ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume X-M-2-2025
30th CIPA Symposium "Heritage Conservation from Bits:
From Digital Documentation to Data-driven Heritage Conservation", 25–29 August 2025, Seoul, Republic of Korea

A Non-Uniform Processing Method (NUPM) for Large Photogrammetry Datasets: Case Study of Shams-ol-Emareh, Golestan Palace (UNESCO World Heritage Site), Iran

Amin Ranjbari[1], Atiyeh Ghorbani[2], and Andrea Jalandoni[3]

[1] Independent researcher, Tabriz, Iran – amin.ranjbari.ac@gmail.com
[2] Independent researcher, Tabriz, Iran – ati.ghorbani.ac@gmail.com
[3] Griffith Centre for Social and Cultural Research, Australian Research Centre for Human Evolution, Gold Coast campus, Griffith University, Australia – a.jalandoni@griffith.edu.au

Keywords: Photogrammetry, Non-uniform 3D Modeling, Big Data, Iran, Cultural Heritage.

Abstract

Cultural heritage documentation benefits from high-quality 3D models that are geometrically and aesthetically accurate. Photogrammetry is used worldwide, yet increases in image resolution, image overlaps, and a combination of drone and terrestrial imagery have resulted in large datasets that raise new challenges. Conventional processing of large datasets requires expensive computational systems with high-capacity RAM. It also leads to large output files that are difficult to store; cloud share; or use in virtual reality, augmented reality, and gaming engines. In this study, we propose a non-uniform processing method (NUPM) to handle the 3D reconstruction of the Shams-ol-Emareh building of the Golestan Palace UNESCO World Heritage Site in Iran. We processed large datasets of photogrammetry on a consumer-grade computer to produce low-size point clouds and meshes with efficient texture size and resolution without sacrificing quality. The workflow first fragmented the object based on importance and roughness, processed each fragment separately, and then joined the fragments together. Non-uniform processing also meant that points, triangles, and pixels with a low level of importance were deleted from all parts of the object. The result was a point cloud, mesh, and texture where the space between points as well as the size of triangles and pixels were variable and non-uniform. In some cases, the number of points in the point cloud and triangles in the mesh were respectively reduced by more than 90% and 97%, leading to usable output sizes without any loss in data quality.

1. Introduction

Close-range photogrammetry has emerged as the most accessible method for recording cultural heritage. However, 3D modelling of cultural heritage sites result in large datasets that pose significant challenges (Bariami et al. 2012). Historical monuments often contain detailed surfaces that must be documented precisely to ensure their preservation for future generations. Consequently, images are captured at a close distance with long lenses and preferably with full-frame cameras, resulting in a large number of high-resolution images.

While technological advances have made it easier to collect more data at higher resolution, they have also increased the need for computational power (Martinez-Rubi et al. 2017). Whether in terms of quantity or high resolution, large image datasets are computationally demanding (Sanchita and Ashwini 2023, Gniady et al. 2017). Both software and hardware developments have focused on the automatic processing of large photogrammetry datasets and storing, visualizing, and rendering large output files (Adrov et al. 2012, Remondino et al. 2017). For example, studies have developed new processing algorithms to reduce the RAM and parallel processing needed to maximize the capacity of computation systems for open-source or commercial photogrammetry software. Another solution is cloud computing services, but their internet-dependency, cost, and loss of control over the photogrammetric process makes their reliability, security, and privacy ambiguous (Peña-Villasenín et al. 2020).

Generating a 3D model is the first problem of working with large datasets. The second problem is limiting the size of the output data without causing a reduction in quality. In general, big data comprises structured and unstructured datasets with massive data volumes that cannot easily be captured, stored, manipulated, analyzed, managed, and presented using traditional hardware, software, and database technologies (Moussa et al. 2013, Li et al. 2016). Given the high acquisition cost, data should be optimized for several outputs (Haubt and Jalandoni 2019). However, the large file sizes of traditional photogrammetry have impacted consumer-grade computers' ability to share data; generate accurate, real-time, realistic visualizations; and create animation in gaming engines.

One common solution has been to decrease file size by reducing the number of points or polygons and decreasing the resolution of the mesh's texture using the Filter Point Cloud, Reduce Mesh, and Resize Texture options. However, this uncontrolled process causes a loss of information and a reduction in the quality of the results (Morales et al. 2010, Anders et al. 2019). In contrast, we propose a method that produces a significant reduction in output data size without reducing the quality of the mesh or texture.

We call our technique for processing large datasets and generating lighter output files that retain quality the "non-uniform processing method" (NUPM). The procedure follows the traditional photogrammetry processing steps, such as aligning photos and generating depth maps, point clouds, meshes, textures, and tiles. However, in each step of the NUPM, the input data from the previous step is divided into smaller fragments that are then processed, edited, decimated, filtered, resized, and ultimately integrated.

We performed a case study on the effectiveness of the NUPM using the Shams-ol-Emareh building in the Golestan Palace (Figure 1). Golestan Palace is on the UNESCO World Heritage List. The Shams-ol-Emareh Tower, built in 1867 during the Qajar period when Nasreddin Shah ruled Iran, is one of the most important buildings on the site because it symbolized Iran's capital city. It is known for its height and decorative elements such as the Gothic-style middle clock tower that contrasts with

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-M-2-2025-293-2025 | © Author(s) 2025. CC BY 4.0 License.

the traditional Iranian-style windows (Zolghadrasli and Hadighi 2021). The building contains two towers with five floors, 56 rooms, corridors, staircases, and other inside spaces; additionally, there are two pergolas, one main porch, four balconies, two terraces, one clock tower and one back alley.

