Bulletin of Earthquake Engineering
https://doi.org/10.1007/s10518-023-01726-7

S.I. : AI, S AND BDA IN EARTHQUAKE ENGINEERING

Automated image-based generation of finite element models for masonry buildings

Bryan German Pantoja-Rosero[1] · **Radhakrishna Achanta**[2] · **Katrin Beyer**[1]

Received: 5 December 2022 / Accepted: 12 June 2023

Abstract
To predict the response of masonry buildings to various types of loads, engineers use finite element models, specifically solid-element and macro-element models. For predicting masonry responses to seismic events in particular, equivalent frame models—a subcategory of macro-element models—are a common choice because of their low computational cost. However, an existing bottleneck in modeling pipelines is generating the geometry of the model, which is currently a slow and laborious process that is done manually using computer-aided design tools. In this paper, we address this by automating the modelling process using recent advancements in computer vision and machine learning. We present an image-based end-to-end pipeline that automatically generates finite element meshes for solid-element and equivalent-frame models of the outer walls of free-standing historical masonry buildings. As the input, our framework requires RGB images of the buildings that are processed using structure-from-motion algorithms, which create 3D geometries, and convolutional neural networks, which segment the openings and their corners. These layers are then combined to generate level of detail models. We tested our pipeline on structures with irregular surface geometries and opening layouts. While generating the solid element mesh from the level of detail model is straightforward, generating equivalent frame models required algorithms for segmenting the façade and the meshing. Experts in the field analyzed the generated equivalent frame models and determined them to be useful for numerical modeling. These finite element geometries will be invaluable for future predictions of the seismic response of damaged and undamaged buildings. The codes and dataset are publicly available for future studies and benchmarking (https://github.com/eesd-epfl/FEM_buildings and https://doi.org/10.5281/zenodo.8094306).

Keywords Masonry buildings · Finite element method · Equivalent frame model · Macro-elements · Computer vision · Structure from motion · Machine learning

1 Introduction

Masonry buildings together with reinforced concrete structures account for the largest number of earthquake-related fatalities (D'Ayala 2013) because they are among the most vulnerable structures for various types of loads, including the dynamic loads of earthquakes as

Extended author information available on the last page of the article

well as quasi-static loads such as differential settlements. This highlights the importance of understanding the mechanical response of these type of structures. To study the response of masonry elements at various scales, numerous experimental campaigns have been conducted over the years (e.g., (Morandi et al. 2018; Vasconcelos and Lourenço 2009; Beyer and Dazio 2012; Senaldi et al. 2020; Rezaie et al. 2020; Ghezelbash et al. 2020; Wilding et al. 2018; Parisse et al. 2021)), but experimental studies are limited due to economic restraints and the availability of appropriate testing infrastructure. As a cheaper and more accessible alternative, numerical modeling is thus used to up-scale experimental findings on individual components to entire buildings and to generalize conclusions (e.g., (Celano et al. 2021; Ma et al. 2022; D'Altri et al. 2018; Kouris et al. 2022; Pereira et al. 2021; Zhang and Beyer 2019; Zhang et al. 2017; Soti et al. 2020)). However, much work is still needed to improve the accuracy of models of masonry structures due to their complex behavior, which is caused by the heterogeneous and anisotropic nature of its material as well as and various sources of uncertainties (Preciado and Orduna 2018).

For modeling the mechanical behavior of masonry material, different approaches, scales, and strategies have been proposed, which can be grouped into four categories: block-based, continuum, geometry-based, and macro-element models (D'Altri et al. 2020). Block-based models (Serpieri et al. 2017; Portioli et al. 2014; Lourenço and Rots 1997; Baraldi and Cecchi 2017) represent the behavior of masonry at the size of the material's primary variability, which is defined by mortar or dry joints connecting blocks. This method may account for the masonry's texture, which has a substantial impact on its anisotropy and failure pattern. Continuum models (Massart et al. 2007; Berto et al. 2002; Milani et al. 2007; Petracca et al. 2016), on the other hand, consider masonry as a deformable body, enabling wider discretization dimensions and reduced computational cost. Nonetheless, constitutive laws for masonry must be defined adequately, which can be accomplished by either direct approaches or homogenization procedures. Geometry-based models (Marmo and Rosati 2017; Fraternali 2010; Chiozzi et al. 2017; Block and Lachauer 2014) depict the structure as a rigid body based solely on its geometry and loading conditions. Typically, solutions based on limit analysis are employed to assess structural equilibrium and/or collapse. Macroelement models (Vanin et al. 2020; Lagomarsino et al. 2013; Belmouden and Lestuzzi 2009; Caliò et al. 2012; Rinaldin et al. 2016), which are commonly used for seismic assessment, idealize the structure into panel-scale components with nonlinear responses.

In this study, we focus on solid element models and on macro element models (to which equivalent frame models (EFM) belong). In EFMs, walls are represented by frames, which are composed of deformable elements connected through rigid nodes. The deformable vertical elements are called piers and the deformable horizontal elements are called spandrels (Lagomarsino et al. 2013). The simplification of the walls as a frame model is based on post-earthquake observations, which showed that damage tends to concentrate in piers and spandrels (D'Altri et al. 2020). EFMs are mainly used for analyzing the global building response, which is associated with the in-plane capacity of walls (e.g., (Penna et al. 2014)), though EFM approaches have recently been proposed that can directly or indirectly consider the out-of-plane response (Vanin et al. 2020; Angiolilli et al. 2021). When compared to solid-element models, the number of degrees of freedom of an EFM of a masonry building is typically three orders of magnitudes lower, which reduces the computational costs and makes EFMs appealing for engineering practice. Of course, however, the equivalent-frame simplification comes at a cost. For example, in addition the lack of an out-of-plane response, which has since been overcome, D'Altri et al. (2020) also lists the toothing between orthogonal walls as an example of a structural detail that cannot be explicitly considered in EFMs.

Modeling masonry buildings with EFMs requires the discretization of walls into piers, spandrels, and nodes, which can be particularly challenging for walls with an irregular layout of openings. Developed based on damage observations in buildings, Quagliarini et al. (2017) present a collection of rules for discretizing walls (Augenti 2006; Bracchi et al. 2015; Dolce 1991; Lagomarsino et al. 2013), and the most recent work on this topic was presented by Morandini et al. (2022). As a general rule, openings occur above and below spandrels and to the left and right of piers. The length of a spandrel element is typically taken as the mean width of the openings above and below the spandrel. If the openings are not aligned, non-overlapping areas between them are considerate to be rigid. To determine the length of the pier element, there are numerous proposals that differ in their consideration of openings to the left and right of the pier, the direction of the seismic load, and numerical observations. These varying sets of rules produce similar discretizations for regular opening layouts, but the results differ significantly for irregular opening layouts. Figure 1 presents example discretizations of walls with regular and irregular opening distributions, as presented by Quagliarini et al. (2017).

Multiple works have used EFMs to study the seismic performance of specific buildings (e.g., (Tomić et al. 2021, 2021; Fenu et al. 2022; Miglietta et al. 2021; Vanin et al. 2020; Angiolilli et al. 2021)). In these studies, the finite element geometry was typically drawn manually using computer-aided design (CAD) tools, which can be cumbersome, particularly when the opening layout is irregular. Because several sets of rules exist, the irregular layouts are typically discretized either by the expert selection of the most appropriate method from several or by modifying a discretization after evaluating the numerical results. As such, automating the discretization will accelerate the modeling process. Automation would also make EFM simulations more feasible in post-earthquake assessments when a large number of buildings need to be modeled in a short amount of time. This automation has been done using tools from computer vision and machine learning. For example, 3D point clouds from LiDAR or laser scan devices have been used previously for the generation of finite element models of masonry buildings (Lucidi et al. 2021; Kujawa et al. 2020; Bassier et al. 2019; Funari et al. 2021; Shabani et al. 2021). Additionally, RGB images and photogrammetry techniques have also been explored previously for generating models for finite or discrete element methods (Shabani et al. 2021; Pantoja-Rosero et al. 2023; Abu-Haifa and Lee 2022; Kassotakis and Sarhosis 2021).

Our work aims to contribute to the automatic generation of finite element models of the facade walls of historical masonry buildings. We assume for this purpose that the outer walls are the main load-bearing elements, and we do not model slabs or ring beams, limiting this work to historical masonry elements. We present in this study the first image-based framework that automatically generates finite-element geometries for solid-element models and EFMs for masonry buildings. Because of the subjectivity of the discretization process, our intention is not to produce EFM models that give the most accurate simulation results, but instead provide a generic EFM that meets

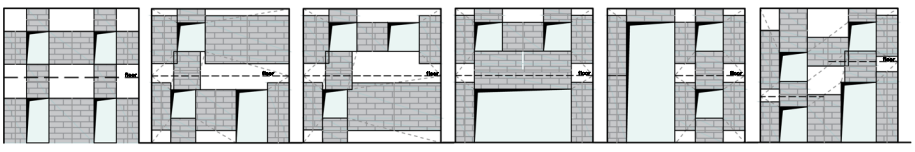

Fig. 1 Example approaches for discretizing facades as piers, spandrels, and rigid nodes (from Quagliarini et al. (2017))

the criteria of experienced design engineers, which can then be refined to suit specific needs. Available refinements include modifying the discretization or including elements such as roofs, floors, and gables for more accurate results. Overall, we believe that our presented framework can be useful for modeling numerous buildings in a short period of time, such as in post-earthquake assessments.

In this paper, we outline our methodology and apply it to various historical masonry buildings. Section 2 describes the framework and its components, including the construction of a simplified 3D model that captures the information required to generate EFMs, the generation of the EFMs from the geometrical model, and the metrics used to assess the performance of our method. Section 3 presents the results for several example buildings with various degrees of irregularity in the floor plan and layout of the openings. The models are tested using modal analyses. In the final Sect. 5, we present the conclusions of this study and outline future developments.

2 Methodology

Our end-to-end pipeline for automatically generating solid FE models and EFMs is depicted in Fig. 2. Our work builds on state-of-the-art computer vision and machine learning technologies. As input, our framework uses multiple-view RGB images, which are processed using structure-from-motion (SfM) to encode the building scene as 3D point clouds (structure) and the camera poses where the images were taken (motion). To extract the geometrical features for the finite element models, we post-process the 3D point cloud to produce a polygonal surface model that represents a simplified geometry of the building, as detailed in Pantoja-Rosero et al. (2022). This produces a level of detail (LOD) model, specifically LOD3, which consists of a polygonal surface model that details extrusions and openings (Verdie et al. 2015). Such models use simple primitives (e.g., planes) to approximate the actual geometry (Luebke et al. 2003), and we generate them by post-processing the 3D point clouds, segmenting the semantic information in the 2D images (e.g., openings) and projecting it to 3D. Here, we adapted this previous work by performing extra regularization and scaling it to the real dimensions. We used our generated LOD3 model for the FE models of the outer walls of the masonry buildings, specifically solid finite element models and EFMs. For the solid finite element models, we used the Gmsh (Geuzaine and Remacle 2009) open source finite element mesh generator. For the EFM, we proposed a simple rule-based algorithm based on suggestions from the literature (Quagliarini et al. 2017). As output, we provide ready-to-use geometry files in the format required for the finite element software Amaru (Durand and da Silva 2021) (solid FEmodels), Tremuri (Lagomarsino et al. 2013) (EFMs), and OpenSees (McKenna 2011) (EFM with the macro-element by Vanin et al. (2020)).

