

3D MODELING OF A COMPLEX BUILDING: FROM MULTI-VIEW IMAGE FUSION TO GOOGLE EARTH PUBLICATION

L. Inzerillo^{1*}, F. Leto Barone², R. Roberts³

¹DARCH, Department of Architecture, University of Palermo, 90128 Palermo, Italy- laura.inzerillo@unipa.it

²Dipartimento di Ingegneria dell'Energia, dei Sistemi, del Territorio e delle Costruzioni, University of Pisa, 56122 Pisa, Italy – francescoletobarone@gmail.com

³DIING, Department of Engineering, University of Palermo, 90128 Palermo, Italy- ronaldanthony.roberts@unipa.it

Commission II, WG II/8

KEY WORDS: Data fusion, UAV, 3D modeling, Google Earth

ABSTRACT:

This paper presents a pipeline that aims at illustrating the procedure to realize a 3D model of a complex building integrating the UAV and terrestrial images and modifying the 3D model in order to publish to Google Earth in an interactive modality so as to provide better available models for visualization and use. The main steps of the procedure are the optimization of the UAV flight, the integration of the different UAV and ground floor images and the optimization of the model to be published to GE. The case study has been identified in a building, The Eremo di Santa Rosalia Convent in Sicily which has more staggered elevations and located in the hills of the hinterland and of which, the online platform only indicates the position on Google Maps (GM) and Google Earth (GE) with a photo from above and a non-urban road whose GM path is not corresponding with the GE photo. The process highlights the integration of the models and showcases a workflow for the publication of the combined 3D model to the GE platform. (Figure 1)



Figure 1. The study site- Eremo di Santa Rosalia

1. INTRODUCTION

Nowadays the use of online Mapping systems is in high demand for tourism, cultural heritage and architecture purposes and research. Given this, mapping platforms such as Google Earth (GE) are relied on to provide clear overviews of areas, buildings and culturally significant landmarks. The imagery provided by these platforms in recent years has shown significant improvements and has led researchers to incorporate their uses within other image replication pipelines (Chen and Clarke, 2016), (Li and Lu, 2018) and (Inzerillo and Roberts, 2019). However, there are numerous instances where it is not easy for the platform to have access to the sites given the location of some landmarks or sites and also with regards to the complexity of the building or landmark (Remondino et al., 2009a). Therefore there are a vast number of models on the platform which do not provide a proper overview of significant points of interest and historical significance. To this end, this paper looks at one such case and aims to develop a workflow for such a site where access is limited but a 3D replicated model of the structure is required for Cultural heritage. The chosen site is the Eremo di Santa Rosalia Convent in Sicily which is an important religious and historical landmark in Quisquina. The case study has been identified in a building with staggered elevations and located in the hills of the hinterland and of which, the online platform only indicates the position on Google Maps (GM) and Google Earth (GE) with a photo from above and a non-urban road whose GM path is not corresponding

with the GE photo. An overview of the study area, which measures approximately 3750 m² is shown in Figure 1.

1.1 Aim of Case Study

The paper presents a framework on how to integrate data from different levels namely images coming from UAV, ground level and provide automated corrections to the imagery for superior model production to a sufficient level capable of being published to the GE platform to provide users with a complete model for their required uses. The pipeline includes optimization and dissemination of the model, processing of the model along with the data fusion aspects and then testing phases and then the final model production. This pipeline is depicted in Section 3.

2. RELATED WORKS

There is a vast amount of work related to different techniques for integration and data fusion of models for graphics and computer vision in the fields of cultural heritage and (Fassi, 2007). As such, this section presents a brief overview of studies done on complex architecture reconstructions.

There have been studies which involve the use of floor plans and existing 2D drawings of complex structures and buildings to retrieve measurements and then upscale these for 3D model production (Gerstweiler et al., 2018; Yin et al., 2009). For more advanced treatments image-based models have generally been

* Corresponding author

employed. This involves images taken from both ground level and aerial levels and then modelling carried out to determine location of points in a 3D space to create 3D point clouds and finally 3D models for complex structures such as churches and castles (Altman et al., 2017; Brusco et al., 2006; Kersten et al., 2004; Remondino et al., 2009b; Santagati et al., 2013). These methods are considered semi-automated with required interaction during the phases of model segmentation and development. Full automated methods also exist utilizing video imagery (Pollefeys et al., 2008) and also laser-scanned sources (Zhu et al., 2015) and the use of sensors (Banno et al., 2008). These methods can be however difficult to implement.

For the full automation of complex structure reconstruction, there are also issues concerning the reliance on surface shape constraints and their relationships. El-Hakim, 2006, showed that to solve this a solution for buildings could be enforcing reduced interaction on typical building shapes like columns, doors and windows using them as constraints for the fully automated dense stereo matching.

The use of UAVs has also been done in an effort to obtain imagery for isolated areas and buildings (Colomina and Molina, 2014; Li et al., 2016) and instances where they can be used to produce orthophoto drawings and maps (Liu et al., 2018) and to reconstruct specific points and features on complex buildings also utilizing terrestrial images (Püschel et al., 2008).