Figure 1. Image of the final 3D model (left) and photograph (right) of the Shams-ol-Emareh building of Golestan Palace.

2. Methods

2.1 General Data Capture

As part of a project for the Organization of Cultural Heritage, Handicrafts, and Tourism in Iran, we collected photogrammetry data to be used in the 3D documentation of the Shams-ol-Emareh building. We used ground points to transfer the coordinate system of the total station into UTM (Universal Transverse Mercator). Then, we measured control points and checkpoints on the outside of the building, the inside rooms, and other parts of the structure using a total station. We captured images from every part of the building using digital cameras and a drone. The fieldwork took two people approximately 43 days to complete the data capture. Fieldwork time for traditional photogrammetry and NUPM is the same.

We used projecting artificial patterns, dual imaging, and laser scanning technology to capture images of textureless walls techniques (Ahmadabadian et al. 2019, Santosi et al. 2019, Hafeez et al. 2023, Alshawabkeh et al. 2021). We utilized minimum DOF (Depth of Field), dual imaging, variable background, textureless background, powder coating, and masking techniques to photograph reflective and transparent surfaces (Palousek et al. 2015).

In total, we captured an estimated 30,000 images using different cameras. We used a Nikon D810 with a 24 mm lens for rooms with decorations and a 50 mm lens for detailed decoration and paintings; a Nikon D3500 with an 18 mm lens for less decorative and small rooms, staircases, and corridors; and a Phantom 4 pro with a 24 mm lens for façades, roofs, and terrace floors.

Images were screened and grouped by datasets. Dataset A, containing 8,450 photos, was the largest independent dataset in this project and represented the building’s exterior, including the main façades, balconies, terraces, roofs, and porches. Dataset B, containing 2,145 photos, was the largest dataset of indoor spaces, which included four rooms connected by three staircases. Dataset C, which contained 1,482 photos, represented the main hall; Dataset D, which contained 1,161 photos, represented the king’s room; Dataset E, which contained 733 photos, represented the lounge room; and Dataset F, which contained 721 photos, was the guest bedroom.

2.2 General Processing

Our consumer-level computer specifications for processing were Intel Core i7 8700K, NVIDIA GeForce GTX 1650 Super, and 32 GB ADATA RAM. Camera alignment requires less RAM than other processing tasks and is not a limiting step (Agisoft Development Team, 2014. Rather, the main limiting tasks of processing are generating point clouds and texture. The NUPM overcomes RAM limitations and was tested on Agisoft Metashape v1.8. The settings and outcomes could be slightly different in other versions and different software. Processing the model of Shams-ol-Emareh took approximately 3 months, of which we estimate 9 days were used specifically for applying the NUPM for point cloud, mesh, and texture generation. However, processing these steps on such a large dataset with traditional photogrammetry processing could not have been done with the RAM limitations on our computer. Alignment time was the same for NUPM and traditional photogrammetry.

2.3 Non-Uniform Processing Method (NUPM)

The basic concept of the NUPM is to subdivide input data into smaller fragments in each step based on the level of detail so that consumer-grade computers can handle the data processing. For Shams-ol-Emarah, we carried out each process according to its level of importance and surface type. Next, we applied edits to reduce noise and errors. The following step involved decreasing the size of the generated file by reducing the number of points or triangles based on object’s level. The advantage of the NUPM is that every part of the data or object can be reduced to a specific amount, independent of the position, which the photogrammetrist decides.

We performed the whole processing procedure in five main steps: alignment, point cloud generation, mesh generation, texture generation, and tiled model generation. The sequence of sub-procedures was not the same in all steps. Figure 2 presents the procedure flowchart for implementing the project.

The first and easiest way to fragment the Shams-ol-Emareh building into smaller pieces was to separate independent spaces and façades to process each one individually. An example of an independent space was an indoor room isolated from other rooms by closing all doors and windows. Each independent space had an independent dataset. All the independent spaces were processed and modeled separately and placed in their exact position with the help of the control points and the coordinate system.

Our consumer-grade computer succeeded in aligning each of these datasets. However, these datasets were too large to be processed in high-quality settings to generate point clouds, meshes, and high-resolution textures, therefore requiring the NUPM to achieve the desired results. We checked the accuracy and correctness of the position of the photos with the help of the checkpoints, and we fixed inaccurate images by realigning them or removing them from the processes.

2.3.1 Non-Uniform Processing of the Point Cloud

After alignment of all the rooms, the second step was to generate non-uniform point clouds. We achieved this by fragmenting each room into pieces by region boxes and generating smaller point clouds separately and in various qualities. We then joined the disparate point clouds together to create a unified point cloud of the whole room. After this, we non-uniformly fragmented the integrated point cloud for the

second time based on smoothness, then edited and decimated, and joined

Step 1: Assigning Quality Level of Processing. There is no need to process all parts of a room in maximum quality. Each room is divided into smaller parts with a region box. Each part is assigned a level of processing from lowest to highest. This level is determined by factors such as historical significance, artistic importance, surface roughness, details, and size of different parts of the room. In our case study, this meant that only important, detailed, and rough parts of the room, such as decorated plastering, artistic doors and windows, statues, carvings, and candlesticks were processed in higher quality. Conversely, less important and undetailed parts, such as undecorated walls, flat floors, and smooth ceilings, received a lower quality processing resolution.