2.1 Structure from motion (SfM)

The core of our methodology is photogrammetry, specifically the SfM framework. SfM uses multiple-view images of a scene (i.e., building) to compute a 3D point cloud that shows both point features (structure) and the camera poses where the images where taken (motion) (Szeliski 2021; Hartley and Zisserman 2001). This method's advantages

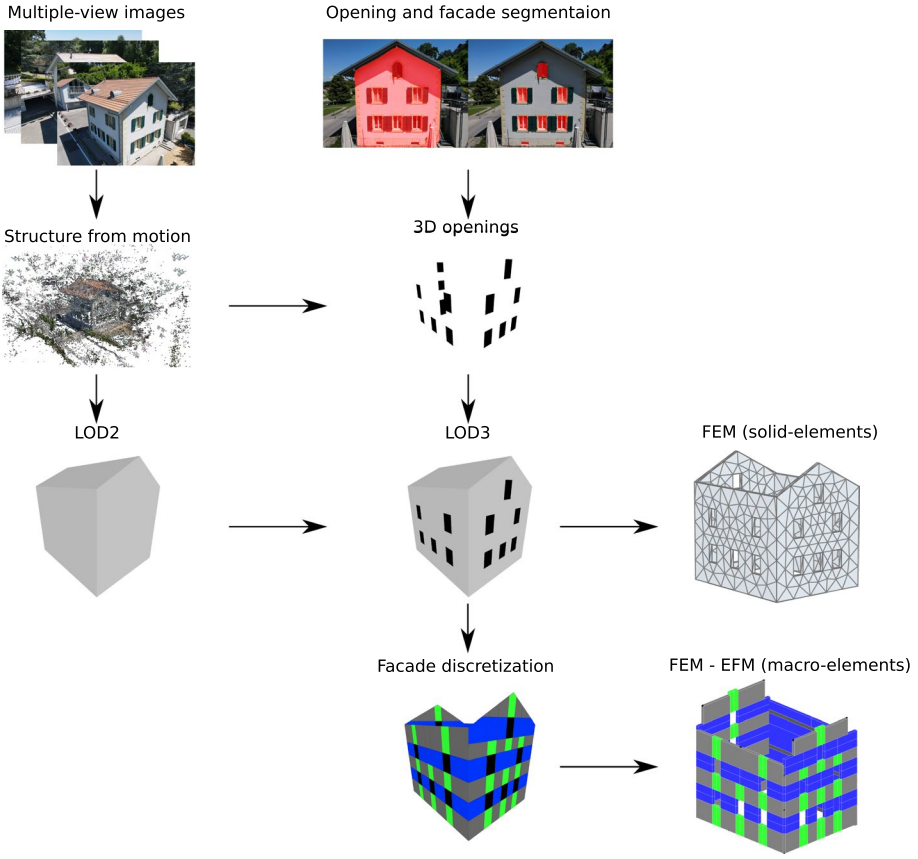

Fig. 2 Pipeline for generating finite element models of freestanding masonry buildings. From top-left to bottom-right: acquire images to generate structure-from-motion (SfM) and regularized LOD2 models; segment and triangulate the openings to 3D; post-process geometric information to generate finite element models with solid- and macro-element approaches

include its low cost (it does not require expensive hardware), accessibility (cameras and software tools are widely available), versatility (it may be used for a variety of applications), and high precision (when appropriate images are taken). On the other hand, some limitations have been noticed, including limited coverage (objects must be seen on photos), limited resolution (model quality is dependent on the quality of the photographs), time-consuming (for huge amounts of image data), and lighting conditions that must be fulfilled.For an example building scene, Fig. 3 shows the images used to generate the SfM model and structure and motion outputs, which were generated using the same open source library MESHROOM (Griwodz et al. 2021) that is also used later to generate the solid finite element models.

Fig. 3 SfM modeling. **a** Some images used for SfM processing. **b** Reconstructed 3D point clouds and camera poses

2.2 LOD2: regularized polygonal surface model

The SfM point cloud is first translated to a polygonal surface model without openings (LOD2 model), wherein the exterior geometry of the building is simplified as a set of planes comprising its facades, its roof planes, and the ground level. To generate this model, we use the *Polyfit* framework proposed by Nan and Wonka (2017) presented in Fig. 4. The input for this methodology is point clouds clustered as plane primitives whose intersections generate candidate faces that might be part of the final model. The faces that will compose the resultant LOD2 model are selected by solving a binary linear optimization problem that accounts for the coverage and fitting of the point cloud over the faces as well as the complexity of the geometry. This problem is formulated using hard constraints to generate a manifold and watertight model. Although this method has proven robustness in the generation of simplified geometries of objects whose main geometry can be represented by plane primitives, it does require the input point cloud to be complete and noise-free, necessitating preprocessing in some cases (e.g., delete points that belong to foreign objects).

Modeling the exterior structural walls of a building with an EFM requires that the walls be vertical. This is not guaranteed when applying the *Polyfit* approach, so we regularized the models with the assumption that the buildings were constructed as per the Manhattan world scenes (Coughlan and Yuille 1999), which assumes most civil

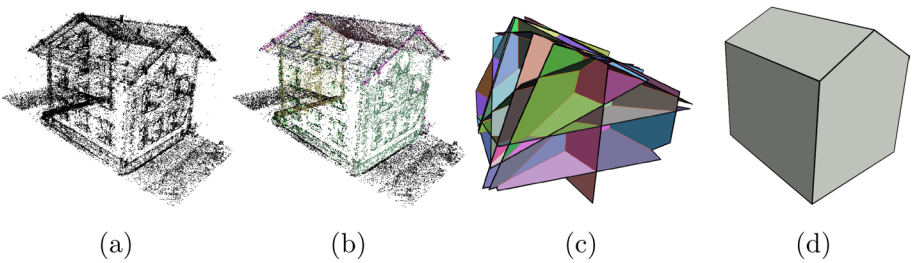

Fig. 4 Polygonal surface models generated using *Polyfit*. **a** Input point cloud. **b** Clustered planar primitives. **c** Candidate faces from plane intersections. (d) Polygonal surface model after optimization process

infrastructure is built following three main perpendicular directions. Note that this does not always apply to historical masonry buildings, so we plan to improve this aspect for future work. The building regularization first defines the three main directions by randomly sampling the LOD2 model as a point cloud and performing a principal component analysis (PCA) (Jolliffe 2002). PCA yields the three main directions, which are approximately aligned with the vertical direction and the normal directions of the walls. The three main directions of the building are then computed as the two mean vectors of normal vectors of the planes identified as facades in the perpendicular directions and their cross product. Once the three main building directions are defined, the building is regularized by: 1) modifying the facade planes such that their normal vector is aligned with one of the two horizontal components of the main directions; 2) replacing the ground plane with a plane that passes through the lowest LOD2 vertex and has a vertical normal vector (note that facade vertices in contact with the ground plane are displaced accordingly). This regularization procedure is depicted in Fig. 5.

2.3 LOD3: mapping opening information to 3D

Analyzing the building via a solid finite element model or EFM requires information on the geometry of the openings. Based on our work presented in Pantoja-Rosero et al. (2022), we first segment openings in the facades in the 2D images inputted to the SfM analysis and then map these openings to the LOD2 using the camera poses. Different from the previous

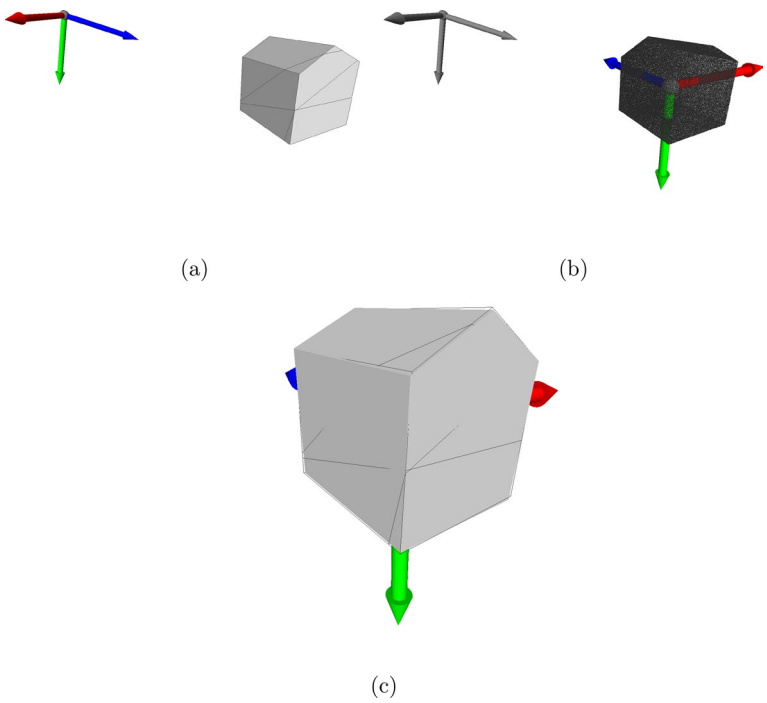

Fig. 5 LOD2 regularization. **a** Irregular LOD2 and global coordinate system. **b** Sampled point cloud and main PCA directions. **c** Regularized LOD2 (filled) overlaying the irregular LOD2 (wireframe)—a regular LOD2 guarantees perpendicularity for the ground and facade planes

work (Pantoja-Rosero et al. 2022), here we first regularize the polygonal surface model and then map the openings using ray-casting instead of two-view triangulation. For the later, we segment the openings and their corners in the 2D images using convolutional neural networks (CNNs). Each opening is represented on the image as a set of four points indicating their vertices **x** that are then mapped over the LOD2 model (finding their 3D corresponding **X**) using ray-casting. This ray-casting was already used in (Pantoja-Rosero et al. 2023) to project damage features (e.g., cracks) onto an LOD3 model. In general, the ray-casting algorithm finds the 3D correspondence **X** of a 2D image point **x** by locating the intersection of a projected ray that passes by the camera center **C** of the view (output of SfM) and **x** with the 3D model (LOD2). This information about the 3D opening can be merged with the LOD2 model to obtain an LOD3 model, which is the basis for the finite element models. This procedure is illustrated in Fig. 6.

Since this method relies on the deep learning technique to segment the requisite 3D mapping openings, it inherits the benefits and drawbacks of such techniques. Despite the fact that the results in general demonstrate excellent accuracy, flexibility, speed, and automation, it is necessary for training to utilize computationally intensive huge data sets.

2.4 Model initialization

The SfM pipeline produces similarity reconstructions (typically called Euclidean reconstructions) (Hartley and Zisserman 2001), meaning that relative distances and angles are preserved but do not represent the absolute dimensions. Furthermore, the orientation and

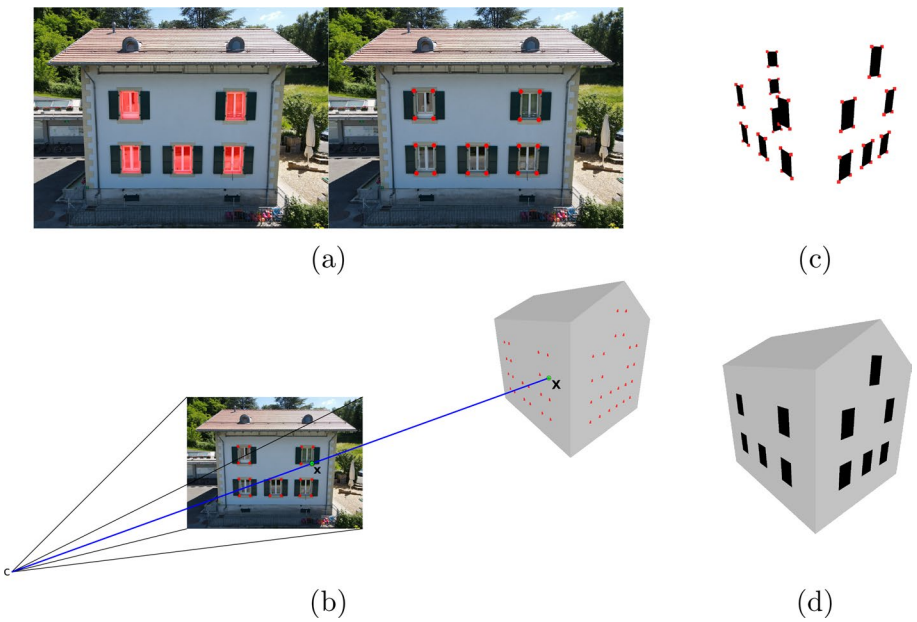

Fig. 6 LOD3 model—mapping information about openings to 3D. **a** Segmentation of openings and their corners using CNNs. **b** Ray-casting the points of the 2D image **x** that correspond to opening corners to obtain 3D correspondences **X**. **c** Openings in 3D formed by their points **X**. **d** LOD3 model with surface and opening information

position of the coordinate system is aleatory. Therefore, before generating FE models, the LOD3 model is *initialized* by applying a similarity transformation that rotates, translates, and scales the model to place it at the origin in alignment with the main directions of the building and scaled to the real dimensions. In homogeneous coordinates, this transformation can be represented as:

$$\mathbf{V}_{init} = \mathbf{T}(\mathbf{R}, \mathbf{t}, s)\mathbf{V}, \tag{1}$$

where $\mathbf{V}$ and $\mathbf{V}_{init}$ are the coordinates of the vertices of the model before and after initialization, and $\mathbf{T}(\mathbf{R}, \mathbf{t}, s)$ is the 4x4 similarity transformation matrix defined as

$$\mathbf{T}(\mathbf{R}, \mathbf{t}, s) = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\intercal} & 1 \end{bmatrix}, \tag{2}$$

