

THE COMBINATION OF TERRESTRIAL LIDAR AND UAV PHOTOGRAMMETRY FOR INTERACTIVE ARCHITECTURAL HERITAGE VISUALIZATION USING UNITY 3D GAME ENGINE

R. Andaru^{1,2}, B. K. Cahyono¹, G. Riyadi³, Istarno¹, Djurdjani¹, G. R. Ramadhan⁴, S. Tuntas⁴

¹ Department of Geodetic Engineering, Universitas Gadjah Mada, Indonesia – (ruliandaru, bambangkun, istarno, djurdjani)@ugm.ac.id

² Department of Geomatics, National Cheng Kung University, Taiwan

³ Department of Earth Technology, Vocational School of Universitas Gadjah Mada, Indonesia - gondangr@ugm.ac.id

⁴ Alumni Department of Geodetic Engineering, Universitas Gadjah Mada, Indonesia – gilangrr11, sahitsadono@gmail.com

Commission II

KEYWORDS: Terrestrial Lidar, UAV Photogrammetry, architectural heritage, unity 3D, game engine

ABSTRACT:

The digital 3D documentation of architectural heritage using advanced 3D measurement technologies such as UAV photogrammetry and terrestrial LiDAR (TLS) becomes a potential and efficient method since it can produce 3D pointclouds in detail and high density of pointclouds levels. However, TLS is unable to scan the roof part of tall building, whereas UAV photogrammetry achieves high density of pointclouds at that area. In order to make a complete 3D pointclouds of heritage building, we merged and integrated the TLS and UAV pointclouds data by using Iterative Closest Point (ICP) algorithms into one reference system. In this study, we collected two architectural heritage building in Yogyakarta, Indonesia, i.e., "Vredenburg Fort Museum (VFM)" and "Kotagede Great Mosque (KGM)", the oldest mosque in Yogyakarta. For the data acquisition, we used Faro Focus X330 and GLS 2000 Laser Scanner. We produced three-dimensional point clouds from UAV imagery by using Structure from Motion and Multi View Stereo (SfM-MVS) technique through Photoscan software. In order to merging and integrating both of pointclouds data, Maptex I-Site Studio 6.1 with Educational License was used. Those data were successfully registered, and according to the registration report, we had observed 20.60 mm of RMS error. The 3D models and their textures in outdoor and indoor side were processed using Autodesk software. Modelling was carried out on the structure of building's façade base on simple geometric primitive as planes, straight lines, circles, spheres and cylinder. For interactive visualization, a modern and widely accessible game engine technology (Unity3D) was used. The result was an interactive displaying 3D model of an architectural heritage building in LOD3 level with spatial function for measuring the size and dimension, as well as the area of object. Finally, we created the online version of interactive 3D viewer utilizing WebGL API and Mapbox Unity SDK.

1. INTRODUCTION

3D modeling can be generated with various methods, such as UAV photogrammetric and laser scanning methods. Data acquisition using terrestrial LiDAR is able to achieve high density of pointclouds. However, it has low level of pointclouds density in the roof part of tall building due to the fact that TLS was only placed on the ground. Thus, it will cause incomplete building's pointclouds at the roof part as shown in Figure 1.



Figure 1. Incomplete building's pointclouds on the roof part (denoted by red circle) as the result of Terrestrial Lidar acquisition.

Fortunately, UAV photogrammetry is able to produce high density of pointclouds at the roof part since it carries the camera sensor and captures the object from above. By acquiring images through UAV, the roof top part can be mapped properly (Figure 2). In order to make a complete 3D pointclouds of heritage building, we then merged and integrated the TLS and UAV pointclouds data. The combination of both data are able to

produce the whole building from the bottom to the roof top in detail, completely, and realistically.

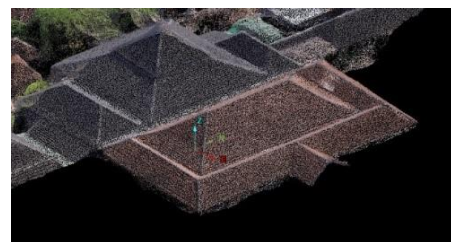


Figure 2. High density of UAV photogrammetric pointclouds at the roof part of building.