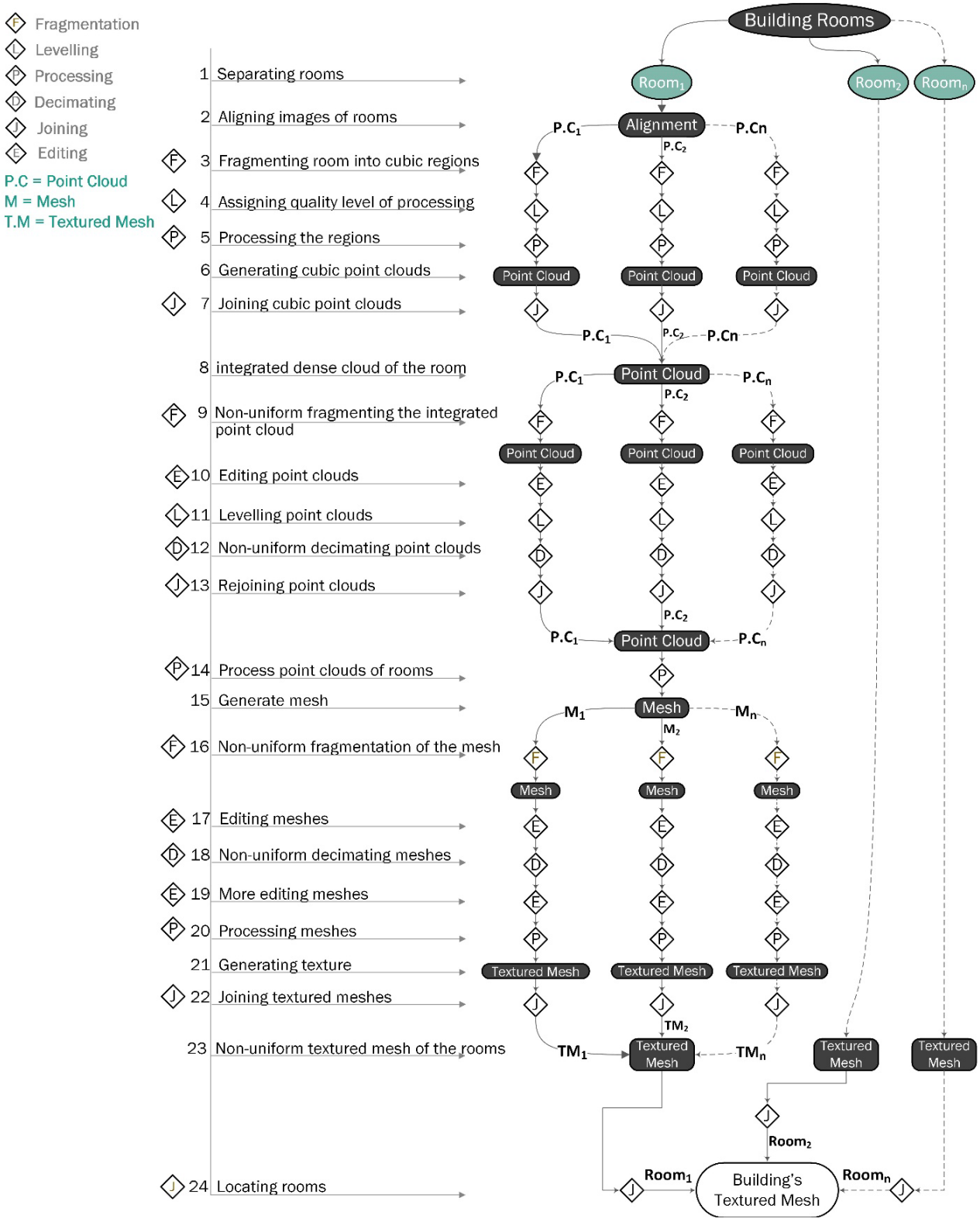

Figure 2. Flowchart of the project implementation process.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-M-2-2025-293-2025 | © Author(s) 2025. CC BY 4.0 License.

Step 2: Generating Cubic Point Clouds. Once all parts of a room were bound inside a region box, a cubic point cloud was generated for each part based on its level of process quality. This helped to reduce RAM requirements and output file size. We chose the position and size of the region boxes so that they covered all parts of the rooms with little overlap between them.

Step 3: Joining and Fragmenting Point Clouds. The point clouds were merged to create a unified point cloud for each room. The unified point cloud had a non-uniform density of points because of the different levels of processing quality. This technique not only helped to reduce the size of the point cloud and processing time but also decreased the computer requirements. If more detail is required, the integrated point cloud can be fragmented again into selective, detailed, smaller fragments that are no longer cubic.

The point clouds must be fragmented based on smoothness. The flatter the surface of an object in real life, the less dense it needs to be in the point cloud. Thus, we selected all the fully flat surfaces of Shams-ol-Emareh, such as walls, ceilings, roofs, or the flat parts of doors or closets, and extracted these into their own fragments to be reduced in density. This process continued with other groups of points until the last remaining group was composed of the roughest surfaces and most elaborate details. The edges and break lines of walls belong to the roughest group.