with $\mathbf{R}$, $\mathbf{t}$, and s respectively representing a 3x3 rotation matrix, a 3x1 translation vector, and a scaling factor. In our algorithm, we first use the main LOD2 directions located using PCA to find the transformation matrix that represents rotation and translation and its lowest ground point (sect. 2.2). This transformation matrix, which places the LOD model at the origin and aligns it with the main directions of the building, is computed solving the equation:

$$\mathbf{T}(\mathbf{R}, \mathbf{t}) = \mathbf{P}_{init}\mathbf{P}^{-1}, \tag{3}$$

where $\mathbf{P}$ and $\mathbf{P}_{init}$ are 4x4 matrices wherein each column represents a point (in homogeneous coordinates) of the LOD model and the origin of the coordinate system. Specifically, $\mathbf{P}$ columns are formed by: $\mathbf{P}[:, 1]$ representing the lowest LOD2 vertex of the ground plane; $\mathbf{P}[:, 2]$ representing a point placed at unit distance from $\mathbf{P}[:, 1]$ in the same direction as the main vertical LOD2 direction; and $\mathbf{P}[:, 3]$ and $\mathbf{P}[:, 4]$ points that are placed at a unit distance from $\mathbf{P}[:, 1]$ in the same direction as the other two main LOD2 directions. At the same time, the $\mathbf{P}_{init}$ columns represent the four points where the $\mathbf{P}$ points should be placed after the transformation. These points are: $\mathbf{P}_{init}[:, 1] = [0, 0, 0, 1]^{\intercal}$ representing the origin; $\mathbf{P}_{init}[:, 2] = [0, 0, 1, 1]^{\intercal}$ representing the point at a unit distance from the origin in the z direction; $\mathbf{P}_{init}[:, 3] = [1, 0, 0, 1]^{\intercal}$ representing a point at a unit distance from the origin in the x direction; and $\mathbf{P}_{init}[:, 4] = [0, 1, 0, 1]^{\intercal}$ representing a point at a unit distance from the origin in the x direction. To compute the scaling factor, the user needs to provide two image points $\mathbf{x}$ and the distance d between them. The ray-casting algorithm finds the corresponding $\mathbf{X}$ over the LOD2 model and then computes their distance D. With this information, the scaling factor is obtained as $s = d/D$. The initialization of the model is presented in Fig. 7.

2.5 Generating a finite element model with solid elements

One of the outputs produced by our framework is a finite element model that uses solid elements. This is produced by applying the Gmsh finite element mesh generator (Geuzaine and Remacle 2009) to the geometrical information encapsuled in the *initialized* LOD models and 3D openings. As a first step in this process, all the planes labeled as facades (sect. 2.2) are built as surface entities of dimension 2 by joining their vertices (entities of dimension 0) with lines (entities of dimension 1). To next generate volume entities of dimension 3, an extrusion operation is applied to these facade surfaces to simulate the wall thickness *th* (user input). This same technique is also used to generate volume entities for

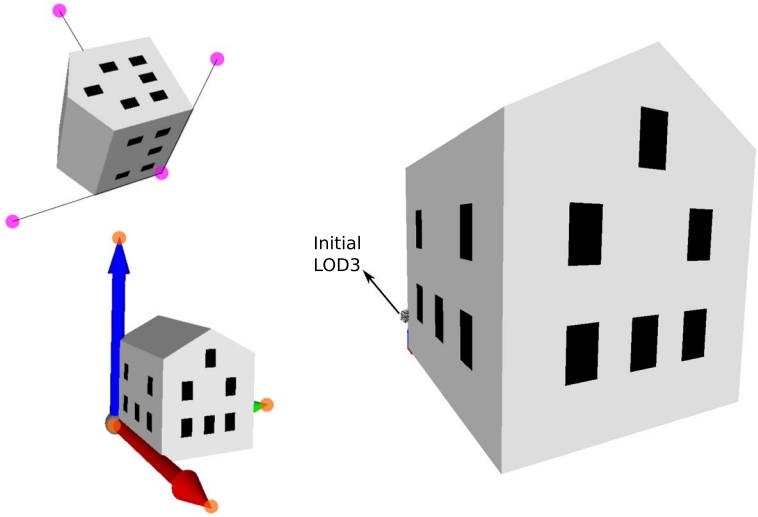

Fig. 7 Model initialization. LOD models and opening coordinates are initialized by placing them at the origin, aligning them with the main building directions, and scaling them to the real dimension. **a** LOD3 models before and after rotation and translation (**P** and $\mathbf{P}_{init}$ points shown in magenta and orange, respectively). **b** Initial (small wireframe) and final LOD3 (filled) models after full initialization (rotation, translation, and scaling)

all the building openings. Finally, the opening volumes are subtracted from the facade volumes using a Boolean difference operation to generate a physical group that the mesh generator discretizes into solid elements. The type of element, its order, and its size are defined by the user. Figure 8 presents the procedure for generating a finite element model for a building using first-order tetrahedral elements.

2.6 Generating an EFM

As outlined in the introduction, EFM requires the discretization of each building facade into deformable parts (piers or spandrels) connected by rigid nodes. To achieve this aim, we coded for regular openings using the rule by Dolce (1991). To adapt the code for irregular opening layouts, we designed a general first step for defining the layout of the cells, which we then label as openings, piers, spandrels, or nodes (Fig. 9). The first part of the algorithm generates the geometry of the polygonal cells in each facade (in 2D local coordinates) as follows:

- Step 0: Find the lines that represent the facade contour (contour lines) and openings (opening lines).
- Step 1: Compute all of the intersections of the facade lines.
- Step 2: Filter the intersections: (1) leave those that lay inside the facade or on the contour; (2) remove intersections produced by a contour line and an opening line that lay inside the facade; (3) remove intersections produced by two contour lines that are not part of the facade contour vertices.

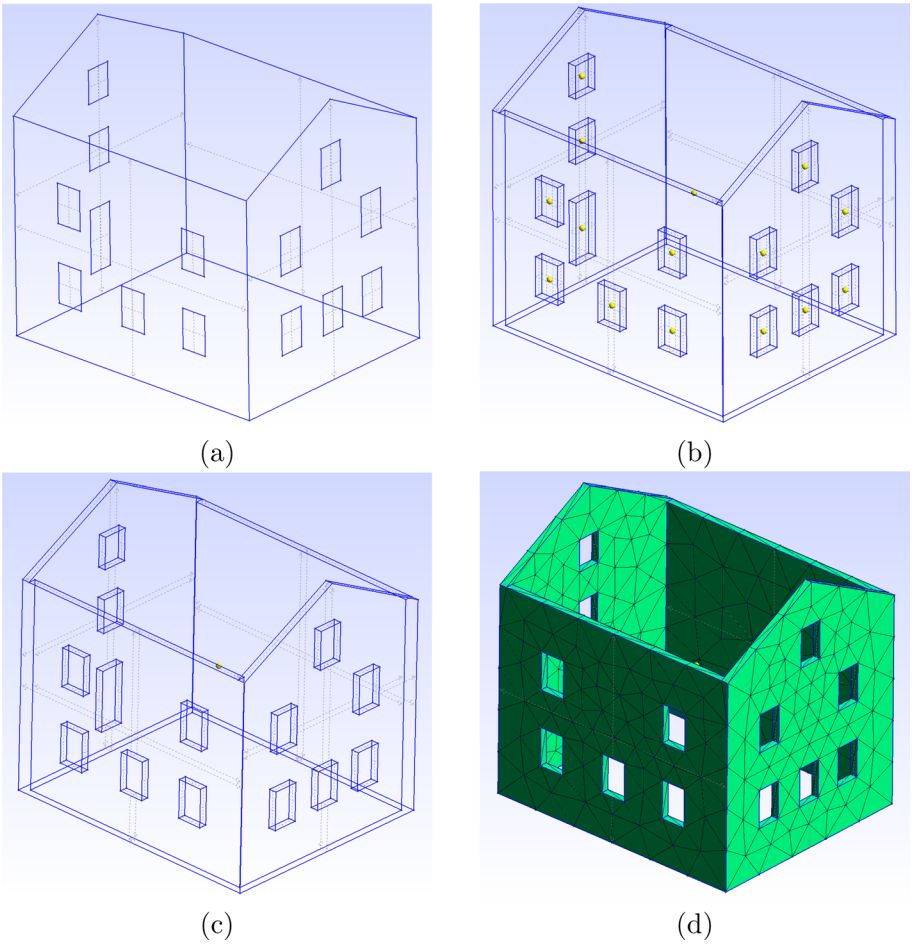

Fig. 8 Finite element model with solid elements generated using GmshAPI (Geuzaine and Remacle 2009). **a** Facades and openings as surface entities. **b** Facades and openings as volume entities after extruding the wall width. **c** Physical group used to generate the solid-element mesh after performing binary operations. **d** Finite element model generated using tetrahedral solid elements

- Step 3: Create line segments by connecting the consecutive points that lay in each of the facade lines.
- Step 4: Produce a graph in which the nodes are formed by the filtered intersections and the edges are formed by the line segments.
- Step 5: Find the polygonal cells formed by the graph using the graph–cycles-based algorithm proposed by Ferreira et al. (2003).

The second part of the algorithm is a rule-based procedure that labels the polygonal cells as openings (o), piers (p), spandrels (s), and nodes (n). Here, we offer the user two options, which yield the same results for a regular layout of openings but different results for irregular layouts. Approach A leads to longer piers while Approach B leads

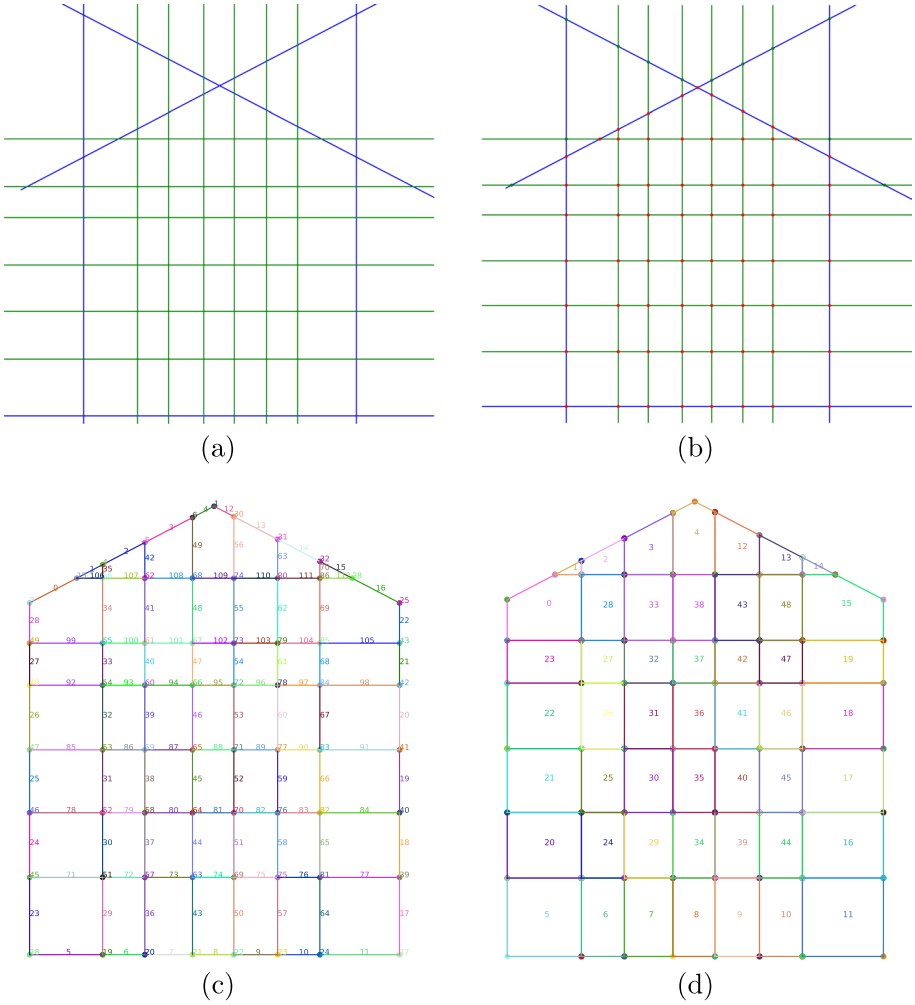

Fig. 9 Generation of polygonal cells in the building facades that will be labeled as openings, piers, spandrels, and nodes. **a** Lines representing the facade contour and openings. **b** Intersections of facade lines, where red points correspond to filtered intersections. **c** Graph formed by filtered intersections and line segments. (**d**) Polygonal cells to be labeled

to deeper spandrels. The process applied to one building facade is presented in Fig. 10, and the results for all facades of the example building are presented in Fig. 11. The labeling algorithm is as follows:

- Step 0: For each cell, determine the eight neighboring cells: top (t), bottom (b), left (l), right (r), top-left (tl), top-right (tr), bottom-left (bl), bottom-right (br).
- Step 1: Label as openings all the cells whose mean coordinate lays inside the LOD openings.
- Step 2: Label as piers the cells to the left (l) and right (r) of the labeled openings. Label as spandrels the cells above (t) and below (b) the labeled openings.