For the data collection, we use Faro Focus X330 and GLS 2000 Laser Scanner. UAV DJi Phantom 3 Pro with 12MP camera resolution was used to acquire the image and produce three-dimensional point clouds by using Structure from Motion (SfM) and Multi View Stereo (SfM-MVS) technique using Photoscan software.

This research mainly has two aims. The first one is to integrate the terrestrial LiDAR and UAV photogrammetry pointclouds data using Iterative Closest Point (ICP) algorithms. The next one is to build interactive displaying of a web 3D model with spatial function utilizing WebGL API and Mapbox. For its purpose, a modern and widely accessible game engine technology

(Unity3D) were used. Unity 3D Game Engine is the highest quality of game engine available in the market, and it is easy to use. 3D game engine can be extended for better visualization, format interoperability and other spatial related to functions or applications in the near future (Buyuksalih et al., 2017).

2. RELATED WORKS

This paper presents an automatic method for integrating Laser scanning and UAV pointclouds data. The roofs are extracted from UAV image and the building's wall are extracted from terrestrial LiDAR equipment. Although 3D models are useful to preserve the information of heritage building, they will not be fully accomplished until it delivers interactively and communicates their potential functions to the users.

(Laksono and Aditya, 2019) explored the Unity 3D game engine visualizing large-scale topographic data from mixed sources of terrestrial laser scanner models and topographic map data. The interactive visualization displaying as a game platform allowed the users to explore the 3D models in LOD3 level. The use of modern and widely accessible game engine technology (Unity3D) was also explored by (Albourae et al., 2017) who integrated pipeline between HBIM, GIS and augmented and virtual reality (AVR) tools.

Data integration are solution to create complete 3D database of the complex and tall architectural structure as proposed by (Achille et al., 2015) who merged with three methods; TLS data, close range photogrammetry and aerial survey. The solution for the 3D modelling of tall structure is integration of laser scanner data of the internal areas with dense image matching DSM of the external facade and roof part-based photogrammetry. Another way to align datasets by matching the surface geometry is proposed by (Akca, 2010). The algorithms estimate the Euclidian distances between surface patches by least squares and minimize the distance iteratively.

3. METHODOLOGY

3.1 Data Acquisition and Study Area

In this study, we collected two architectural heritage building in Yogyakarta, i.e. "Vredenburg Fort Museum (VFM)", a former colonial fortress built by the Dutch in 1760 and "Kotagede Great Mosque (KGM)", the largest mosque attributed to the kingdom of Mataram, Yogyakarta City, Indonesia.



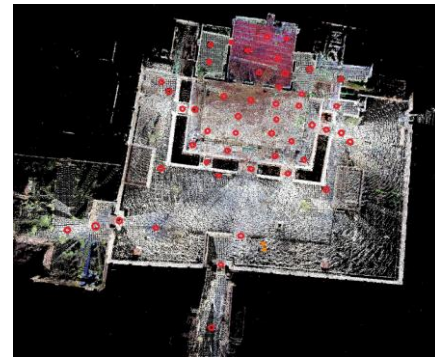
Figure 3. Vredenburg Fort Museum Building



Figure 4. Kotagede Great Mosque Building

We collected 49 scan stations at KGM object, and 102 scan stations for VFM object with an average scan distance of 25m (Figure 5). The terrestrial LiDAR instrument distributed the laser beam at a vertical range of 270° and horizontal range of 360° which was able to produce 1 million points per second. The result of a scan acquisition was a huge number of points in 3D space which represented the object's shape. Each point had an x, y, z coordinate and usually a laser reflectance value. Moreover, the points had a colour information in the form of RGB values. It was provided by a digital camera installed on the scanner system. For the UAV data collection, we had more than 600 images with 60m flying altitude and achieved the Ground Sampling Distance 2.18 cm.

(a)



(b)



Figure 5. Terrestrial Lidar's scan station at Kotagede Great Mosque Building (a) and Vredenburg Fort Museum (b). Red circle denotes TLS scan station

Those data were then processed with the suitable algorithms and compatible software. The workflow and corresponding software used in the process were illustrated in Figure 6.