Combinations of these different data sources have also been extensively studied with the fusion of photogrammetry and lasers (Barsanti et al., 2012) and of different imagery sources such as

UAV and ground level (Farella et al., 2019). Within the integration of different data sources, there are several issues, which arise due to misalignments, color and texture mismatches and general noise. Farella et al., 2019, developed a pipeline within python, which used to eliminate points, which negatively affect the image orientation step and, consequently, the dense point cloud generation.

3. DEVELOPED METHODOLOGY

The developed methodology is divided into four main phases from the initial survey to the final publication on Google Earth. The first phase involves the test design and data collection with the generation of two separate sparse point clouds, one from the captured terrestrial imagery files and the other from the captured UAV imagery files.

The second phase is the data fusion incorporating the python script aimed at bringing the chunks' coordinate system in accordance with the sides of the bounding box and also to make the necessary colour corrections.

The third phase is the processing phase, which includes the 3D model shadow detection and the multi-image texture compensation.

The last phase is the 3D model optimization and the final publication on Google Earth. These phases and their association with each other are depicted in Figure 2 showing the main aspects of the methodology and the novel elements.

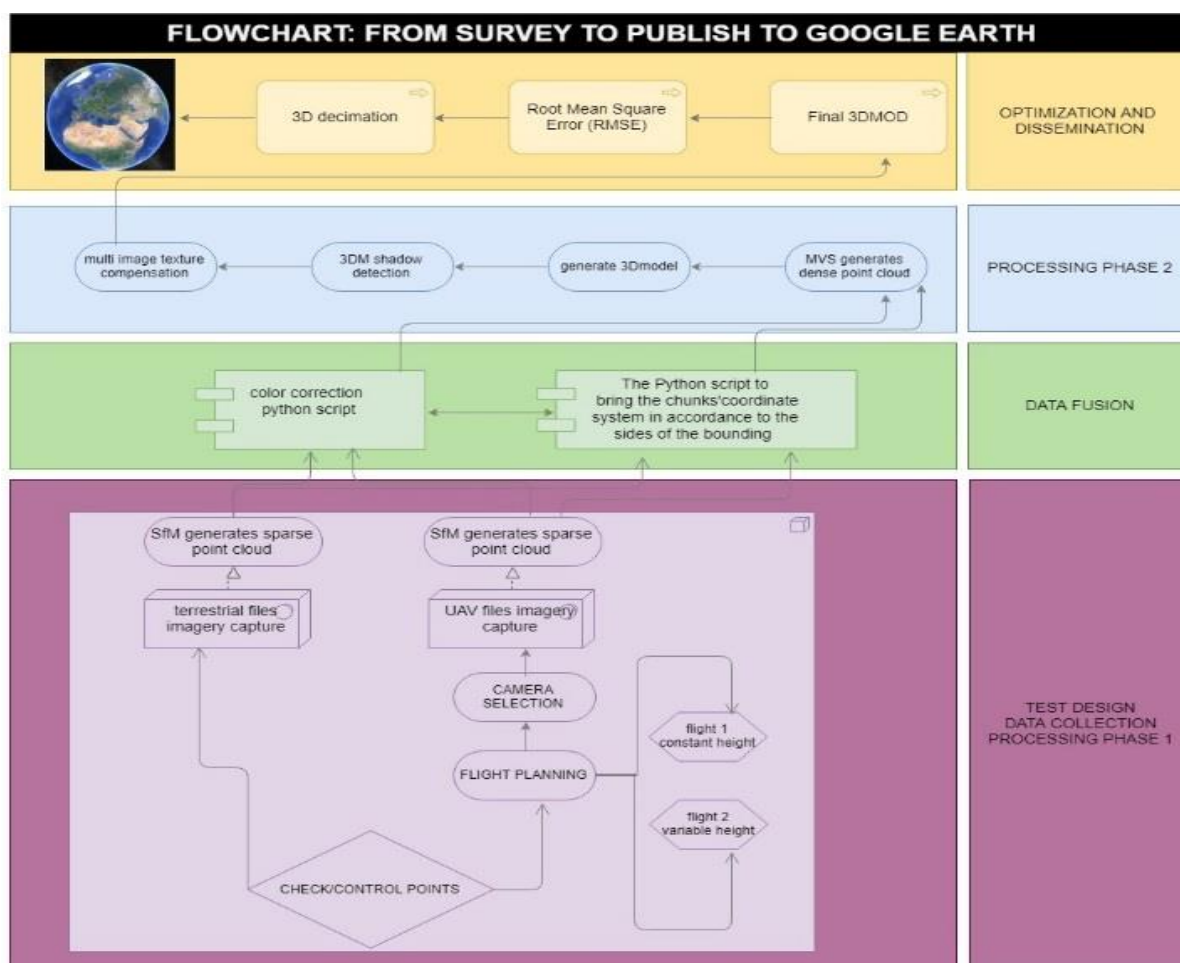


Figure 2. Flowchart from the images to the publishing to GE

3.1 Flight Planning

The most important and delicate step is that of UAV flight planning and the setting up of the two cameras to achieve the same variables; Density, GSD and Accuracy of the 3D model obtained both terrestrial and UAV procedures.