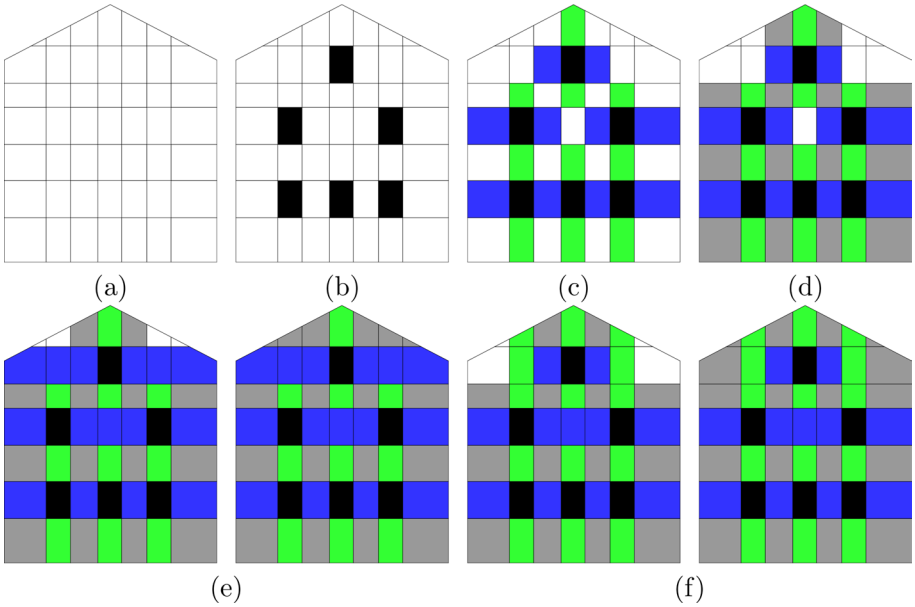

Fig. 10 Facade discretization process. **a** Facade with unlabeled polygonal cells. **b** Facade polygonal cells after labeling step 1. **c** Facade polygonal cells after labeling step 2. **d** Facade polygonal cells after labeling step 3. **e** Approach A: Facade polygonal cells after labeling steps 4 and 5. **f** Approach B: Facade polygonal cells after labeling steps 4 and 5

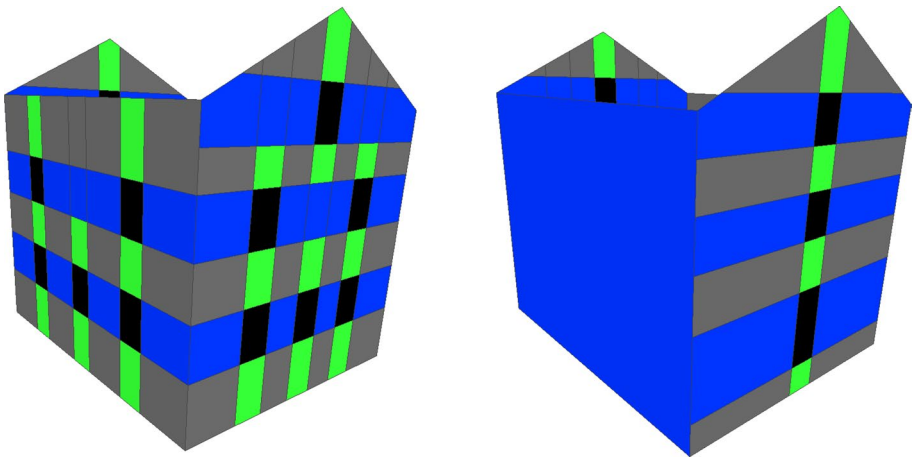

Fig. 11 3D results of the facade discretization as seen from two views

- Step 3: Label as nodes the neighboring cells of the labeled openings in the *tl*, *tr*, *bl*, and *br* positions.

- Step 4: Grow horizontally the cells labeled as piers and grow vertically the cells labeled as spandrels. This item has two approaches. Approach A: longer piers; approach B: deeper spandrels.

  Approach A Loop through all the polygonal cells. If a cell is labeled as a pier, also sequentially label as a pier all the unlabeled cells to its left and right until an already-labeled cell is found. Loop again through all the polygonal cells. If a cell is labeled as a spandrel, also sequentially label as a spandrel all the unlabeled cells above and below until an already-labeled cell is found.
  Approach B Loop through all the polygonal cells. If a cell is labeled as a pier, also sequentially label as a pier all the unlabeled cells to its left and right until the t or b neighbor of the unlabeled cell is a spandrel (without labeling this cell). Loop again through all the polygonal cells. If a cell is labeled as a spandrel, also sequentially label as a spandrel all the unlabeled cells above and below until the l and r neighbors of the unlabeled cell are piers (labeling this cell as pier).

- Step 5: Loop through all the polygonal cells. If a cell is labeled as a node, also sequentially label as a node all the unlabeled cells to its left and right until reaching an already-labeled cell. Loop again through all the polygonal cells. If a cell is labeled as a node, also sequentially label as a node all the unlabeled cells above and below until reaching an already-labeled cell.

Note that our framework was designed to model only piers and spandrels as macro-elements. The use of gable elements, such as the one used by Vanin et al. (2020), should improve the discretization and will be considered in future work. Similarly, we will consider implementing variable pier lengths depending on the loading direction, as has been suggested by several research groups (e.g., Quagliarini et al. (2017)).

2.7 Meshing the EFM

The last part of our end-to-end pipeline is an algorithm that transforms the discretized facade into a mesh for an EFM. The EFM consists of a wire frame composed of special frame elements (macro-elements) defined by their end points (structural nodes), center point, and geometric dimensions. We assume that the cross sections of all piers and spandrels are rectangular. The finite element mesh we generate here matches the rules set by Tremuri Lagomarsino et al. (2013), which can be used either with the original nonlinear beam formulation or the macro-element by Penna et al. (2014). The same mesh is also adopted when using the macro-element by Vanin et al. (2020) in OpenSees McKenna (2011). In our mesh, structural nodes are defined based on the polygonal cells labeled as nodes and on the connectivity between each pair of walls. The macro-elements (piers and spandrels) are then defined based on their connection with neighboring nodes. The process followed by the algorithm is shown in Fig. 12, and the results for all the facades are shown in Fig. 13. The algorithm follows these steps:

- Step 0: Cluster the macro-element cells. Neighboring polygonal cells that have the same label are assigned to a single element (pier, spandrel, or node). If the cells that compose the element produce a non-rectangular shape, the neighboring cells that have different labels are re-labeled to guarantee rectangular elements.

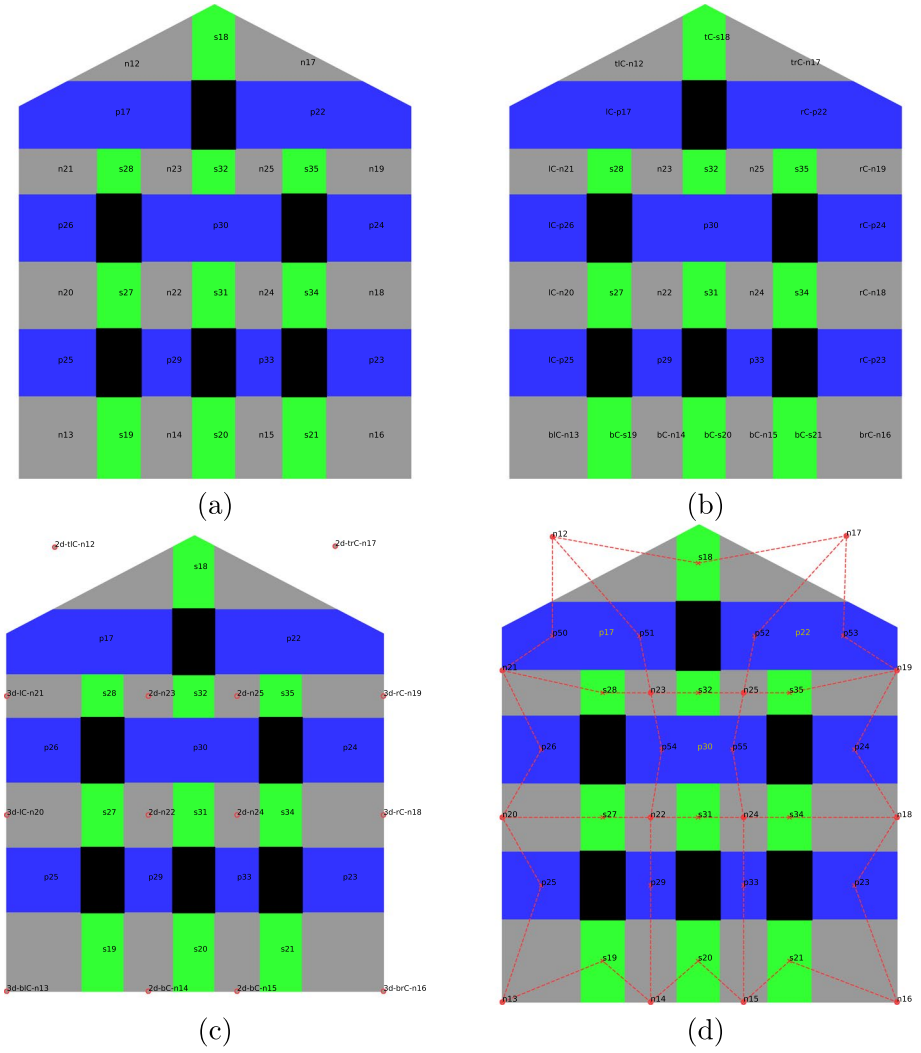

Fig. 12 Generating the geometric components of an equivalent frame model (EFM). **a** Step 0: Neighboring polygonal cells with same label (piers, spandrels, and nodes) are clustered into a single facade element. **b** Step 1: Facade elements labeled as *contour* or *non − contour* elements at *t*, *b*, *l*, *r*, *tl*, *tr*, *bl*, *br* positions. **c** Steps 2–4: 3D and 2D nodes and the position of the structural node point. **d** Steps 5,6: Representation of the finite macro-element as wire elements that connect to the final position of the structural nodes

- Step 1: Label facade elements (nodes, spandrels, and piers) as *contour* or *non − contour*. Assign an extra label to those placed at the contour as top (*t*), bottom (*b*), left (*l*), right (*r*), top-left (*tl*), top-right (*tr*), bottom-left (*bl*), bottom right (*br*).
- Step 2: Define whether nodes are 2D or 3D and find their structural node coordinates. Because TREMURI (Lagomarsino et al. 2013) captures only the in-plane and not out-of-plane behavior of the piers and spandrels, the elements representing the piers and spandrels only have degrees of freedom in the plane of the façade. As a result, TREMURI distinguishes between 3D nodes, which are those that are on the vertical