Step 4: Editing Point Clouds. The point cloud fragments were smaller and represented a semi-uniform level of detail. We edited the point clouds for noise. Filtering points by their confidence was a helpful option to select inaccurate points and delete them or to decrease their density. Each point was assigned a confidence value between 1 and 255.

Step 5: Levelling and Decimating Point Clouds. After editing the point clouds, we assigned them a level depending on the complexity of the surface to determine decimation level and reduce density (Figure 3). The Filter Point Cloud option decimates the point cloud according to the user-defined point spacing distance. We assigned the point cloud with the flattest surface the maximum decimation (Level 7), resulting in the lowest density of points, while we gave the minimum decimation (Level 1) to the most detailed, jagged, and decorated surfaces to retain the highest density of points.

Step 6: Rejoining Point Clouds. When all point clouds were reduced in density, they were merged to create a unified point cloud of the room with non-uniform density. A reduction in density equates to a smaller file size.

2.3.2 Non-Uniform Mesh

The main objective for the mesh was to ultimately generate a texture with optimum resolution. For each room, the mesh was generated based on the point cloud rather than directly from depth maps because the workflow used assumes point cloud editing before mesh reconstruction. Since RAM is not a limiting factor for mesh generation, higher quality settings can be used to obtain more detailed and accurate geometry. However, this requires a longer time for processing, creates an unnecessarily large file, and hinders the generation of an optimum texture. We needed to edit the mesh of Shams-ol-Emareh to improve the quality of the model, and the number of polygons needed to be reduced to decrease the file size.

The resolution of the final texture for the whole room needs to be non-uniform to avoid unnecessary pixels that increase output file size. Most importantly, there is a memory limitation for generating high-resolution texture. As such, the generated mesh was fragmented, edited, decimated, optimally texturized, and then rejoined like a puzzle.

Step 1: Non-Uniform Fragmentation of the Mesh. Our main objective for breaking a mesh into smaller fragments was to generate textures for each fragment with various resolutions based on the texture’s importance and detail of the texture. This procedure is like the third step of non-uniform point cloud processing, where different parts of the room are extracted, but this procedure differs in its selection criteria. Unlike the point cloud, which is fragmented based on the smoothness of the surfaces, the mesh is fragmented based on the necessary detail of the texture for the area of the room. For example, an intricate

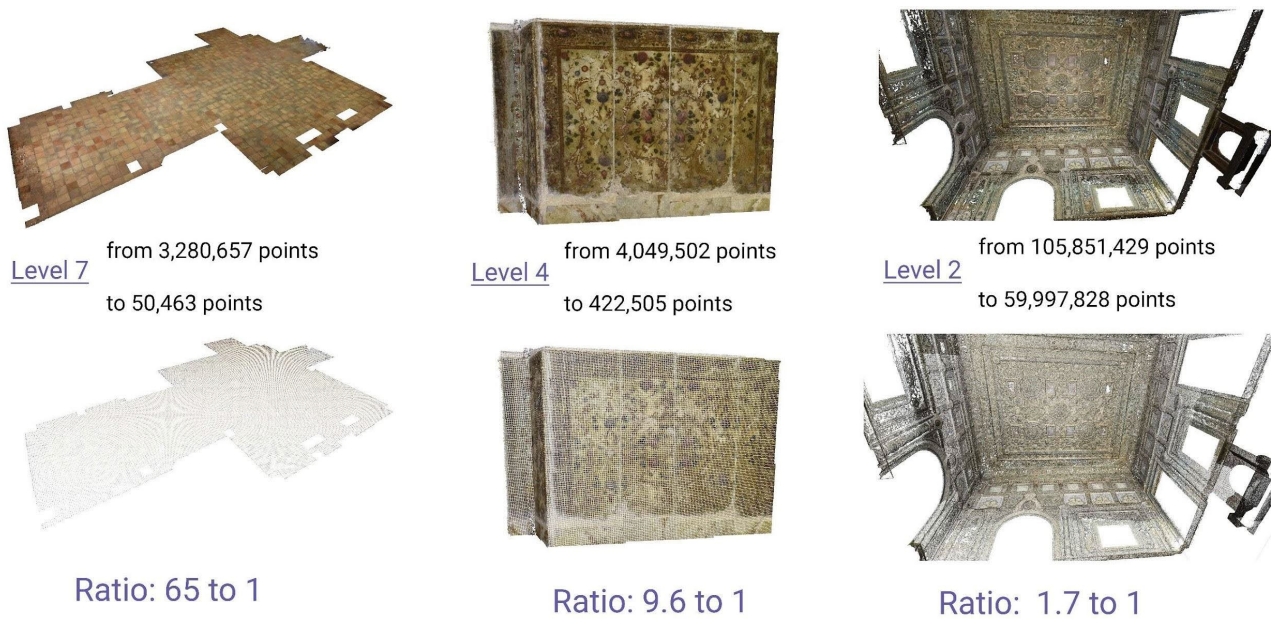

Figure 3. Decimated point clouds and results of the decimation ratio based on the level of smoothness.

painting on a flat surface is much more important than a jagged surface with unimportant texture like a column or a texture-less plaster; hence, the mesh of the painting receives a higher resolution of texture than the column or plaster. In Metashape v1.8 there is no option to select a group of points for decimation; however, there is an option to decimate the mesh for selected faces. This procedure also circumvented RAM limitations for decimating large numbers of faces. The amount of memory required for model decimation depends on the initial polygon count, not the target face count.