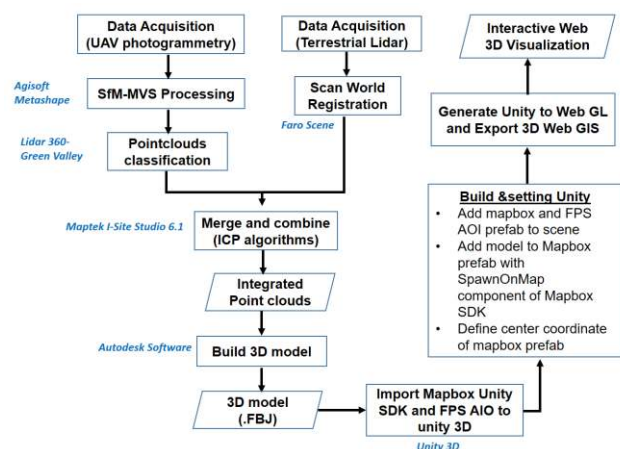


Figure 6. Workflows for data integration and 3D visualization

3.2 Photogrammetric Processing and Pointclouds Classification

We used Agisoft Metashape software to generate 3D point clouds for both of heritage building. The workflow consists several processing steps. It detect and match the feature points in different camera perspectives by using Scale Invariant Feature Transform (SIFT) algorithms, reconstructing 3D scene structure as well as solving the intrinsic and extrinsic orientation parameters of the camera and generating a sparse point cloud through SfM (Smith et al., 2015). We also conducted on-the-job self-calibration for calculating interior orientation parameter. Finally, we used MVS dense image matching algorithm to produce 3D dense point cloud. UAV dense pointclouds are reliable, applicable, acceptable accuracy, and almost has equal resolution to TLS techniques.

However, the pointclouds computed from SfM and MVS algorithms were still noisy and sparse. To overcome these problems, we applied Random Forest Machine Learning (RFML) to classify and remove non-building object provided by Lidar 360-Green Valley Software. There are 3 main steps while using RFML to remove noisy pointclouds, i.e. creating and defining the training data, generating trained model and classifying the data (Green-Valley, 2018). The machine learning classification uses the trained model for determining individual point classifications based on a statistical model of user's defined feature types. Random Forests were successfully applied to classify pointclouds derived from photogrammetric reconstruction, especially for building object.

3.3 TLS Pointclouds Registration

Since each scan world is defined from the coordinate scanner system (local coordinate), then it needs to register all of scan world pointclouds data. Pointclouds registration needs a fixed position and orientation from the coordinate system scanner which appropriate for global coordinate system. Generally, registration techniques are divided into two categories: direct and indirect procedures. Here, we used indirect registration base on cloud registration. It uses the result of overlap (30% - 40%) of point clouds which is also called the Iterative Closed Point (ICP) processing technique. This algorithms is used to find the rigid transformation between the target point and the reference point, so that the two matching data satisfy the optimal match under some kind of metric criterion (He et al., 2017). For this purposes, Faro Scene software was used, and what we got was that the RMS error of registration was 2mm.

3.4 Combining and Integrating Pointclouds Data

From the previous steps, we already had a registered pointclouds data from TLS data and UAV imagery. However, both of these data were not registered yet. Thus, we integrated them into one reference system in order to produce the entire building as well as the roof part. For this purpose, we applied the same algorithms, i.e. ICP, to integrate both of pointclouds data. It requires an estimation of the relative position between two data source at the overlapping area.

We set pointclouds from TLS as reference data. Based on overlapping area, algorithm searches for each point in TLS data which is the closest point in the UAV data and uses the corresponding point pairs to compute a new relative position. This process is repeated iteratively until the relative position of the scans converges (Alshwabkeh and Haala, 2004). The ICP algorithm works well if the pointclouds sources contain relatively large areas of continuous surfaces and have sufficient overlap.