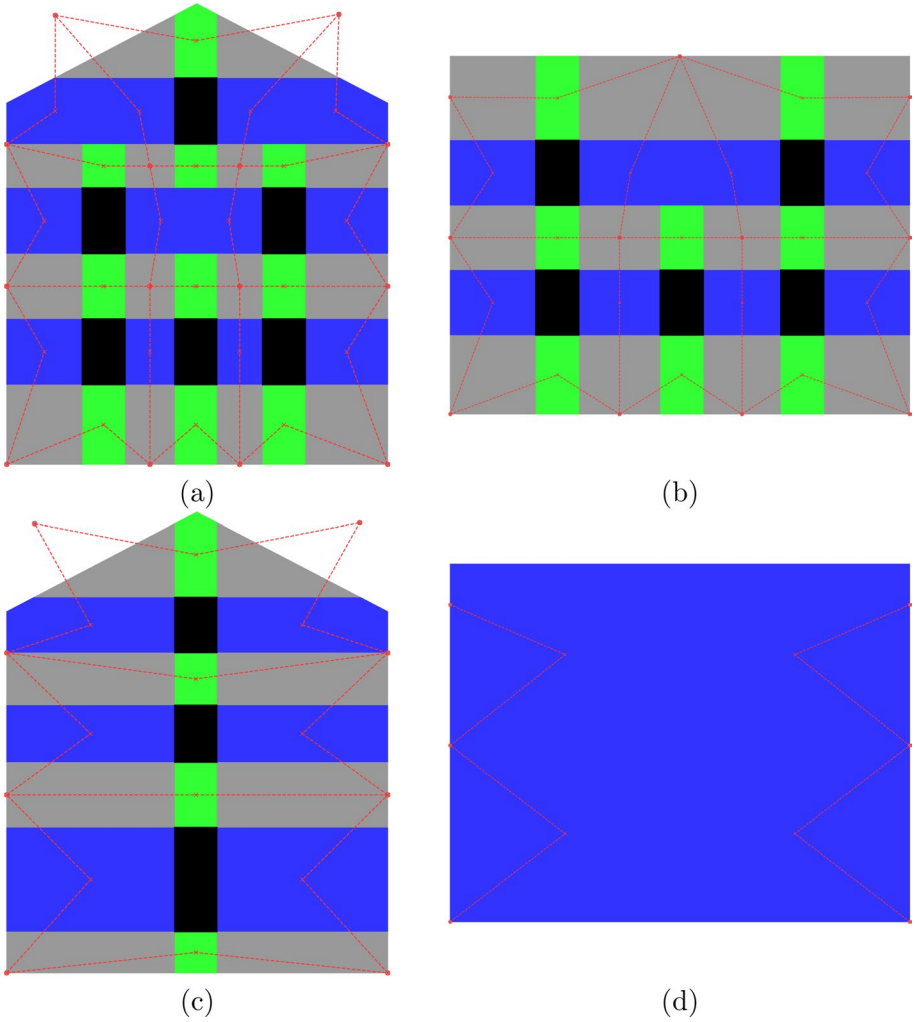

Fig. 13 Geometric components of EFM for the four facades that comprise the building. **a** Facade 1. **b** Facade 2. **c** Facade 3. **d** Facade 4

edges of the contour and therefore in contact with other facades, and all other nodes, which are considered 2D nodes. The structural nodes are placed on the bounding box (*bbx*) that surrounds each node element and its type. *non − contour* node elements: middle point of *bbx*. *t*: middle-top point of *bbx*. *b*: middle-bottom point of *bbx*. *l*: middle-left point of *bbx*. *r*: middle-right point of *bbx*. *tl*: top-left point of *bbx*. *tr*: top-right point of *bbx*. *bl*: bottom-left point of *bbx*. *br*: bottom-right point of *bbx*.

- Step 3: Determine the center point of the pier and spandrel elements.
- Step 4: Place extra structural nodes (without mass) at the ground level of the model beneath each pier element. The new nodes are placed below *contour bl*, *b*, and *br* piers. *b*: middle-bottom point of *bbx*. *bl*: bottom-left point of *bbx*. *br*: bottom-right point of *bbx*.

- Step 5a: Define pier and spandrel end points n_i and n_j for facades with openings. For each pier, n_i and n_j are the neighboring nodes to the bottom and top, respectively. If there are multiple neighboring nodes at the top and/or bottom, the pier is split into several piers according to the number of nodes. For each spandrel, n_i and n_j are the neighboring nodes to the left and right, respectively. If there are multiple neighboring nodes to the left and/or right, the spandrel is split into several spandrels according to the number of nodes.
- Step 5b: Define piers for facades without openings. When there are no openings in a facade, the pier element is split horizontally into two. To do this, two additional structural nodes (without mass) are created at the top-left and top-right of the *bbx* surrounding the pier element. Then, n_i and n_j represent the bottom-left and top-left nodes for the first pier, and bottom-right and top-right for the second. If the facade is connected to another facade containing more than one row of openings in its layout, the pier corresponding to this connection is split according to the number of extra nodes in the neighboring facade. Every time a pier is split, rigid nodes are generated to link the new piers to guarantee a proper connectivity within the wall.
- Step 6: Merge corresponding 3D nodes between facade pairs. Two facades are connected through their 3D nodes by merging the closest 3D nodes to the two facades into one new coordinate equal to the mean of their individual coordinates. If the new coordinate lays outside one of the node element regions, the new coordinate is changed to the largest or lowest point of that node element. If the number of opening rows from one facade and the next do not match, there will be more 3D nodes in the facade with the highest number of row openings. In this case, the extra nodes are shared between the facades, and a rigid link is created to connect each extra node to its closest node. When one of the facades does not have openings, as described in the previous step, the piers of that facade are split at the position of the 3D nodes of a neighboring facade that has more than one row of openings in its layout.
- Step 7: Create finite element geometry files for Tremuri (Lagomarsino et al. 2013) and OpenSees (McKenna 2011) considering rectangular geometry for the elements, structural node placements, and frame element connections. As boundary conditions, all the nodes placed at the ground level are constrained to have no displacements on all degrees of freedom (Fig. 14).

2.8 Metrics and performance assessment

A general rule that works well for regular openings states that wall regions with openings above and below correspond to spandrels and wall regions with openings left and right correspond to piers. For facades with irregular opening layouts, this procedure is subjective, and the discretization relies on the stated criteria as well as the experience of the engineer. Often, a group of engineers discuss the definition of a model that could best represent the expected behavior of the building. We attempted to reproduce the discretization of experienced engineers with the algorithm that we introduced in this section, but the proposed solution remains only one of many. To quantitatively assess the performance of our framework for modeling masonry buildings using macro-elements in light of this subjectivity, we propose two metrics for its evaluation by experienced engineers (*evaluator*). The two metrics are as follows:

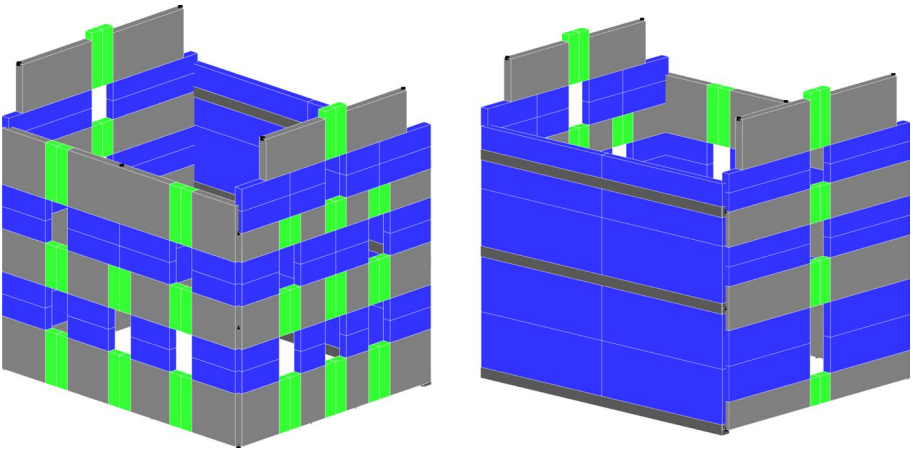

Fig. 14 3D results for the EFM seen from two views

Metric A (*MA*): For this metric, the *evaluator* was given images of building facades containing unlabeled polygonal cells (Fig. 15a). The *evaluator* assigned each unlabeled cell as what she/he considered to be the most reasonable label. The cell labels given by the *evaluator* were compared with the labels automatically assigned by our algorithm. *Metric A* is the percentage of cells that were labeled in the same manner.

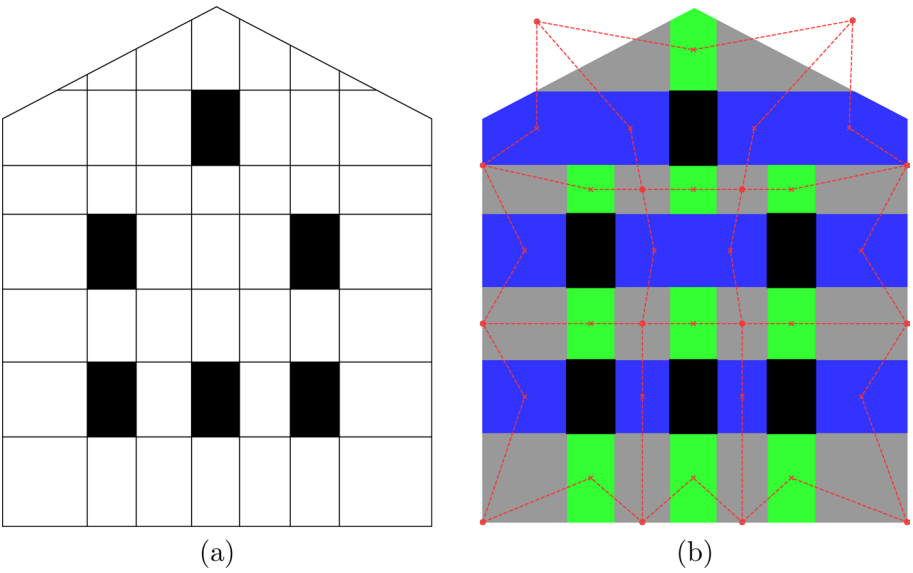

Fig. 15 Images given to the *evaluator* for assessing our methodology. **a** Unlabeled polygonal cells of a facade. **b** Building facades with the areas defined as elements (piers, spandrels, and nodes) with wire-frames that represent the geometrical components of the EFM

Metric B (*MB*): For this metric, the *evaluator* was asked to give a grade between 1.0 to 4.0 to the mesh of the EFM produced by our framework (Fig. 15b). The grades were assigned as follows:

1. Poor: the model makes no sense. Totally misaligned with the evaluator's criteria.
2. 2 Sufficient: the model might produce acceptable results and can be used for numerical analyses. Partially misaligned with the evaluator's criteria.
3. 3 Good: the model might produce accurate results and can be used for numerical analyses. Almost completely aligned with the evaluator's criteria.
5. 4 Excellent: the model should produce accurate results and can be used for numerical analyses. Completely aligned with the evaluator's criteria.

3 Experiments

In this section, we present the results of applying our framework to 12 example free-standing historical masonry buildings in Switzerland or Croatia, as published by Pantoja-Rosero et al. (2022), Pantoja-Rosero et al. Pantoja-Rosero et al. (2023). The buildings in Croatia were damaged in the 2020 Zagreb or Petrinja earthquakes (Atalić et al. 2021; Miranda et al. 2021), which fits well with a proposed application of this method for the numerical modeling of earthquake-damaged buildings. For each building, the input data set consisted of multiple-view images. We first present various models generated within the pipeline, starting with the SfM up to the finite element model with solid elements and the EFM. For the EFMs, we also show the scores assigned by four engineers with experience in modeling masonry buildings. To perform modal analyses of the solid element models and the EFMs, we used the Amaru (Durand and da Silva 2021) and OpenSees (McKenna 2011) finite element libraries, respectively. Some of the deformed shapes of this modal analysis are presented in the last section.

3.1 Finite element geometry

Figures 16 and 17 present some of the models obtained from our framework, focusing on non-damaged and damaged buildings, respectively. Across both images, column A shows the photogrammetry output from the meshroom software, which is a textured mesh. Column B shows the SfM model, which consists of a sparse point cloud and registered camera poses. Next, the polygonal surface model is displayed as an LOD3 model in column C. The finite element geometries using solid elements and EFMs are split by discretization approach, with Approach A presented Fig. 16 and Approach B presented in Fig. 17. In addition to the visual results, the four scores computed for each building and discretization approach are presented in Table 1 ($MA-A$: Metric A; discretization approach A. $MA-B$: Metric A; discretization approach B. $MB-A$: Metric B; discretization approach A. $MB-B$: Metric B; discretization approach B). The metrics were computed based on the assessments of four engineers, who are experienced in modeling masonry buildings using EFMs as described in Sect. 2.8.