Step 1: Editing Meshes. Editing the fragmented meshes included deleting or replacing redundant parts not attached to the main mesh, smoothing meshes, closing holes, sharpening edges, enabling and disabling images to adjust the texturing process, masking images and calibrating image colors. To edit and smooth noisy parts on flat surfaces like walls or mirrors and glass, we selected those triangles and used the Smooth Mesh option, checked the Apply for Selected Faces option, and set a high number for the strength. When some of the selected triangles were placed on the borders of the mesh, we checked the Fix Borders option. If the surface was so noisy that this option did not solve it, then we deleted the selected area and filled the empty part using the Close Holes option and checked the Apply to Selection option.

Step 3: Decimating Meshes. All the fragments of the mesh were decimated into a triangle count that reduced the file size. As breaking the mesh was not based on the smoothness of the surfaces, there were different room components with varying surface smoothness in a single model. Although most photogrammetric software can decimate a mesh to some extent with minimal deformation of the model, high levels of decimating a whole mesh can still result in data loss in rougher sections and eliminate the sharpness of the edges (Farella et al. 2022, Verhoeven et al. 2017, Gotsman et al. 2002). Therefore, triangles of smooth surfaces on the mesh were selected and reduced by the Decimate Mesh for Selected Faces option. Unlike decimating point clouds, where the density was reduced according to level, in the Decimating Mesh step it was possible to decimate selected groups of triangles independently to a specific target triangle count, resulting in greater reduction than the point cloud step (Figure 4).

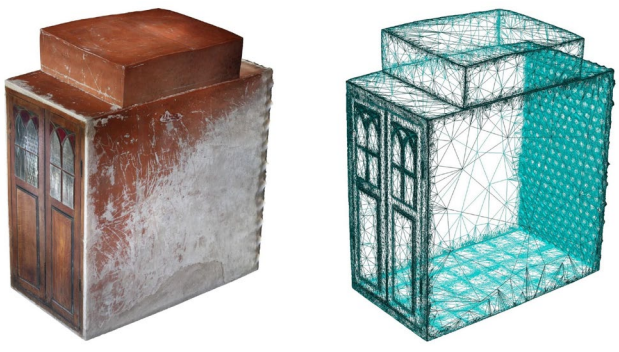

Figure 4. 3D model of a room in Shams-ol-Emareh showing the various triangle sizes based on the smoothness of surfaces. No reduction is evident in the detail of the break lines like door and wall boundaries and frontiers between floor tiles (right image).

Step 4: More Editing of Meshes. When we decimated meshes in flat surfaces, those parts were not well smoothed. Hence, before generating a high-resolution texture for the non-uniform mesh, we edited them again and slightly flattened the decimated surfaces. The edges and break-lines became sharper by using the Smooth Mesh Option at low strength and checking the Fix Borders and Preserve Edges options. These measures were taken to sharpen the model’s edges and smooth the flat surfaces. Checking the Fix Borders option keeps the triangles on the borders of the meshes unchanged, and when joining the meshes in the next steps, it preserves gaps or overlaps between them.

2.3.3 Non-Uniform Texture

The maximum resolution of the texture is equal to the resolution of the best image, or the one that shows the most detail. Generating a unified texture of optimum quality requires setting the texture size to a high number. With these datasets, generating texture with maximum resolution in a single process is impossible on most consumer-grade computers with limited RAM. Exporting texture to several files allows for greater resolution of the final model texture, while export of high-resolution texture to a single file can fail due to RAM limitations. One objective of this project was to reduce the final file size, and therefore reducing texture size was necessary. However, reducing texture size the standard way results in data loss in important sections. Using NUPM for texture resulted in a final model with a non-uniform texture resolution and a decreased file size.

A texture was generated for each mesh fragment in various resolutions. Walls or sections without texture or patterns or with less important surfaces were processed in low resolution, while important and well-textured sections, such as paintings and decorations, were processed in higher resolution. By enabling only photos captured with Nikon D810 and 50 mm lens, we reduced the RAM requirement, increased processing speed, and improved texture quality for fragments that included important textures, such as paintings and decorations. These images were captured at a perpendicular angle to surfaces that were important in terms of texture. The Nikon D810 has a larger sensor size, higher dynamic range, higher sensitivity, lower noise levels, and higher resolution than Nikon D3500. In addition, the longer camera lens makes distant objects appear magnified. As a result, images captured by the Nikon D810 and 50 mm lens had better quality than the other image groups. Large overlaps between these images were not necessary as the main purpose of capturing them was generating textures, not point clouds or mesh. This significantly reduced the number of photos involved in the processing step of generating texture for those fragments, thereby resulting in a considerably lower RAM requirement and faster processing. The generated texture also had a higher quality and resolution due to the use of a better camera and magnified images. This technique was applied for the areas with the most important texture.

For the less important areas, we applied another technique to reduce the RAM requirement and increase processing speed when generating textures. We disabled some images that were similar to their neighbors. In photogrammetry, captured images have substantial overlaps that are needed for the 3D reconstruction of objects (Róg and Rzonca 2021); this means that every part of an object is seen in a few images, but it is not mandatory that all of them be involved in generating texture. Thus, we disabled some of them while generating textures. For some fragments, we disabled every other image until 50% of images remained enabled. For some flat surfaces, we disabled more than 70% of images because they had too much overlap. All the fragmented textured meshes of a room were then merged like a puzzle to create a unified, non-uniform textured model

with sharp edges. The final file size was thus significantly decreased without any meaningful loss of data in meshes and texture. Furthermore, all textured models of the rooms and the building’s exterior were in the right position due to their integration in the coordinate system.