3.5 Build 3D Model

Different types of 3D models can be constructed from pointclouds data, such as primitives model, mesh models or hybrid models. Based on pointclouds data, the 3D model can be reconstructed semi automatically and integrated into BIM software (Macher et al., 2017). In this research, we used Autodesk Maya, a program to build model geometry into various primitive models. The easiest way to start creating complicated models is to begin with 3D primitive shapes. Maya has multiple pre-made shapes that we can use as a starting base. These are known as Primitives model. These Primitives model includes cone, sphere, pyramid, torus, cylinder, wedge and box (Autodesk, 2017) as shown in Figure 7.

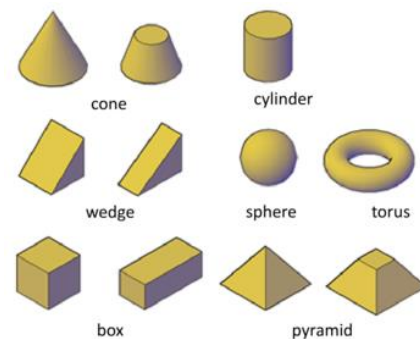


Figure 7. 3D primitive shapes

We built these primitive shape manually base on integrating pointclouds data (Figure 8). In Maya we could integrate images from other sources for texturing the 3D models. We generated a color texture base on TLS color pointclouds combined with terrestrial photo captured by digital camera. After texturing had been completed, then it exported the 3D model into FBX format for further data processing in Unity3D.

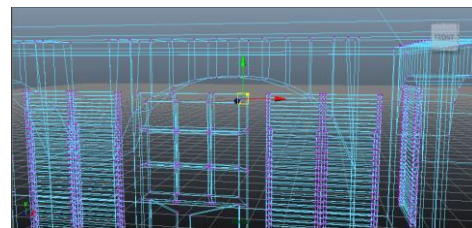


Figure 8. Build 3D primitive model by using Autodesk Maya

3.6 Game Engine Visualization

Game engines such as Unity3D (www.unity3d.com) has been used in various cases for 3D visualization of real-world data (Buyuksalih et al., 2017). The main advantage of Unity3D is that it can run on multiplatform OS including Windows, Mac, Linux and also mobile. It is capable of importing other 3D models in fbx, sbs and obj formats (Mat et al., 2014). The added value in visualizing 3D heritage building by using Unity game engine is the ability to design interactions and user interface. Moreover, we can design spatial function for measuring the size and dimension of object. For this purposes, we build custom script using C# language for spatial function and modified FPS AOI packages for navigation and movement function. Figure 9 shows an example of a script used for calculating distance.

```

CalcDistance.cs
Miscellaneous files
CalcDistance

using System;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class CalcDistance : MonoBehaviour {
    private Vector3 StartPosition;
    private Vector3 EndPosition;
    private GameObject lineGO;
    private LineRenderer lineRenderer;
    private int lineCount = 0;
    private string CalculationMode = "distance";
    private bool Drawing = false;

    private List<GameObject> SphereForArea = new List<GameObject> ();
    private List<GameObject> LineForArea = new List<GameObject> ();

    private List<Vector3> PointsOfArea = new List<Vector3> ();
    private Vector3 scale0 = new Vector3 (0.0025f, 0.0025f, 0.0025f);

    private GameObject MovingCylinder;

    public Text LabelDistance;
    public Text txt;
    public GameObject sphere1;
    public GameObject sphere2;
    public String correctionDistance;
    private float correctFloat;

    int WhichPoint = 1;
    // Start is called before the first frame update
    void Start () {
        print ("Calculate");
        sphere1.GetComponent<Renderer> ().material.color = Color.red;
        sphere2.GetComponent<Renderer> ().material.color = Color.red;
        sphere1.transform.localScale = scale0;
        sphere2.transform.localScale = scale0;
        sphere1.SetActive (false);
        sphere2.SetActive (false);
    }
}
    
```

Figure 9. Example of python script for spatial function (calculating distance)

After we had 3D model (.FBJ) format, we imported Mapbox Unity SDK and FPS AOI to unity 3D. The model was added to mapbox prefab using modified spawn OnMap component of mapbox SDK. Finally, we defined center coordinate of mapbox prefab and generated Unity to Web GL also export to 3D Web GIS.