Table 1 shows that the MA scores average over 90% (metric mean values - $MA-A$: 92.7%; $MA-B$: 90.6%) and MB scores, with one exception, are equal to or larger than 2.0 (metric mean values - $MB-A$: 3.2; $MB-B$: 3.1). In general, the experts were satisfied with the performance of the methodology for all example

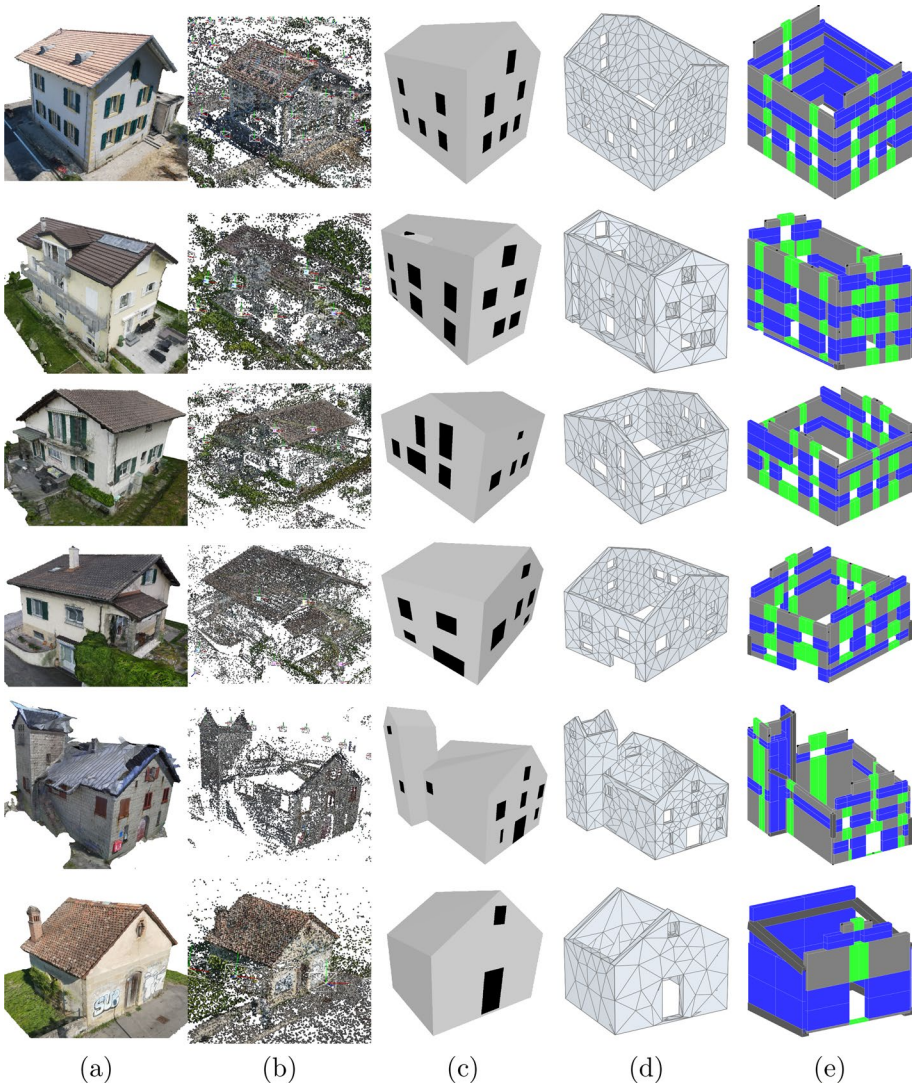

Fig. 16 Models generated by our framework for non-damaged buildings. **a** Photogrammetry textured mesh. **b** SfM model as point cloud and camera poses. **c** LOD3 models containing simplified surface and opening information. **d** Finite element model using solid-elements (tetrahedral elements of the first order). **e** Finite element model using macro-elements with discretization **Approach A**

buildings studied, with the framework producing appropriate discretizations and definition in the EFMs for numerical simulations of the building. The only disagreement with the experts stemmed from personal criteria in the discretization process, which they considered to be expected and normal. Their most relevant comments for improvement were: 1) the systematic methodology sometimes produces overly large and rigid nodes—their size should be reduced and adjacent piers or spandrels should be increased; 2) small openings should be disregarded to avoid creating too many rigid

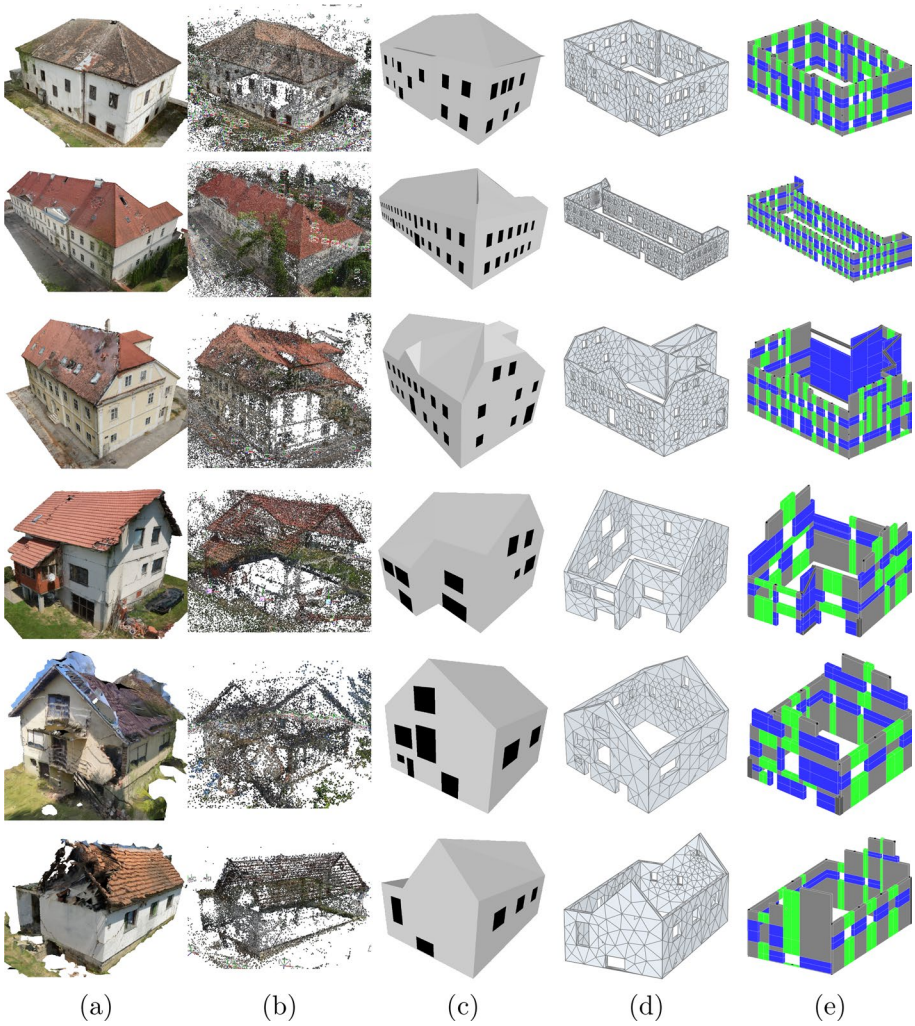

Fig. 17 Models generated by our framework for damaged buildings. **a** Photogrammetry textured mesh. **b** SfM model as point cloud and camera poses. **c** LOD3 models containing simplified surface and opening information. **d** Finite element model using solid-elements (tetrahedral elements of the first order). **e** Finite element model using macro-elements with discretization **Approach B**

nodes. Improvements along these directions will be considered in the future (see Conclusions). Overall, these results show the clear potential for the use of image-based approaches in automatically generating finite element geometries of the facades of historical masonry buildings.

Table 1 Metric values for assessing the generation of EFMs using our framework (MA-A: Metric A%; discretization approach A. MA-B%: Metric A; discretization approach B. MB-A: Metric B; discretization approach A. MB-B; Metric B, discretization approach B)

		Non-damaged buildings (NB)						Damaged buildings (DB)					
		*NB*1	*NB*2	*NB*3	*NB*4	*NB*5	*NB*6	*DB*1	*DB*2	*DB*3	*DB*4	*DB*5	*DB*6
Evaluator 1	*MA*-A	98.9	89.7	96.3	93.0	89.6	91.7	96.7	–	92.5	93.8	92.7	90.6
	MA-B	92.4	79.4	90.8	91.7	88.2	91.7	95.8	99.9	87.5	90.6	92.7	87.1
	MB-A	3.8	3.0	2.5	2.5	2.3	4.0	3.4	–	3.4	3.2	2.8	2.3
	MB-B	3.3	2.5	2.5	2.8	2.4	4.0	3.5	3.8	3.4	3.0	3.3	2.3
Evaluator 2	*MA*-A	98.9	84.6	98.2	91.7	92.4	100.0	97.1	–	92.2	92.2	91.9	89.4
	MA-B	92.4	82.4	89.9	90.4	87.5	100.0	96.3	99.4	88.4	90.6	91.9	85.9
	MB-A	4.0	3.0	3.0	2.8	3.2	4.0	3.8	–	3.3	2.7	2.6	3.0
	MB-B	3.8	2.8	3.0	2.8	3.2	4.0	3.8	3.5	3.3	2.7	2.8	2.9
Evaluator 3	*MA*-A	95.7	86.0	88.1	93.6	86.8	100.0	96.3	–	97.8	95.3	87.9	87.1
	MA-B	93.5	89.0	89.0	93.6	86.1	100.0	93.8	100.0	97.5	95.3	87.9	84.7
	MB-A	3.0	2.0	3.0	3.0	4.0	4.0	3.0	–	4.0	3.0	3.0	3.5
	MB-B	3.0	2.0	3.0	3.0	4.0	4.0	3.0	4.0	4.0	3.0	3.0	3.5
Evaluator 4	*MA*-A	94.6	88.2	88.1	89.2	91.0	95.8	97.1	–	95.3	84.4	87.9	98.8
	MA-B	88.0	77.9	77.1	89.2	91.0	95.8	90.4	96.0	90.0	82.8	87.9	88.2
	MB-A	4.0	3.0	4.0	3.5	2.0	4.0	4.0	–	3.5	3.0	3.0	2.5
	MB-B	3.0	2.0	3.0	3.0	2.0	4.0	3.0	3.5	2.5	2.5	3.0	1.0

3.2 Modal analysis

To evaluate the performance of the models in the elastic domain, we performed modal analyses using two different open source software, Amaru (Durand and da Silva 2021) and OpenSees (McKenna 2011), for the solid-element model and the EFM, respectively. For the material properties, we chose values representative of stone masonry. As our intention with this analysis is only to present the functionality of the models, details of the analysis, such as material properties and input and output information, are not discussed in detail, and we refer the reader to the dataset published together with this work. The results consisting of some of the deformed shapes obtained after performing such simulations are presented in Figs. 18 and 19.

4 Discussion

Our findings have significant implications for future structural engineering and construction research. Using image-based methodologies to generate finite element geometries offers the potential to automate the structural modeling procedure. This could result in substantial time and cost savings when analyzing existing buildings. Moreover, the methodologies described in this study could be combined with other developing technologies, including machine learning and artificial intelligence, to enhance the precision and efficiency of structural modeling of damaged structures. For instance, the

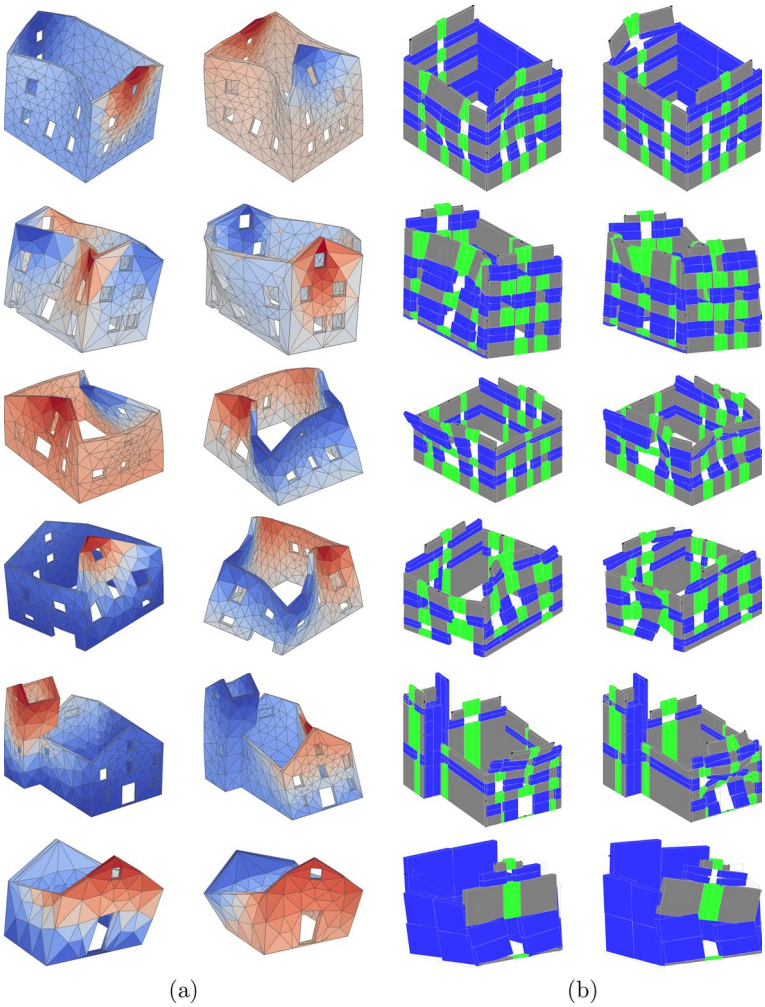

Fig. 18 Deformed shapes output by modal analyses using the FE models generated by our framework as the input geometry—non-damaged buildings. **a** Two deformed shapes obtained using solid-elements in Amaru Durand and da Silva (2021). **b** Two deformed shapes using macro-elements in OpenSees McKenna (2011)

addition of damage observed on images can be associated to the mechanical properties of damaged material, which can then be incorporated into numerical models.