3. Results

There were two main objectives of this research: first, to overcome the limitations of processing large datasets using consumer-grade computers; and second, to reduce the file size of the output data to facilitate storing, sharing, and visualization. The NUPM we propose was used on approximately 40 independent spaces in the Shams-ol-Emareh building, but we selected two of them to demonstrate the results at different processing stages. The first is the lounge (Dataset E), which displays maximum variety in the level of artwork, decorations, and surface roughness. The second is the multi-space room (Dataset B) with the largest number of images captured to compare output size of the traditional method versus the NUPM.

After generation of the non-uniform point cloud, mesh, and texture, we needed to compare these with data produced via traditional methods. However, generating a high-quality unified point cloud for Dataset E and Dataset B was beyond the capabilities of our consumer-grade system. To overcome this obstacle, we generated cubic point clouds and merged them together.

3.1 Non-Uniform Point Cloud

In the non-uniform point cloud, the density of points and the space between them were not fixed values, unlike point clouds produced by traditional uniform methods. The result of building the point cloud non-uniformly is that the parts of the room with rough and detailed surfaces contain denser points, while flat and undetailed parts have lower point density (Figure 5).

3.2 Non-Uniform Mesh-Triangles

The non-uniform point cloud was used to produce the mesh, and the mesh was edited to create the non-uniform mesh. Like the non-uniform point cloud, the non-uniform mesh has various triangle densities and sizes (Figure 4). Only a few triangles are necessary to represent a flat surface in a mesh. Traditional mesh generation of walls and floors would have resulted in numerous small triangles and large file sizes. The NUPM reduced the number of triangles for these surfaces considerably without a meaningful loss of data, whereas the triangles on more detailed surfaces underwent less reduction to maintain their geometry.

3.3 Non-Uniform Texture

After the non-uniform mesh was complete, we generated the texture. The texture was also built non-uniformly, which means that visually important and detailed surfaces (e.g., paintings and decorations) had much smaller pixel sizes than the textureless and unimportant features of the room (e.g., walls, mirrors, and modern construction). The texture of newly painted walls and mirrors had a 1.8 cm pixel size, while embossed motifs had 0.249 mm pixel size, and plaster had 0.4 mm pixel size, showing pixels of important features were 72.3 times and 45 times smaller less important features (Figure 6).

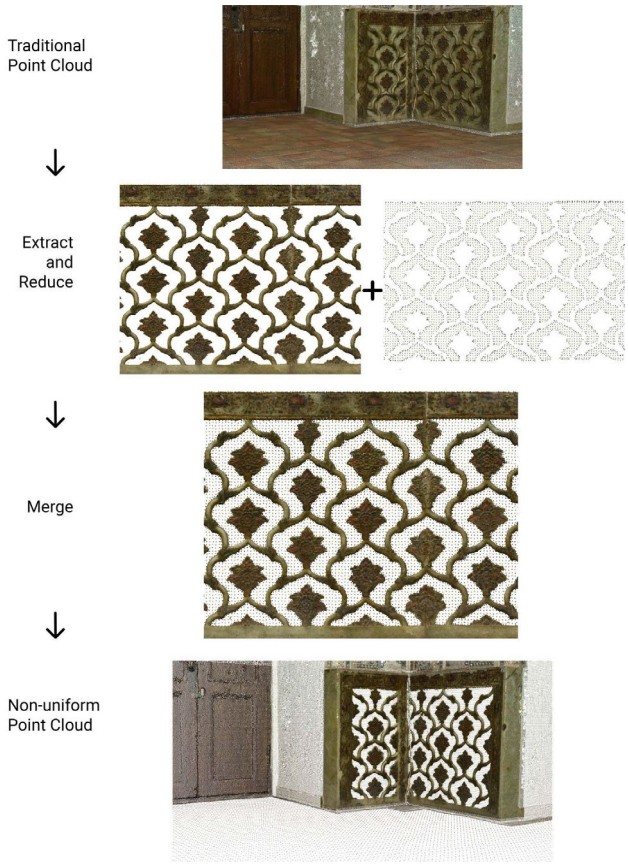

Figure 5. The point cloud generated via the traditional method (top) undergoing the NUPM (bottom).

Figure 6. The non-uniform pixel size of 1.8 cm for newly painted walls and mirrors (top and bottom), 0.249 mm for embossed motif (top), and 0.4 mm for plaster (bottom).

Table 1. Comparison of traditional photogrammetry and the NUPM in terms of number of points and triangles and output file sizes of various common formats.

	Traditional (a)	NUPM (b)	Difference (a – b)	Ratio (a/b)	Percent Reduced (1 - b/a)
Number of points	268,030,955	27,835,320	240,195,635	9.63 to 1	**89.62%**
Number of triangles	38,041,522	1,000,728	37,040,794	38.02 to 1	**97.37%**
TXT file size (point cloud)	10,200 MB	990 MB	9,210 MB	10.31 to 1	**90.3%**
LAS file size (point cloud)	6,480 MB	690 MB	5,790 MB	9.4 to 1	**89.36%**
OBJ file size (mesh)	1,610 MB	39 MB	1,571 MB	41.29 to 1	**97.58%**
FBX file size (mesh)	522 MB	14.7 MB	507.3 MB	35.52 to 1	**97.19%**

3.4 Output Data

As mentioned in the introduction, decreasing output file size without meaningful loss of data was one of our main objectives. The NUPM generated non-uniform point clouds, meshes, and textures that significantly decreased output file size by decimating the number of points, triangles, and pixels with the lowest loss of data quality.