We could not use GE to establish the control points and the checkpoints. Therefore, we chose five control points and three checkpoints distributed in the survey area: along the main road and at building inflexion points. For the flight, we used the Dà-Jiāng Innovations (DJI) S900 UAV six-rotor flight platform, which is a highly portable, lightweight UAV but a strong, stable, and powerful aerial system, which can be used for photogrammetry. This device has a weight of 3.3kg with a recommended maximum take-off weight of 8.2kg, which allows easy use and implementation. When it is used with a 6S 12000mAh battery, the device can be flown for up to 18 minutes (for this study the flight was tested on a breezeless day simulating the same conditions of the case study). We planned two different flights: one at the same altitude (40 m) with a serpentine direction and the other one with a different height due to the different roof heights (average altitude of 15 m). (Figures 3 and 4) These altitude ranges have been chosen considering the distance from the roof in order to have distance values close to the terrestrial images. The camera used was a Sony α7r digital camera providing 36 million pixels and a resolution of 7360×4912 pixels. It had a sensor size of 35.9×24 mm, a Vario-Tessar T* FE 24-70 mm F4 ZA OSS(ZEL2470Z) lens, a shutter time of $1/1600$ s, an aperture of f6.3, and an ISO of 250. The camera weight was 998 g, the focus length was set to 50 mm, and the field of view (FOV) was 46.7° , calculated using Equation (1):

$$FOV = 2 \times \tan^{-1}(d/2f) \quad (\text{Eq. 1})$$

where d is the diagonal length of the sensor size and f is the focus length. Camera calibration was performed as part of the SfM process, which calculated the initial and optimized values of interior orientation parameters.

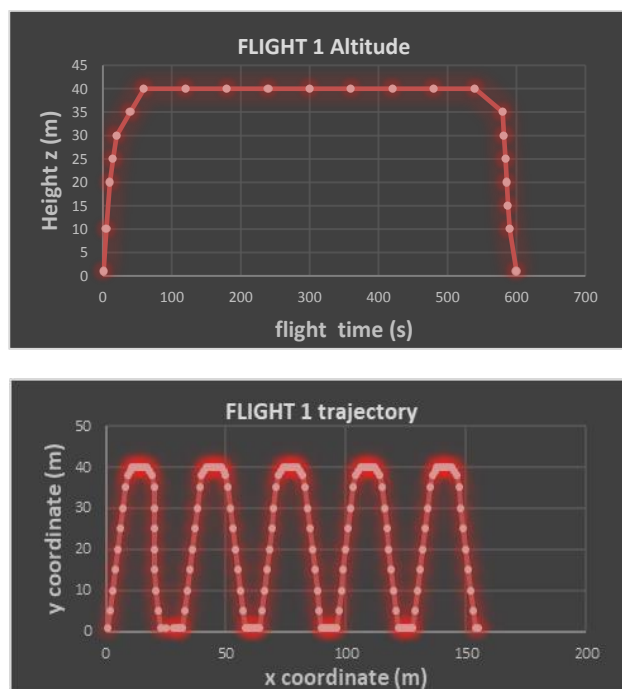


Figure 3. Altitude and planar projection of flight 1

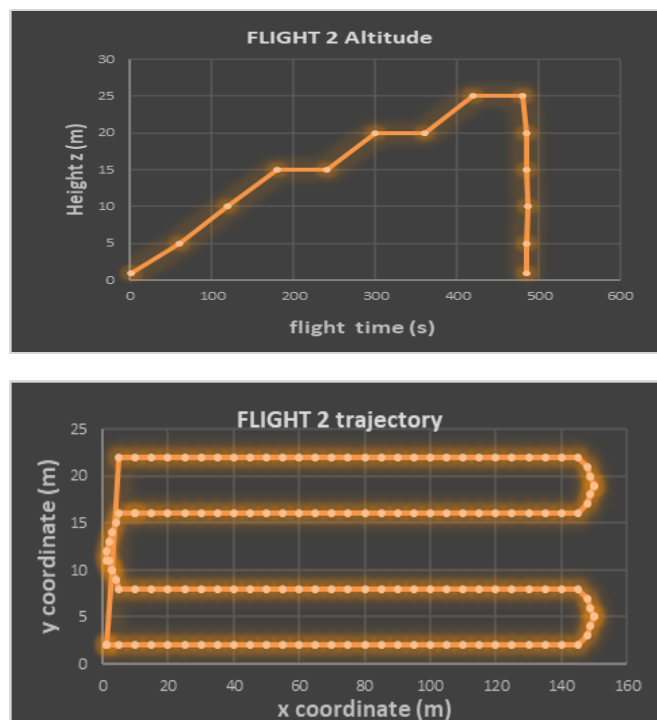


Figure 4. Altitude and planar projection of flight 2

3.2 Python Scripts for corrections

The terrestrial images have been taken considering the lowest distance that the environmental allowed. To avoid the problems due to the integration of different sources of images (visible misalignments, color and texture mismatches, noisy, etc.) there have been procedures developed. The pipeline proposed by (Farella et al., 2019) eliminating points that negatively affect the image orientation step and, consequently, the dense point cloud generation was utilized.

In this work, since we used the same camera with the same sensor for the terrestrial images but with different light exposition and different distances, we experimented two different Python scripts to bring the chunks' coordinate system in accordance to the sides of the bounding box and to make a color correction between the different image chunks.

