.29	-0.77
J_2 length	0.88	0.57	0.21	0.28	-0.48
J_3 length	0.76	0.5	0.2	0.21	-0.67
J_4 length	0.01	0.48	0.55	-0.45	-0.06
J_1 midpoint x	0.89	0.51	0.1	0.41	-0.47
J_2 midpoint x	0.83	0.65	0.28	0.3	-0.26
J_3 midpoint x	-0.69	-0.53	-0.21	-0.21	0.73
J_4 midpoint x	-0.05	-0.5	-0.37	0.5	0.01
J_1 midpoint z	0.75	0.44	0.06	0.38	-0.21

J_2 midpoint z	-0.73	-0.64	-0.33	-0.26	0.19
J_3 midpoint z	0.57	0.5	-0.01	0.03	-0.44
J_4 midpoint z	-0.1	-0.27	0.28	0.38	-0.11
B_1 volume	-0.46	0.59	0.25	0.36	-0.46
B_2 volume	-0.27	0.23	0.08	0.2	-0.27
B_3 volume	0.21	-0.39	-0.18	-0.06	0.21
B_1 centroid x	0.09	0.35	0	0.44	0.09
B_2 centroid x	-0.19	0.55	0.14	0.28	-0.19
B_3 centroid x	0.13	-0.48	-0.12	-0.27	0.13
B_1 centroid z	0.13	-0.41	-0.06	-0.31	0.13
B_2 centroid z	0.24	-0.5	-0.07	-0.18	0.24
B_3 centroid z	-0.25	0.5	0.15	0.12	-0.25

Table B- 10. manual CAD-based vs IPT-based block segmentation. Correlation matrix of geometrical properties with joint forces at each hinge formation.

Correlation	1st hinge formation				2nd hinge formation				3rd hinge formation				4th hinge formation			
	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n	F_1^n	F_2^n	F_3^n	F_4^n
J_1 dip	0.87	0.47	0.2	0.27	0.87	0.47	0.2	0.27	0.87	0.47	0.2	0.27	0.87	0.47	0.2	0.27
J_2 dip	-0.21	-0.32	-0.26	-0.16	-0.21	-0.32	-0.26	-0.16	-0.21	-0.32	-0.26	-0.16	-0.21	-0.32	-0.26	-0.16
J_3 dip	-0.31	-0.14	0.11	0.08	-0.31	-0.14	0.11	0.08	-0.31	-0.14	0.11	0.08	-0.31	-0.14	0.11	0.08
J_4 dip	0.13	0.4	0.3	-0.01	0.13	0.4	0.3	-0.01	0.13	0.4	0.3	-0.01	0.13	0.4	0.3	-0.01
J_1 length	0.41	0.24	0.22	0.25	0.41	0.24	0.22	0.25	0.41	0.24	0.22	0.25	0.41	0.24	0.22	0.25
J_2 length	0.09	0.03	0.17	0.29	0.09	0.03	0.17	0.29	0.09	0.03	0.17	0.29	0.09	0.03	0.17	0.29
J_3 length	0.26	0.09	0.12	0.25	0.26	0.09	0.12	0.25	0.26	0.09	0.12	0.25	0.26	0.09	0.12	0.25
J_4 length	-0.14	-0.24	-0.27	0.32	-0.14	-0.24	-0.27	0.32	-0.14	-0.24	-0.27	0.32	-0.14	-0.24	-0.27	0.32
J_1 midpoint x	0.01	0	0.13	0.17	0.01	0	0.13	0.17	0.01	0	0.13	0.17	0.01	0	0.13	0.17
J_2 midpoint x	-0.16	-0.15	0.05	0.2	-0.16	-0.15	0.05	0.2	-0.16	-0.15	0.05	0.2	-0.16	-0.15	0.05	0.2
J_3 midpoint x	-0.32	-0.09	-0.05	-0.22	-0.32	-0.09	-0.05	-0.22	-0.32	-0.09	-0.05	-0.22	-0.32	-0.09	-0.05	-0.22
J_4 midpoint x	0.09	0.14	0.25	-0.45	0.09	0.14	0.25	-0.45	0.09	0.14	0.25	-0.45	0.09	0.14	0.25	-0.45
J_1 midpoint z	-0.27	-0.08	0.06	0.12	-0.27	-0.08	0.06	0.12	-0.27	-0.08	0.06	0.12	-0.27	-0.08	0.06	0.12
J_2 midpoint z	0.2	0.18	-0.02	-0.15	0.2	0.18	-0.02	-0.15	0.2	0.18	-0.02	-0.15	0.2	0.18	-0.02	-0.15
J_3 midpoint z	0.15	0	0.15	0.21	0.15	0	0.15	0.21	0.15	0	0.15	0.21	0.15	0	0.15	0.21
J_4 midpoint z	-0.03	-0.21	0.01	-0.45	-0.03	-0.21	0.01	-0.45	-0.03	-0.21	0.01	-0.45	-0.03	-0.21	0.01	-0.45
B_1 volume	0.04	-0.02	0.14	0.2	0.04	-0.02	0.14	0.2	0.04	-0.02	0.14	0.2	0.04	-0.02	0.14	0.2
B_2 volume	0.14	-0.05	-0.45	0.06	0.14	-0.05	-0.45	0.06	0.14	-0.05	-0.45	0.06	0.14	-0.05	-0.45	0.06
B_3 volume	0	0.15	0.45	-0.03	0	0.15	0.45	-0.03	0	0.15	0.45	-0.03	0	0.15	0.45	-0.03
B_1 centroid x	-0.53	-0.26	0.03	0.06	-0.53	-0.26	0.03	0.06	-0.53	-0.26	0.03	0.06	-0.53	-0.26	0.03	0.06
B_2 centroid x	-0.17	-0.11	-0.07	0.34	-0.17	-0.11	-0.07	0.34	-0.17	-0.11	-0.07	0.34	-0.17	-0.11	-0.07	0.34
B_3 centroid x	0.14	0.07	0.03	-0.34	0.14	0.07	0.03	-0.34	0.14	0.07	0.03	-0.34	0.14	0.07	0.03	-0.34
B_1 centroid z	0.27	0.11	-0.11	-0.22	0.27	0.11	-0.11	-0.22	0.27	0.11	-0.11	-0.22	0.27	0.11	-0.11	-0.22
B_2 centroid z	0.1	0.07	-0.09	-0.32	0.1	0.07	-0.09	-0.32	0.1	0.07	-0.09	-0.32	0.1	0.07	-0.09	-0.32
B_3 centroid z	-0.04	-0.03	0.13	0.32	-0.04	-0.03	0.13	0.32	-0.04	-0.03	0.13	0.32	-0.04	-0.03	0.13	0.32