Table 1 presents a comparison between traditional photogrammetry results and the NUPM results for the largest indoor space (Dataset B). The NUPM yields an 89.62% reduction in points for the point cloud and a 97.37% reduction in triangles for the mesh. Furthermore, the NUPM reduced the output file size of the point cloud by about 90% and the mesh by about 97% compared to the traditional method. The NUPM allows control in every stage of the photogrammetry process, which helps to generate better-quality 3D models (Figure 7).

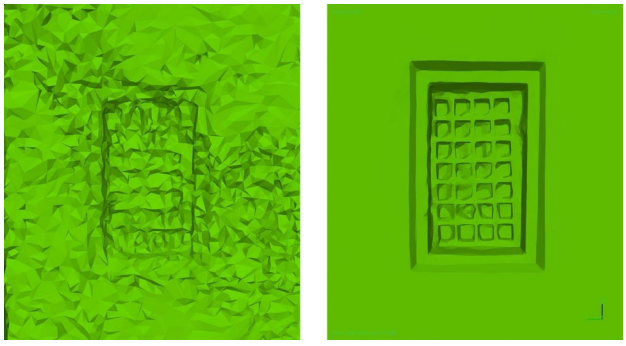

Figure 7. A window processed by the traditional method (left) and the NUPM (right). The NUPM produced an output with less noise and more detail.

4. Discussion and Conclusion

The results demonstrate that the NUPM is an effective workflow to process large datasets of photogrammetry imagery without a meaningful reduction in the quality of output data using a consumer-grade computer with a limited RAM. Instead of fragmenting objects based on their position using a region box, we split objects into smaller fragments non-uniformly based on the importance of the parts of the object in each step (generating point cloud, mesh, texture). This gave us more control over the model, particularly the ability to decide the level of processing settings, density, detail, texture resolution, smoothness, file size, space occupied, precision, and quality for every single part of the object individually. In this way, the important parts of an object can be processed in higher quality with more density, resulting in more details.

The main limitation of the NUPM is that processing is manual and time-consuming. It involves numerous additional steps beyond the traditional photogrammetry workflow that increase processing time. However, many large datasets cannot be processed in the traditional way. Furthermore, the results featured a significant reduction in output data size for point clouds, mesh models, and texture. The reduced data size also alleviated problems related to storing, managing, manipulating, analyzing, transferring, visualizing, and rendering large files of photogrammetry outputs for traditional hardware, software, and database technologies.

The non-uniform method has been tested on the historical buildings of Shams-ol-Emareh, but we believe this method has the capacity to be expanded and used in other aerial and close-range photogrammetry areas beyond cultural heritage. Since the main procedures for 3D reconstruction of modern buildings, smaller objects, historical sites, and the surface of a ground are almost identical, it would be feasible to employ this technique to process generic large datasets. The NUPM can even be tested on satellite imagery and super-large datasets where the aligning step is also a limiting task for the computer system.

This case study demonstrated the benefits of non-uniform processing for generating non-uniform point clouds, meshes, and textures. However, it is possible to expand this idea to non-uniform data-gathering, such as photographing, target points distribution, and camera resolution, to prepare raw photogrammetry data. We did this during data capture of this project when we took fewer but higher quality photographs intended for use in texturizing. Another possibility is in drone control apps. If the distances between images or flight lines are not fixed and instead change from station to station or line to line non-uniformly based on roughness, texture, or importance of the ground surface, non-uniform data gathering would help decrease the number of images and flight time. Regardless of whether a user has computational limitations, the NUPM can still be beneficial for most datasets because of its reduced output file size and better-quality data output.

Acknowledgements

Our sincere thanks to Afarin Emami, Director of Golestan Palace World Heritage Complex, and her supportive staff for

making this research possible. We acknowledge Andrea Jalandoni’s Australian Research Council Discovery Eary Career Fellowship (DE240100030) and the Griffith Centre for Social and Cultural Research for partial funding which was crucial to our exploration. Special appreciation to Farshad Salehi for invaluable assistance during data gathering and to Sue Jarvis for her skilled copyediting.

References

Adrov, V. N., Drakin, M., & Sechin, A.Y., 2012. High performance photogrammetric processing on computer clusters. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XXXIX-B4, 109–112. doi.org/10.5194/isprsarchives-xxxix-b4-109-2012.

Ahmadabadian, A. H., Karami, A., & Yazdan, R., 2019. An automatic 3D reconstruction system for texture-less objects. *Robotics and Autonomous Systems*, 117, 29–39. doi.org/10.1016/j.robot.2019.04.001.

Alshawabkeh, Y., Baik, A., & Miky, Y., 2021. Integration of laser scanner and photogrammetry for heritage BIM enhancement. *ISPRS International Journal of Geo-information*, 10(5), 316. doi.org/10.3390/ijgi10050316.

Agisoft Development Team, 2014. Agisoft Photoscan – Metashape. www.agisoft.com/pdf/tips_and_tricks/PhotoScan_Memory_Requirements.pdf (9 July 2025).