The Python script to bring the chunks' coordinate system in accordance to the sides of the bounding box is:

```
> #rotates model coordinate system in accordance of
> #scale is kept
> #compatibility: Agisoft PhotoScan Professional 1.3.0
> import PhotoScan
> import math
> doc = PhotoScan.app.document
> chunk = doc.chunk
> R = chunk.region.rot #Bounding box
> rotation matrix
> C = chunk.region.center #Bounding box
> center vector
> if chunk.transform.matrix:
> T = chunk.transform.matrix
> s = math.sqrt(T[0,0] ** 2 + T[0,1] ** 2 + T[0,2] ** 2)
> #scaling
> S = PhotoScan.Matrix().Diag([s, s, s, 1]) #scale
> matrix
> else:
> S = PhotoScan.Matrix().Diag([1, 1, 1, 1])
> T = PhotoScan.Matrix([ [R[0,0], R[0,1], R[0,2], C[0]],
> [R[1,0], R[1,1], R[1,2], C[1]], [R[2,0], R[2,1],
> R[2,2], C[2]], [0, 0, 0, 1]])
> chunk.transform.matrix = S * T.inv()
> #resulting chunk transformation matrix
```

* Corresponding author

The color correction Python script is:

```
> #creates masks for cameras in the active chunk, based
> on user defined color and tolerance
> #compatibility PhotoScan Pro 1.3
> import PhotoScan
> from PySide2 import QtGui, QtCore, QtWidgets
> class MaskByColor(QtWidgets.QDialog):
>     def __init__(self, parent):
>         [...]
>     def main():
>         global doc
>         doc = PhotoScan.app.document
>         global app
>         app = QtWidgets.QApplication.instance()
>         parent = app.activeWindow()
>         dlg = MaskByColor(parent)
>         PhotoScan.app.addMenuItem("Custom/Masking by color",
>                                     main)
```

3.3 Shadow detection

Due to the different heights of the building and its complexity, the presence of shadows was inevitable. To overcome the problem generated on the texture of the shaded areas, we used a manual multi-view image compensation, drawing manually the masked areas. This one and the adjacent images were selected for manual sorting. The filling work was performed one-by-one. The occlusion analysis was performed on the first adjacent image after sorting to avoid the adjacent image also having a shaded area and, shadow detection was performed on the next adjacent image; otherwise, if the first adjacent image was not shaded, a texture compensation was performed; using this procedure it has been possible to generate a 3D model without shadows as depicted in Figure 5.



Figure 5. 3D model shadow detection

Once applied, the python scripts and made the compensation of the 3D model, we calculated Root Mean Square Error (RMSE) to evaluate the accuracy, using the checkpoints' coordinates (x, y, and z) in the model and those measured in the field. We calculated the plan error (X, Y) and the elevation one (Z), using the following equations:

$$RMSE_X = \sqrt{\frac{\sum_{i=1}^n [(X_{Fi} - X_{Mi})^2]}{n}} \quad RMSE_Y = \sqrt{\frac{\sum_{i=1}^n [(Y_{Fi} - Y_{Mi})^2]}{n}} \\ RMSE_Z = \sqrt{\frac{\sum_{i=1}^n [(Z_{Fi} - Z_{Mi})^2]}{n}} \quad (Eq. 2)$$

$$RMSE_{XY} = \sqrt{\frac{\sum_{i=1}^n [(X_{Fi} - X_{Mi})^2 + (Y_{Fi} - Y_{Mi})^2]}{n}} \quad (Eq. 3)$$

where: RMSE is the error, X_{Fi} , Y_{Fi} , and H_{Fi} are the value measured in the field, X_{Mi} , Y_{Mi} , H_{Mi} are the coordinates obtained from the 3D model, and n is the number of checkpoints. This was done to establish the accuracy of the created models as well as to ensure the use of images at different heights and

therefore different Ground Sample Distances was not an issue for the final model.

3.4 3D model optimization

Concerning the last step, the best result was obtained using an algorithm that guarantees the reduction of the polygonal mesh of the 3D model. Most mesh simplification algorithms based on vertex decimation assign a weight to each individual vertex, further referred to as the relevance weight that signifies its importance to the shape of the object. If the value is small, the vertex can be removed without significantly altering the mesh, while if the weight is larger the vertex must be kept. The values used as weights for the vertices are different for each implementation but generally are based on the geometrical properties of the surrounding region, such as the distances, areas or measures of curvature. The typical methodology for this involves the removal of a single vertex, v with a re-triangulation of its crown. By doing this for each step patches of n , triangles (the valence of v) are replaced new patches of $n-2$ triangles. Generally, to guarantee a reasonable quality of the re-triangulated patch, a local edge-swapping optimization is necessary. If a polygon mesh has to be decimated it must be converted into a triangle mesh before.