Advanced technologies enable faster and more accurate data collection and analysis, allowing the identification of potential structural issues before they become critical. The use of drones, remote sensors, and other cutting-edge technologies, for instance, can provide important details about a structure's condition, including any damage or deterioration. This data can be used to generate more precise structural models, which can be used to simulate various scenarios and identify potential vulnerabilities. This information can be used to guide decisions on maintenance, repair, and retrofitting, and to ensure that structures are better prepared to withstand natural disasters.

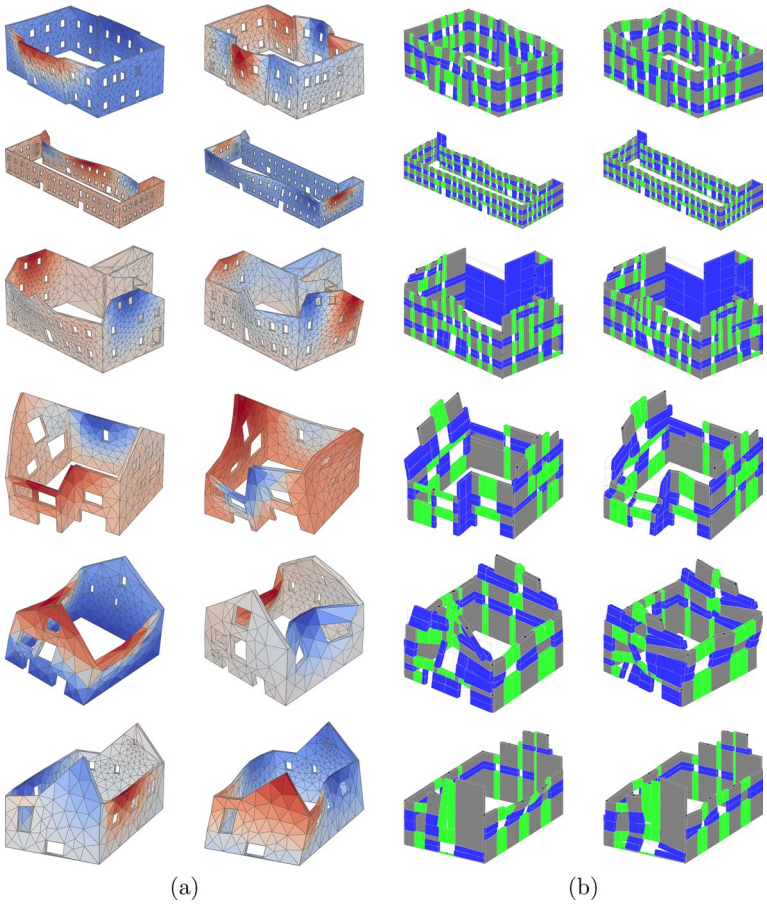

Fig. 19 Deformed shapes output by modal analyses using the FE models generated by our framework as the input geometry—damaged buildings. **a** Two deformed shapes obtained using solid-elements in Amaru Durand and da Silva (2021). **b** Two deformed shapes using macro-elements in OpenSees McKenna (2011)

5 Conclusions

The objective of this study was to contribute to the numerical modeling of historical masonry buildings by providing an end-to-end framework that automatically generates finite element meshes of facade walls for solid-element models and EFMs. This automated approach is an alternative to the manual operation, in which engineers draw the geometry of the building using CAD software. The only input required by our methodology is multiple-view images of the building, which are processed using SfM to encode the geometrical information of the building as a point cloud with camera poses. Convolutional neural networks extract information about openings and their corners, which together with the point clouds can produce the LOD3 model.

Though the solid element model of the facades can be obtained simply by extruding the facades along the wall thickness, generating EFMs is more challenging. The discretization of a facade into an EFM is still a subjective process that requires the expertise of

experienced engineers on numerical modeling of masonry buildings. Our algorithms were design to aid this process by capturing the overall concepts applied by engineers as an initial generic model that the user can modify. Even so, the results obtained during our experiments were satisfactory to four expert engineers, which motivates further development of our work. To this end, we plan to perform additional investigations in various directions, including the addition of gable elements and other building elements such as roof and floor levels and varying the length of the pier element when the openings to the left and right have different heights or if the pier is at the edge of a facade. Additionally, we will increase the capabilities of the discretization process by allowing the consideration of different literature criteria with available pier and spandrel dimensions. We will also include information from interior point clouds, from which we can retrieve wall thicknesses, interior wall geometries, and spanning directions of timber slabs. Finally, to develop inputs for rapid post-earthquake damage assessments, we plan to develop techniques for assessing the response of damaged buildings after earthquakes for several aftershock scenarios, including uncertainty analyses.

This study demonstrated that automated image-based approaches may produce accurate finite element models for masonry buildings, which can be used for estimating their structural response. This method can considerably reduce the time and effort necessary to build finite element models, which is now a difficult manual procedure using CAD software. By automating this procedure, our system can rapidly simulate several buildings, which is required for some engineering applications such as post-earthquake assessments. Overall, our research emphasizes the capability of automated image-based techniques in creating finite element models for masonry structures. We are confident that our approach can be integrated into the software utilized by civil engineering practitioners.

Acknowledgements Many thanks to the experts on modeling masonry structures Ivana Bozulic, Mathias Haindl, Igor Tomic and Christian Salvatori for the time spent assessing the framework.

Funding Open access funding provided by EPFL Lausanne. This project is partially funded by the Swiss Data Science Center under grant C18–04 (“Towards an automated post-earthquake damage assessment”).

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abu-Haifa M, Lee SJ (2022) Image-based modeling-to-simulation of masonry walls. J Archit Eng 28(4):1–12. https://doi.org/10.1061/(asce)ae.1943-5568.0000569

Angiolilli M, Lagomarsino S, Cattari S, Degli Abbati S (2021) Seismic fragility assessment of existing masonry buildings in aggregate. Eng Struct 247. https://doi.org/10.1016/j.engstruct.2021.113218

Atalić J, Uroš M, Šavor Novak M, Demšić M, Nastev M, (2021) The Mw5.4 Zagreb (Croatia) earthquake of March 22, (2020) impacts and response. Bull Earthq Eng 19(9):3461–3489. https://doi.org/10.1007/s10518-021-01117-w

Augenti N (2006) Seismic behaviour of irregular masonry. Proceedings of the 1st European conference on earthquake engineering and Seismology, 3–8

Baraldi D, Cecchi A (2017) A full 3D rigid block model for the collapse behaviour of masonry walls. Eur J Mech A/Solid 64:11–28. https://doi.org/10.1016/j.euromechsol.2017.01.012

Bassier M, Hardy G, Bejarano-Urrego L, Drougkas A, Verstrynge E, Van Balen K, Vergauwen M (2019) Semi-automated creation of accurate fe meshes of heritage masonry walls from point cloud data. Struct Anal Hist Constr 18:305–314. https://doi.org/10.1007/978-3-319-99441-3

Belmouden Y, Lestuzzi P (2009) An equivalent frame model for seismic analysis of masonry and reinforced concrete buildings. Constr Build Mater 23(1):40–53. https://doi.org/10.1016/j.conbuildmat.2007.10.023

Berto L, Saetta A, Scotta R, Vitaliani R (2002) An orthotropic damage model for masonry structures. Int J Numer Meth Eng 55(2):127–157. https://doi.org/10.1002/nme.495

Beyer K, Dazio A (2012) Quasi-static cyclic tests on masonry spandrels. Earthq Spectra 28(3):907–929. https://doi.org/10.1193/1.4000063

Block P, Lachauer L (2014) Three-dimensional funicular analysis of masonry vaults. Mech Res Commun 56:53–60. https://doi.org/10.1016/j.mechrescom.2013.11.010

Bracchi S, Rota M, Penna A, Magenes G (2015) Consideration of modelling uncertainties in the seismic assessment of masonry buildings by equivalent-frame approach. Bull Earthq Eng 13(11):3423–3448. https://doi.org/10.1007/s10518-015-9760-z

Caliò I, Marletta M, Pantò B (2012) A new discrete element model for the evaluation of the seismic behaviour of unreinforced masonry buildings. Eng Struct. https://doi.org/10.1016/j.engstruct.2012.02.039

Celano T, Argiento LU, Ceroni F, Casapulla C (2021) In-plane behaviour of masonry walls: numerical analysis and design formulations. Materials **14**(19). https://doi.org/10.3390/ma14195780

Chiozzi A, Milani G, Tralli A (2017) A genetic algorithm NURBS-based new approach for fast kinematic limit analysis of masonry vaults. Comput Struct 182:187–204. https://doi.org/10.1016/j.compstruc.2016.11.003

Coughlan JM, Yuille AL (1999) Manhattan World: compass direction from a single image by Bayesian inference. Proc IEEE Int Conf Comput Vis 2:941–947. https://doi.org/10.1109/iccv.1999.790349

Dolce M (1991) Schematizzazione e modellazione degli edifici in muratura soggetti ad azioni sismiche. Ind delle costruzioni 25(242):44–57

Durand R, da Silva FHBT (2021) Three-dimensional modeling of fracture in quasi-brittle materials using plasticity and cohesive finite elements. Int J Fract 228(1):45–70. https://doi.org/10.1007/s10704-021-00514-1

D'Altri AM, de Miranda S, Castellazzi G, Sarhosis V (2018) A 3D detailed micro-model for the in-plane and out-of-plane numerical analysis of masonry panels. Comput Struct 206:18–30. https://doi.org/10.1016/j.compstruc.2018.06.007

D'Altri AM, Sarhosis V, Milani G, Rots J, Cattari S, Lagomarsino S, Sacco E, Tralli A, Castellazzi G, de Miranda S (2020) Modeling strategies for the computational analysis of unreinforced aasonry structures: review and classification. Arch Comput Method Eng 27(4):1153–1185. https://doi.org/10.1007/s11831-019-09351-x

D'Ayala D (2013) Assessing the seismic vulnerability of masonry buildings, pp 334–365. https://doi.org/10.1533/9780857098986.3.334

Fenu L, Colasanti V, Parisi F(2022) Numerical simulation of shaking table test on an adobe masonry building through nonlinear macro-element analysis. Int J Masonry Res Innov, 504–524. https://doi.org/10.1504/IJMRI.2022.125339

Ferreira A, Fonseca MJ, Jorge Ja (2003) Polygon detection from a set of lines. Proceedings of 120 Encontro Português de Computação Gráfica (12th EPCG), 159–162

Fraternali F (2010) A thrust network approach to the equilibrium problem of unreinforced masonry vaults via polyhedral stress functions. Mech Res Commun 37(2):198–204. https://doi.org/10.1016/j.mechrescom.2009.12.010

Funari MF, Hajjat AE, Masciotta MG, Oliveira DV, Lourenço PB (2021) A parametric scan-to-FEM framework for the digital twin generation of historic masonry structures. Sustainability (Switzerland) **13**(19). https://doi.org/10.3390/su131911088

Geuzaine C, Remacle J-F (2009) Gmsh: a 3-D finite element mesh generator with built-in pre- and post-processing facilities. Int J Numer Methods Eng 79(11):1309–1331. https://doi.org/10.1002/nme.2579

Ghezelbash A, Beyer K, Dolatshahi KM, Yekrangnia M (2020) Shake table test of a masonry building retrofitted with shotcrete. Eng Struct 219(January):110912. https://doi.org/10.1016/j.engstruct.2020.110912