4. RESULTS AND DISCUSSION

4.1 TLS and UAV Pointclouds data

In this study, we observed 35,000 square meter area of Vredenburg Fort Museum and 4,000 square meter area of Kotagede Great Mosque with 46 million and 12 million pointclouds elicited by TLS respectively. Figure 10 shows the registered pointclouds from UAV and TLS data.

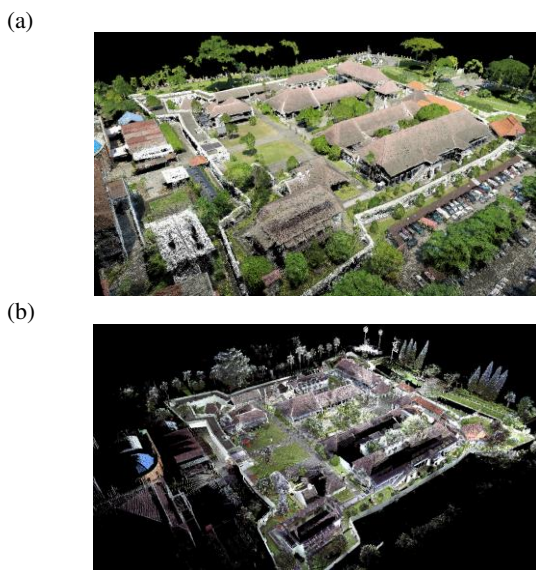


Figure 10. 3D pointclouds at VFM from UAV photogrammetry (a) and TLS measurement (b).

From sFM-MVS algorithm, we produced high density of UAV pointclouds and needed to classify in order to generate building class object. Here, we applied RFML algorithms and the results are shown in Figure 11.

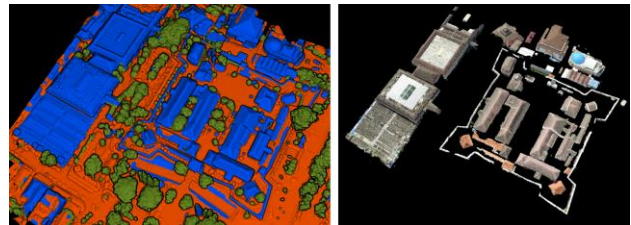


Figure 11. Pointclouds classification results. The building denoted by blue color, the vegetation as green color and the ground as brown color.

4.2 Integration of Two Data Sets

Figure 10b shows low density pointclouds at the roof part which was observed by TLS equipment. However, we got high density of pointclouds at the roof part of building acquired by UAV (Figure 10a). Thus, integrating those two data sets are necessary. Here, we used Maptek I-Site Studio 6.1 software to perform the integration. TLS data provide bottom part of the building and UAV data provide roof surface of the building. It was successfully registered and the whole part of the building could be merged properly as shown in Figure 12. According to the registration report, we had observed 20.60 mm of RMS error. The results of accuracy allowed data from UAV photogrammetry to be fully integrated with laser scanner point clouds. This research, therefore, produced a 3D realistic heritage building which was done by combining TLS and UAV pointclouds data.



Figure 12. Integration of TLS and UAV pointclouds data by using Iterative Closest Point (ICP) algorithms

4.3 3D Primitive model

The 3D model can be drawn accurately corresponding to its size and shape. When the shape of a 3D object is known, it can be described by geometric primitives. The 3D model is built as a solid model using simple geometric primitives representing roof, wall, and other details part such as balcony, stairs, doors and furniture. Figure 13 illustrates an example of LoD3 primitive model based on integrated pointclouds data with manual digitization. This method can be extremely time-consuming and requires advanced skill, especially when the object is detail, occluded by another object, and absent of semantic information in pointclouds. Therefore, more scan stations are required in order to solve those issues.



Figure 13. Original image (left). 3D primitive model result (right).

For VFM heritage building, we built on the exterior area only, whereas at KGM heritage building, we built both on both exterior and indoor areas. Figure 14 shows the 3D primitive model of both heritage building.