The decimation process consists usually into the vertex classification (characterizes the local geometry and topology for a given vertex), the decimation criterion (estimation of the error, when a given vertex is removed) and triangulation (after a vertex has been removed, the resulting hole has to be triangulated). We used the Multiple- Choice Algorithm (MCA) with the following framework:

1. Feed a section of the mesh into the main memory
2. Select a random value for d , the atomic decimation operators from all candidates; calculate their respective decimation costs utilizing a particular error/quality metric.
3. Choose the operator, which has the smallest cost/error, and execute this operator
4. Repeat steps 2 and 3 until there the user-defined criteria are met thus terminating the process

The MCA allows us to carry out a high-performance mesh running in a faster time. The triangulated 3D model was then downloaded on SketchUp to allow the GE procedure to publish it.

4. EXPERIMENTAL SETUP AND DISCUSSION

Based on the methodology established in Section 3 the case study was carried out on the Eremita di Santa Rosalia Convent. The imagery dataset comprised of 145 images of the Convent (44 terrestrial and 101 from the UAV, with the UAV images being split as follows: 40 taken at a constant height and 61 at variable heights as defined by the flight plans).

Both the terrestrial images and the UAV images were taken with a Sony $\alpha 7r$ digital camera providing 36 million pixels and a resolution of 7360×4912 pixels. By doing this and using one camera this allowed easier use of images as the camera parameters were the same therefore making it easier for the data fusion.

After this, the images were imported to PhotoScan wherein the typical photogrammetry rules were followed (M. Mikhail et al., 2001) to carry out alignment and point cloud creation. Within this PhotoScan environment, the python scripts for color correction and coordinate system alignment (shown in section 3.2) were then subsequently applied. This step allowed necessary corrections to enable a highly accurate final product. Following this, the dense clouds generated and the shadow correction and the texture compensation steps carried out and this product is

shown in figure 5. The next step in the process was the decimation and the calculation of the RSME errors as highlighted in section 3.3 and 3.4. The resulting 3D model was able to demonstrate a high performance with the fusion of the different sets along with the applied corrections, therefore, making it suitable for publication on GE.

For the ground-level models, we achieved accuracies of 1mm to 2mm, which give an average relative accuracy of one part in 6,000 and represents 0.3 pixels.

The final step in this process was the publication of the model to GE from the SketchUp environment. The decimated model was imported to the SketchUp environment by utilizing blender to convert the 3D model from PhotoScan to a suitable format for SketchUp wherein planar faces were resolved to allow the model to be used within the GE environment whilst not reducing the overall quality of the model to an unsuitable state as shown in figure 6.

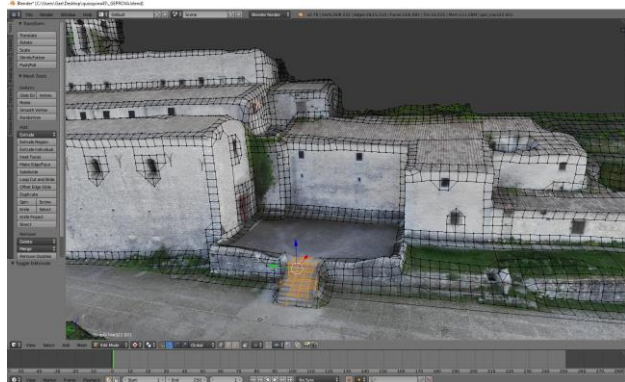


Figure 6. The model in the Blender environment

Subsequently, the model was imported to SketchUp. Then it was aligned with the coordinate system applicable for the GE platform as shown in figure 7.

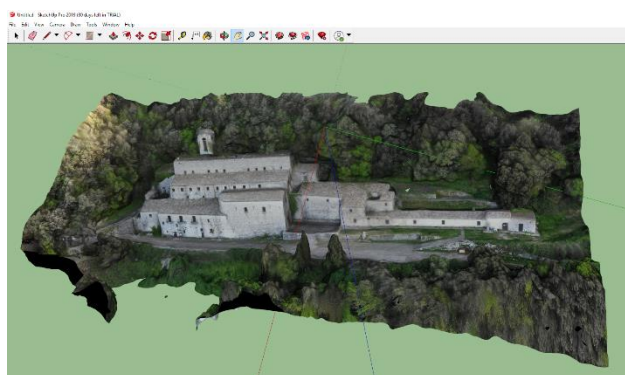


Figure 7. The decimated model on the SketchUp environment

Consequently, the model was published to the GE platform and this is highlighted in figure 8. After this, the model and related files are sent to Google for review and publication. This process is lengthy as there is a review of the data. After this, pop-up window visualizations on the 3D model within the platform could then be possible. It is also possible to ascertain a GE ready model directly from the PhotoScan platform but this requires the mesh to be less than 64,000 vertices. This results in a very low-quality model which does not offer the representative features required given the significance of the landmark. In figure 8, we show the result of the procedure from the 3D modelling of the complex building, multi-view image fusion and optimization to the publication on GE. The GE view represents a vastly improved

3D model and one that can accurately capture features on the convent that were not possible before. Ortho photographic representations of the convent were developed for the GE implementation and these are depicted in the Appendix.