Anders, N., Valente, J. P., Masselink, R., & Keesstra, S., 2019. Comparing filtering techniques for removing vegetation from UAV-based photogrammetric point clouds. *Drones*, 3(3), 61. doi.org/10.3390/drones3030061.

Bariami, G., Faka, M., Georgopoulos, A., Ioannides, M., & Skarlatos, D., 2012. Documenting a UNESCO WH Site in Cyprus with Complementary Techniques. *International Journal of Heritage in the Digital Era*, 1(1_suppl), 27–32. doi.org/10.1260/2047-4970.1.0.27.

Farella, E. M., Morelli, L., Rigon, S., Grilli, E., & Remondino, F., 2022. Analysing key steps of the photogrammetric pipeline for museum artefacts’ 3D digitisation. *Sustainability*, 14(9), 5740. doi.org/10.3390/su14095740.

Gniady, T., Ruan, G., Sherman, W. R., Tuna, E., & Wernert, E., 2017. Scalable Photogrammetry with High Performance Computing. *Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact*, 1–3. doi.org/10.1145/3093338.3104174.

Gotsman, C., Gumhold, S., & Kobbelt, L., 2002. Simplification and compression of 3D meshes. *Tutorials on multiresolution in geometric modelling: summer school lecture notes*. Berlin, Heidelberg: Springer Berlin Heidelberg, 319–361. doi.org/10.1007/978-3-662-04388-2_12.

Hafeez, J., Jeon, H., Hamacher, A., Kwon, S., & Lee, S., 2023. The effect of patterns on image-based modelling of texture-less objects. *Metrology and Measurement Systems*, 25(4), 755–767. doi.org/10.24425/mms.2018.124883.

Haubt, R. A., & Jalandoni, A., 2019. Optimizing the potential of research data through an integrated data management approach: Considering research method, data life cycle, big data and linked data in a research example in Australian rock art. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci*, IV-2/W6, 77–82. doi.org/10.5194/isprs-annals-iv-2-w6-77-2019.

Li, S., Dragicevic, S., Castro, F. A., Sester, M., Winter, S., Coltekin, A., Pettit, C., Jiang, B., Haworth, J., Stein, A., & Cheng, T., 2016. Geospatial big data handling theory and methods: A review and research challenges. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115, 119–133. doi.org/10.1016/j.isprsjprs.2015.10.012.

Peña-Villasenín, S., Docampo, M. L. G., & Sanz, J. O., 2020. Desktop vs cloud computing software for 3D measurement of building façades: The monastery of San Martín Pinario. *Measurement*, 14. doi.org/10.1016/j.measurement.2019.106984.

Palousek, D., Omasta, M., Koutny, D., Bednar, J., Koutecky, T., Dokoupil, F., 2015. Effect of matte coating on 3D optical measurement accuracy. *Optical Materials*, 40, 1–9. doi.org/10.2478/amtm-2019-0005.

Martinez-Rubi, O., Nex, F., Pierrot-Deseilligny, M., & Rupnik, E., 2017. Improving FOSS photogrammetric workflows for processing large image datasets. *Open Geospatial Data, Software and Standards*, 2(1). doi.org/10.1186/s40965-017-0024-5.

Morales, R., Wang, Y., & Zhang, Z., 2010. Unstructured point cloud surface denoising and decimation using distance RBF K-Nearest Neighbor Kernel. In Qiu, G., Lam, K.M., Kiya, H., Xue, XY., Kuo, CC.J., Lew, M.S. (eds) *Advances in Multimedia Information Processing-PCM 2010*. Lecture Notes in Computer Science, vol 6298. Springer Berlin Heidelberg, 214–225. doi.org/10.1007/978-3-642-15696-0_20.

Moussa, W., Wenzel, K., Rothermel, M., Abdel-Wahab, M., & Fritsch, D., 2013. Complementing TLS Point Clouds by Dense Image Matching. *International Journal of Heritage in the Digital Era*, 2(3), 453–470. doi.org/10.1260/2047-4970.2.3.453.

Sanchita, N. & Ashwini, P., 2023. Use of big data in the cloud computing. *International Journal for Research in Applied Science and Engineering Technology*, 11(9), 984–987. doi.org/10.22214/ijraset.2023.55619.

Remondino, F., Nocerino, E., Toschi, I., & Menna, F., 2017. A critical review of automated photogrammetric processing of large datasets. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-2/W5, 591–599. doi.org/10.5194/isprs-archives-xlii-2-w5-591-2017.

Róg, M., & Rzonca, A., 2021. The impact of photo overlap, the number of control points and the method of camera calibration on the accuracy of 3D model reconstruction. *Geomatics and Environmental Engineering*, 15(2), 67–87. doi.org/10.7494/geom.2021.15.2.67.

Santosi, Z., Budak, I., Stojakovic, V., Sokac, M., & Vukelić, Đ., 2019. Evaluation of synthetically generated patterns for image-based 3D reconstruction of texture-less objects. *Measurement*, 147. doi.org/10.1016/j.measurement.2019.106883.

Verhoeven, G. 2017. Mesh is more: Using all geometric dimensions for the archaeological analysis and interpretative mapping of 3D surfaces. *Journal of Archaeological Method and Theory*, 24(4), 999–1033. doi.org/10.1007/s10816-016-9305-z.