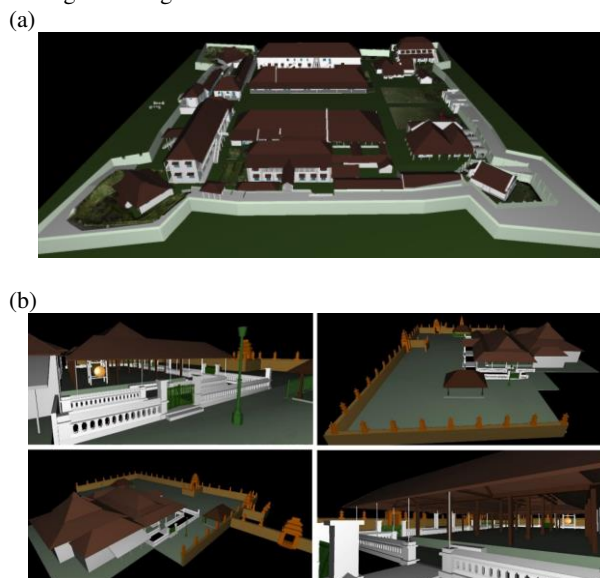


Figure 14. 3D primitive model at Vredeborg Fort Museum (a) and Kotagede Great Mosque (b)

4.4 Interactive Web 3D

The online version of interactive 3D can be accessed from website jogjaheritage3d.com. The main interface of 3D visualization shown in Figure 15. Through the Unity GE platform, users can explore the building environment from a first-person view without needing to walk out of the building and have a great spatial information of the objects in an easy and attractive way. Users can use keyboard buttons W-S-D-A to move forward, backward right, and left respectively. In order to get larger viewpoint, users can increase the height view using R button.



Figure 15. Main interface of 3D visualization

We also add a spatial function to calculate the distance as well as the area in simple way. For calculating distance between 2 points, users can use I button and define 2 points which want to measure. Figure 16 shows the web 3D version from unity as built into the Web GL platform.

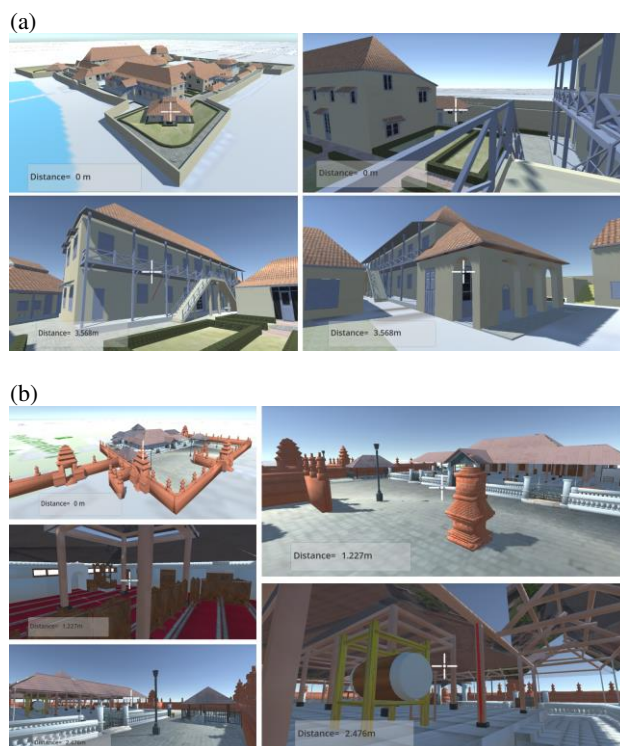


Figure 16. Interactive 3D model at Vredeborg Fort Museum (a) and Kotagede Great Mosque (b).

5. CONCLUSIONS

This paper presents an automatic method for integrating Laser scanning and UAV pointclouds data. The roofs are extracted from UAV image and the building wall are extracted from Laser scanning equipment. This experience proves that UAV photogrammetry and Terrestrial Lidar is an effective way to achieve quick and precise results.