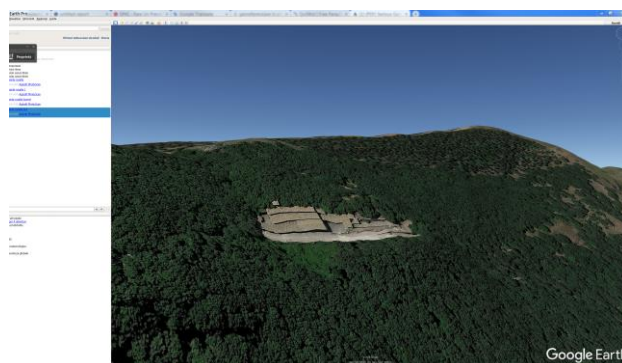


Figure 8. Google Earth before and after the publication of the 3D model

5. CONCLUSION AND FUTURE WORK

In this paper, we demonstrated that, despite the challenge due to the size and complexity of the building, we experimented with a flow chart that optimizes the procedure to realize a 3D model of a complex building, from multi-view image fusion to google earth publication.

The Python application for data fusion, the MCA algorithm for the optimization of the 3D model present a new procedure to publish the 3D model to GE using the novelty and originality of this workflow.

The integration of different methodologies and technologies allowed us to perform an efficient workflow, satisfying each one of them as better as they can. In a case like this one, where the complexity and size of the building are considerable, it is necessary to integrate different methodologies and use all the strategies to achieve the final result. The past iconography, the ground and drone images, the integrated survey and the experimented technicalities and efficient management of the procedure and methodology, become essential to carry out the best result.

Using the Python script, the Blender shadow detection and the optimization of the 3D model to be used in GE environment, we automated some phase of the procedure. Nevertheless, the full automation of the flow chart is the highest challenge of the researchers and this work is still ongoing.

ACKNOWLEDGEMENTS

The research presented in this paper was carried out as part of the H2020-MSCA-ETN-2016. This project has received funding from the European Union's H2020 Programme for research,

technological development and demonstration under grant agreement number 721493.

REFERENCES

- Altman, S., Xiao, W., Grayson, B., 2017. Evaluation of Low-Cost Terrestrial Photogrammetry for 3D Reconstruction of Complex Buildings. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. IV-2/W4, pp. 199–206. <https://doi.org/10.5194/isprs-annals-IV-2-W4-199-2017>.
- Banno, A., Masuda, T., Oishi, T., Ikeuchi, K., 2008. Flying laser range sensor for large-scale site-modeling and its applications in bayon digital archival project. *International Journal of Computer Vision*, 78(2-3), pp. 207–222. <https://doi.org/10.1007/s11263-007-0104-6>.
- Barsanti, S.G., Remondino, F., Visintini, D., 2012. Photogrammetry and Laser Scanning for archaeological site 3D modeling—Some critical issues. In: *Proceedings of the 2nd Workshop on 'The New Technologies for Aquileia'*, V. Roberto, L. Fozzati.
- Brusco, N., Capeleto, S., Fedel, M., Paviotti, A., Poletto, L., Cortelazzo, G.M., Tondello, G., 2006. A system for 3D modeling frescoed historical buildings with multispectral texture information. *Machine Vision and Applications*, 17(6), pp. 373–393. <https://doi.org/10.1007/s00138-006-0026-2>.
- Chen, J., Clarke, K.C., 2016. Rapid 3D Modeling Using Photogrammetry Applied to Google Earth, In: *Proceedings, AutoCarto2016, The 19th International Research Symposium on Computer-Based Cartography*. Albuquerque, New Mexico, USA.
- Colomina, I., Molina, P., 2014. Unmanned aerial systems for photogrammetry and remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol.92, pp. 79–97. <https://doi.org/10.1016/j.isprsjprs.2014.02.013>.
- El-Hakim, S., 2006. A Sequential Approach to Capture Fine Geometric Details from Images, in: *ISPRS Commission V Symposium - Image Engineering and Vision Metrology*, Vol. XXXVI, Part 5, pp. 97–102.