Griwodz C, Gasparini S, Calvet L, Gurdjos P, Castan F, Maujean B, Lillo GD, Lanthony Y(2021) AliceVision Meshroom: An open-source 3D reconstruction pipeline. Proceedings of the 12th ACM multimedia systems conference, 241–247. https://doi.org/10.1145/3458305.3478443

Hartley, R., Zisserman, A.: multiple view geometry in computer vision, 2nd edn. (2001). https://www.robots.ox.ac.uk/~vgg/hzbook/

Jolliffe IT (2002) Principal component analysis for special types of data, 2nd edn. Springer, New York . https://doi.org/10.1007/b98835.https://link.springer.com/book/10.1007/b98835

Kassotakis N, Sarhosis V (2021) Employing non-contact sensing techniques for improving efficiency and automation in numerical modelling of existing masonry structures: a critical literature review. Structures 32(March):1777–1797. https://doi.org/10.1016/j.istruc.2021.03.111

Kouris LAS, Penna A, Magenes G (2022) Assessment of a full-scale unreinforced stone masonry building tested on a shaking table by inverse engineering. Buildings **12**(8) . https://doi.org/10.3390/buildings12081235

Kujawa M, Lubowiecka I, Szymczak C (2020) Finite element modelling of a historic church structure in the context of a masonry damage analysis. Eng Fail Anal 107(2019):1–18. https://doi.org/10.1016/j.engfailanal.2019.104233

Lagomarsino S, Penna A, Galasco A, Cattari S (2013) TREMURI program: an equivalent frame model for the nonlinear seismic analysis of masonry buildings. Eng Struct 56:1787–1799. https://doi.org/10.1016/j.engstruct.2013.08.002

Lourenço PB, Rots JG (1997) Multisurface interface model for analysis of masonry structures. J Eng Mech 123(7):660–668. https://doi.org/10.1061/(asce)0733-9399(1997)123:7(660)

Lucidi A, Giordano E, Clementi F, Quattrini R (2021) Point cloud exploitation for structural modeling and analysis: a reliable workflow. Int Arch Photogram Remote Sens Spatial Inf Sci ISPRS Arch 43(B2–2021):891–898. https://doi.org/10.5194/isprs-archives-XLIII-B2-2021-891-2021

Luebke D, Reddy M, Cohen JD, Varshney A, Watson B, Huebner R (2003) Level of Detail for 3D Graphics, pp 185–228 . https://doi.org/10.1016/B978-155860838-2/50009-1.http://www.sciencedirect.com/science/article/pii/B9781558608382500091

Ma P, Yao J, Hu Y (2022) Numerical analysis of different influencing factors on the in-Plane failure mode of unreinforced aasonry (URM) structures. Buildings **12**(2). https://doi.org/10.3390/buildings12020183

Marmo F, Rosati L (2017) Reformulation and extension of the thrust network analysis. Comput Struct 182:104–118. https://doi.org/10.1016/j.compstruc.2016.11.016

Massart TJ, Peerlings RHJ, Geers MGD (2007) An enhanced multi-scale approach for masonry wall computations with localization of damage. Int J Numer Meth Eng 69(5):1022–1059. https://doi.org/10.1002/nme.1799

McKenna F (2011) OpenSees: A framework for earthquake engineering simulation. Comput Sci Eng 13(4):58–66. https://doi.org/10.1109/MCSE.2011.66

Miglietta M, Damiani N, Bracchi S, Guerrini G, Graziotti F, Penna A (2021) Macroelement numerical simulation of the seismic response of a timber-retrofitted masonry pier. In: International Conference on Structural Analysis of Historical Constructions . https://doi.org/10.23967/sahc.2021.148

Milani G, Lourenço P, Tralli A (2007) 3D homogenized limit analysis of masonry buildings under horizontal loads. Eng Struct 29(11):3134–3148. https://doi.org/10.1016/j.engstruct.2007.03.003

Miranda E, Brzev S, Bijelic N, Arbanas Z, Bartolac M, Jagodnik V, Lazarevic D, Mihalic Arbanas S, Zlatovic S, Acosta A (2021) Petrinja, Croatia December 29, 2020, Mw 6.4 earthquake. Joint Reconnaissance Report PRJ-2959, 0–2. https://doi.org/10.3929/ethz-b-000465058

Morandi P, Albanesi L, Graziotti F, Li Piani T, Penna A, Magenes G (2018) Development of a dataset on the in-plane experimental response of URM piers with bricks and blocks. Constr Build Mater 190:593–611. https://doi.org/10.1016/j.conbuildmat.2018.09.070

Morandini C, Malomo D, Penna A (2022) Equivalent frame discretisation for URM façades with irregular opening layouts. Bull Earthq Eng 20(5):2589–2618. https://doi.org/10.1007/s10518-022-01315-0

Nan L, Wonka P (2017) PolyFit: polygonal surface reconstruction from point clouds. In: International Conference on Computer Vision, pp. 2353–2361 . https://doi.org/10.1109/ICCV.2017.258.https://openaccess.thecvf.com/content_iccv_2017/html/Nan_PolyFit_Polygonal_Surface_ICCV_2017_paper.html

Pantoja-Rosero BG, Achanta R, Kozinski M, Fua P, Perez-Cruz F, Beyer K (2022) Generating LOD3 building models from structure-from-motion and semantic segmentation. Autom Constr 141:104430. https://doi.org/10.1016/j.autcon.2022.104430

Pantoja-Rosero BG, Saloustros S, Achanta R (2023) Beyer K (2023) image-based geometric digital twinning for stone masonry elements. Autom Construct 145:104632. https://doi.org/10.1016/j.autcon.2022.104632

Pantoja-Rosero BG, Achanta R, Beyer K (2023) Damage-augmented digital twins towards the automated inspection of buildings. Autom Construct (Under review)

Pantoja-Rosero BG, Achanta R, Kozinski M, Fua P, Perez-Cruz F, Beyer K. Dataset for generating LOD3 building models from structure-from-motion and semantic segmentation. https://doi.org/10.5281/zenodo.6651663.

Pantoja-Rosero BG, Achanta R, Beyer K. Dataset for damage-augmented digital twins towards the automated inspection of buildings. https://doi.org/10.5281/zenodo.7331655.

Parisse F, Cattari S, Marques R, Lourenço PB, Magenes G, Beyer K, Calderoni B, Camata G, Cordasco EA, Erberik MA, İçel C, Karakaya M, Malomo D, Manzini CF, Marano C, Messali F, Occhipinti G, Pantò B, Saygılı Sousamli M (2021) Benchmarking the seismic assessment of unreinforced masonry buildings from a blind prediction test. Structures 31(2021):982–1005. https://doi.org/10.1016/j.istruc.2021.01.096

Penna A, Lagomarsino S, Galasco A (2014) A nonlinear macroelement model for the seismic analysis of masonry buildings. Earthq Eng Struct Dyn 43(2):159–179

Pereira JM, Correia AA, Lourenço PB (2021) In-plane behaviour of rubble stone masonry walls: Experimental, numerical and analytical approach. Construct and Build Mater 271. https://doi.org/10.1016/j.conbuildmat.2020.121548

Petracca M, Pelà L, Rossi R, Oller S, Camata G, Spacone E (2016) Regularization of first order computational homogenization for multiscale analysis of masonry structures. Comput Mech 57(2):257–276. https://doi.org/10.1007/s00466-015-1230-6

Portioli F, Casapulla C, Gilbert M, Cascini L (2014) Limit analysis of 3D masonry block structures with non-associative frictional joints using cone programming. Comput Struct 143:108–121. https://doi.org/10.1016/j.compstruc.2014.07.010

Preciado A, Orduna A (2018) Numerical modeling strategies for the seismic analysis of masonry structures. In: Masonry: Design, Materials and Techniques vol. 1, pp. 55–79. Nova Science publishers, New York . Chap. 3. https://www.researchgate.net/profile/Adolfo-Preciado/publication/330688439_Numerical_Modeling_Strategies_for_the_Seismic_Analysis_of_Masonry_Structures/links/5c4f5aee92851c22a397304a/Numerical-Modeling-Strategies-for-the-Seismic-Analysis-of-Masonry-Struct

Quagliarini E, Maracchini G, Clementi F (2017) Uses and limits of the equivalent frame model on existing unreinforced masonry buildings for assessing their seismic risk: a review. J Build Eng 10(2016):166–182. https://doi.org/10.1016/j.jobe.2017.03.004

Rezaie A, Godio M, Beyer K (2020) Experimental investigation of strength, stiffness and drift capacity of rubble stone masonry walls. Constr Build Mater 251:118972. https://doi.org/10.1016/j.conbuildmat.2020.118972

Rinaldin G, Macorini L, Amadio C (2016) A macro-model with nonlinear springs for seismic analysis of URM buildings. Pac Conf Earthq Eng. https://doi.org/10.1002/eqe

Senaldi IE, Guerrini G, Comini P, Graziotti F, Penna A, Beyer K, Magenes G (2020) Experimental seismic performance of a half-scale stone masonry building aggregate. Bull Earthq Eng 18(2):609–643. https://doi.org/10.1007/s10518-019-00631-2

Serpieri R, Albarella M, Sacco E (2017) A 3D microstructured cohesive-frictional interface model and its rational calibration for the analysis of masonry panels. Int J Solids Struct 122–123:110–127. https://doi.org/10.1016/j.ijsolstr.2017.06.006

Shabani A, Skamantzari M, Tapinaki S, Georgopoulos A, Plevris V, Kioumarsi M (2021) 3D simulation models for developing digital twins of heritage structures: Challenges and strategies. Procedia Structural Integrity 37(C), 314–320 . https://doi.org/10.1016/j.prostr.2022.01.090

Soti R, Abdulrahman L, Barbosa AR, Wood RL, Mohammadi ME (2019) Olsen MJ (2020) Case study: post-earthquake model updating of a heritage pagoda masonry temple using AEM and FEM. Eng Struct 206:109950. https://doi.org/10.1016/j.engstruct.2019.109950

Szeliski R (2021) Computer vision : algorithms and applications vol. 2nd Edition, p 1196. https://szeliski.org/Book

Tomić I, Vanin F, Beyer K (2021) Uncertainties in the seismic assessment of historical masonry buildings. Appl Sci (Switzerland) 11(5):1–36. https://doi.org/10.3390/app11052280

Tomić I, Vanin F, Božulić I, Beyer K (2021) Numerical simulation of unreinforced masonry buildings with timber diaphragms. Buildings **11**(5) . https://doi.org/10.3390/buildings11050205

Vanin F, Penna A, Beyer K (2020) A three-dimensional macroelement for modelling the in-plane and out-of-plane response of masonry walls. Earthq Eng Struct Dynam 49(14):1365–1387. https://doi.org/10.1002/eqe.3277

Vanin F, Penna A, Beyer K (2020) Equivalent-frame modeling of two shaking table tests of masonry buildings accounting for their out-of-plane response. Front Built Environ **6**(42). https://doi.org/10.3389/fbuil.2020.00042

Vasconcelos G, Lourenço PB (2009) Experimental characterization of stone masonry in shear and compression. Constr Build Mater 23(11):3337–3345. https://doi.org/10.1016/j.conbuildmat.2009.06.045

Verdie Y, Lafarge F, Alliez P (2015) LOD generation for urban scenes. Trans Graph 34(3):30. https://doi.org/10.1145/2732527

Wilding BV, Dolatshahi KM, Beyer K (2018) Shear-compression tests of URM walls: various setups and their influence on experimental results. Eng Struct 156(2017):472–479. https://doi.org/10.1016/j.engstruct.2017.11.057

Zhang S, Beyer K (2019) Numerical investigation of the role of masonry typology on shear strength. Eng Struct 192:86–102. https://doi.org/10.1016/J.ENGSTRUCT.2019.04.026

Zhang S, Taheri Mousavi SM, Richart N, Molinari JF, Beyer K (2017) Micro-mechanical finite element modeling of diagonal compression test for historical stone masonry structure. Int J Solids Struct 112:122–132. https://doi.org/10.1016/j.ijsolstr.2017.02.014

Authors and Affiliations

Bryan German Pantoja-Rosero[1] · Radhakrishna Achanta[2] · Katrin Beyer[1]

✉ Katrin Beyer
katrin.beyer@epfl.ch

Bryan German Pantoja-Rosero
bryan.pantojarosero@epfl.ch

Radhakrishna Achanta
radhakrishna.achanta@epfl.ch

[1] Earthquake Engineering and Structural Dynamics Laboratory (EESD), EPFL, 1015 Lausanne, Switzerland

[2] Swiss Data Science Center (SDSC), EPFL - ETHz, 1015 Lausanne, Switzerland