Although 3D models are useful to preserve the information of heritage building, they will not be fully accomplished until it delivers interactively and communicates their function to the users. For this purpose, a modern and widely accessible game engine technology (Unity3D) is used. The result is interactive displaying 3D model of an architectural heritage building in LOD3 level with spatial function for measuring the size and dimension of the objects. Furthermore, we create online version of interactive 3D viewer utilizing WebGL API and Mapbox Unity SDK. These platforms allow visitors to travel the heritage building environment without the need to walk out of the building and have a great spatial information of the objects in an easy and attractive way.

ACKNOWLEDGEMENTS

We would like to thank Ministry of Research, Technology and Higher Education of the Republic of Indonesia for the research development grant (No. 2649/2019). Also for Mr. Aditya Arie and Mr. Lilik Riyadi (PT. Asaba Surveying Division) for supporting Maptex I-Site Studio 6.1 software and the fieldwork. In addition, we acknowledge for M. Iqbal Taftazani, Wahyu Martha M., Rio Andi P., and Endy Prasetyo for contributing fieldwork in 2019.

REFERENCES

Achille, C., Adami, A., Chiarini, S., Cremonesi, S., Fassi, F., Fregonese, L., & Taffurelli, L., 2015. UAV-Based Photogrammetry and Integrated Technologies for Architectural Applications—Methodological Strategies for the After-Quake Survey of Vertical Structures in Mantua (Italy). *Sensors* 2015, 15, 15520-15539; doi:10.3390/s150715520

Akca, D., 2010. Co-registration of Surfaces by 3D Least Squares Matching. *Photogrammetric Engineering & Remote Sensing* Vol. 76, No. 3, March 2010, pp. 307–318

Albourae, A.T., Armenakis, C., Kyan, M., 2017. Architectural heritage visualization using interactive technologies, in: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 7-13, <https://doi.org/10.5194/isprs-archives-XLII-2-W5-7-2017>

Alshawabkeh, Y., 2006. Integration of Laser Scanning and Photogrammetry for Heritage Documentation. *Ph.D Thesis Institut für Photogrammetrie der Universität Stuttgart*, <http://dx.doi.org/10.18419/opus-3730>

Autodesk. (2014). *About Creating Primitive 3D Solids*, Autodesk Help. Retrieved 2019, from <http://help.autodesk.com/view/ACD/2015/ENU>

Buyuksalih, I., Bayburt, S., Buyuksalih, G., Baskaraca, A.P., Karim, H., Rahman, A.A., 2017. 3D Modelling and Visualization Based on The Unity Game Engine – Advantages and Challenges, in: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. vol. IV-4/W4, 161-166, <https://doi.org/10.5194/isprs-annals-IV-4-W4-161-2017>

Carrivick, J.L., Smith, M.W., Quincey, D.J., 2016. *Structure from Motion in the Geosciences*. John Wiley & Sons, Ltd, London. <https://doi.org/10.1177/0309133315615805>

Green-Valley. (2018). *LiDAR 360 User Guide*.

He, Y., Liang, B., Yang, J., Li, S., He, J., 2017. An Iterative Closest Points Algorithm for Registration of 3D Laser Scanner Point Clouds with Geometric Features. *MDPI Sensors*, 17, 1862. <https://doi.org/10.3390/s17081862>

Laksono, D., Aditya, T., 2019. Utilizing A Game Engine for Interactive 3D Topographic Data Visualization. *ISPRS International Journal of Geo-Information* 8(8):361. <https://doi.org/10.3390/ijgi8080361>

Macher, H., Landes, T., Grussenmeyer, P., 2017. From Point Clouds to Building Information Models: 3D Semi-Automatic Reconstruction of Indoors of Existing Buildings. *MDPI journal Applied Sciences, Appl. Sci.* 2017, 7, 1030. <https://doi.org/10.3390/app7101030>

Mat, R.C., Shariff, A.R.M., Zulkifli, A.N., Rahim, M.S.M., Mahayudin, M.H., 2014. Using game engine for 3D terrain visualisation of GIS data: A review. 7th IGRSM International Remote Sensing & GIS Conference and Exhibition. IOP Conference Series: Earth and Environmental Science. <https://doi.org/10.1088/1755-1315/20/1/012037>