- Farella, E.M., Torresani, A., Remondino, F., 2019. Quality Features for the Integration of Terrestrial and UAV Images. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XLII-2/W9, pp. 339–346. <https://doi.org/10.5194/isprs-archives-XLII-2-W9-339-2019>.
- Fassi, F., 2007. 3D Modeling of Complex Architecture Integrating Different Techniques – a Critical Overview. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Trento, 3D-ARCH 2007, Vol.XXXVI-5/W47.
- Gerstweiler, G., Furlan, L., Timofeev, M., Kaufmann, H., 2018. Extraction of Structural and Semantic Data from 2D Floor Plans for Interactive and Immersive VR Real Estate Exploration. *Technologies*, 6(4), pp. 101. <https://doi.org/10.3390/technologies6040101>.
- Inzerillo, L., Roberts, R., 2019. 3D Image Based Modelling Using Google Earth Imagery for 3D Landscape Modelling. In: *Advances in Intelligent Systems and Computing*, 919, pp. 627–634. https://doi.org/10.1007/978-3-030-12240-9_65.
- Kersten, T., Acevedo Pardo, C., Lindstaedt, M., 2004. 3D Acquisition, Modelling and Visualization of North German Castles by Digital Architectural Photogrammetry. In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXV/B2, pp. 126–131.
- Li, D., Lu, M., 2018. Integrating geometric models, site images and GIS based on Google Earth and Keyhole Markup Language. *Automation in Construction*, 89, pp. 317–331. <https://doi.org/10.1016/j.autcon.2018.02.002>.
- Li, M., Nan, L., Smith, N., Wonka, P., 2016. Reconstructing building mass models from UAV images. *Computers and Graphics*, 54, pp. 84–93. <https://doi.org/10.1016/j.cag.2015.07.004>.
- Liu, Y., Zheng, X., Ai, G., Zhang, Y., Zuo, Y., 2018. Generating a High-Precision True Digital Orthophoto Map Based on UAV Images. In: *ISPRS International Journal of Geo-Information* 7(9), p. 333. <https://doi.org/10.3390/ijgi7090333>.
- M. Mikhail, E., S. Bethel, J., Mcglone, C., 2001. Introduction to Modern Photogrammetry, John Wiley & Sons, New York, USA.
- Pollefeys, M., Nistér, D., Frahm, J.M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.J., Merrell, P., Salmi, C., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G., Towles, H., 2008. Detailed real-time urban 3D reconstruction from video. *International Journal of Computer Vision*, 78, pp. 143–167. <https://doi.org/10.1007/s11263-007-0086-4>.
- Püschel, H., Sauerbier, M., Eisenbeiss, H., 2008. A 3D Model of Castle Landenberg (CH) from combined photogrammetric processing of terrestrial and UAV-based images. In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXVII, pp. 93–98.
- Remondino, F., El-Hakim, S., Girardi, S., Rizzi, A., Benedetti, S., Gonzo, L., 2009a. 3D Virtual Reconstruction and Visualization of Complex Architectures - The 3D-ARCH Project. In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXVIII-5/W1.
- Santagati, C., Inzerillo, L., Di Paola, F., 2013. Image-Based Modeling Techniques for Architectural Heritage 3D Digitalization: Limits and Potentialities. In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XL-5/W2, pp. 555–560. <https://doi.org/10.5194/isprsarchives-XL-5-W2-555-2013>.
- Yin, X., Wonka, P., Razdan, A., 2009. Generating 3D Building Models from Architectural Drawings: A Survey. *IEEE computer graphics and applications*, 29(1), pp. 20–30. <https://doi.org/10.1109/mcg.2009.9>.
- Zhu, L., Lehtomäki, M., Hyypä, J., Puttonen, E., Krooks, A., Hyypä, H., 2015. Automated 3D scene reconstruction from open geospatial data sources: Airborne laser scanning and a 2D topographic database. *Remote Sensing*, 7(6), pp. 6710–6740. <https://doi.org/10.3390/rs70606710>.

APPENDIX

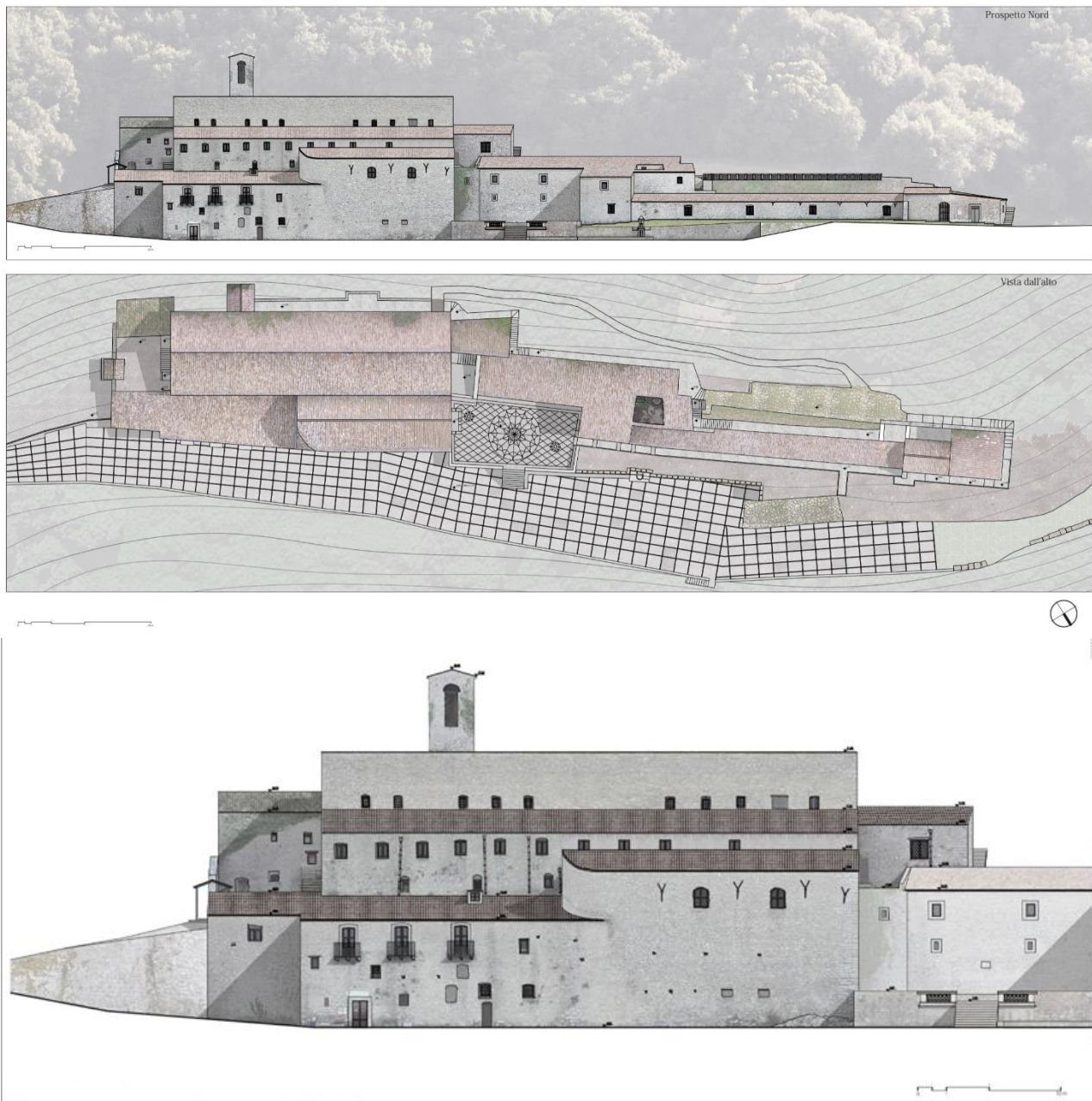


Figure 9. Orthoprojections of the façade and of the above view of the building.

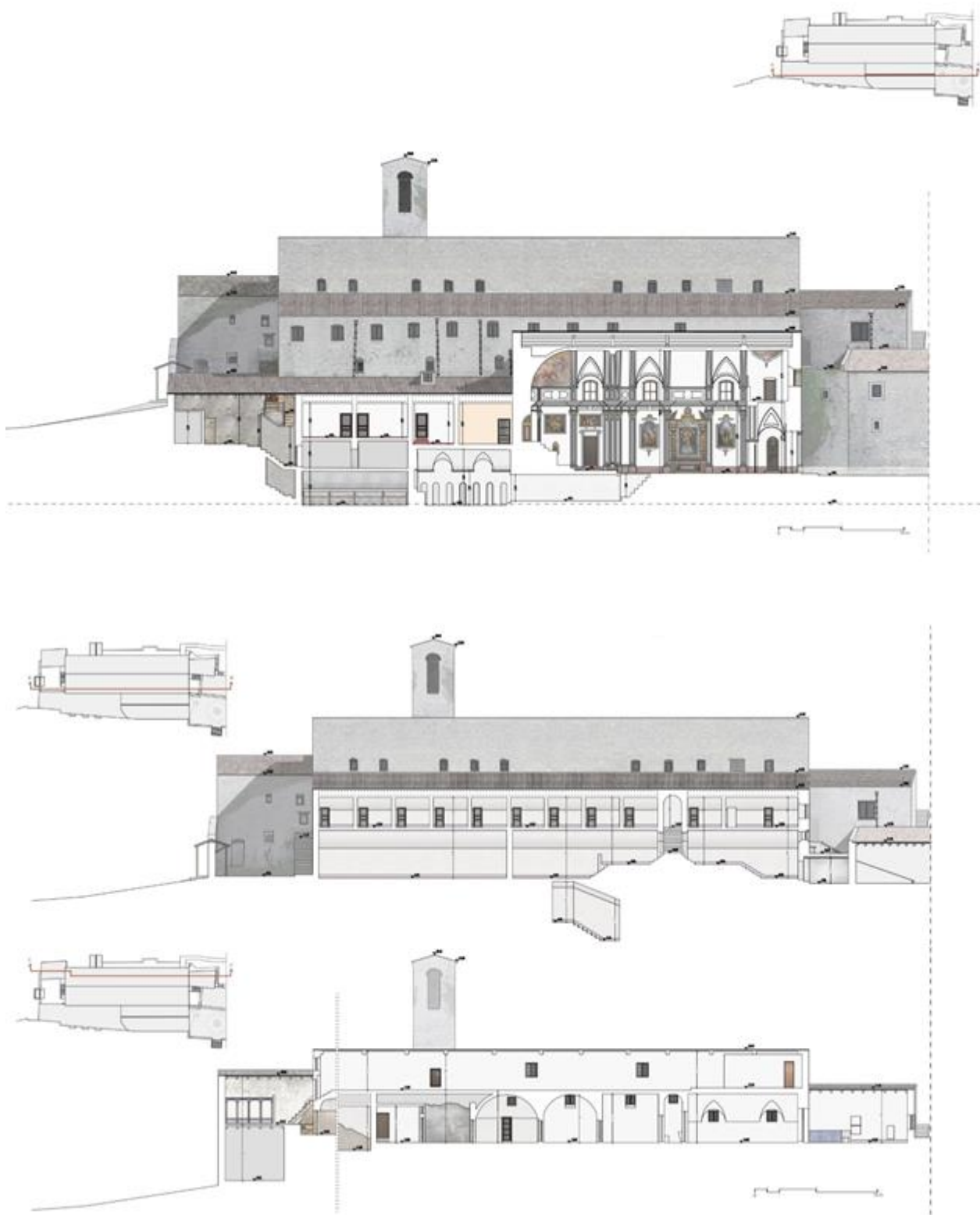


Figure 10. Transversal sections of the building at different levels and plant positions.