Cultural Heritage Assets Optimization Workflow for Interactive System Development

Kedong Chai
Design School
Xi'an Jiaotong-Liverpool University
Suzhou, China
Kedong.Chai21@student.xjtlu.edu.cn

Yue Li
Department of Computing, School of Advanced Technology
Xi'an Jiaotong-Liverpool University
Suzhou, China
Yue.Li@xjtlu.edu.cn

Fig. 1: Optimization demonstration using the *Figure of a Seated Court Lady* as an example. (a) Photography image; (b) Meshes of the photogrammetry scanned high-poly model; (c) Meshes of the retopologized low-poly model; (d) 3D rendering of the photogrammetry scanned model; (e) 3D rendering of the retopologized model; (f) 3D rendering of the photogrammetry scanned model with texture; (g) 3D rendering of the retopologized model with baked texture; (h) Detailed maps of generated for interactive system development.

***Abstract*—An increasing number of reconstructed digital assets are being created worldwide to preserve cultural heritage. These assets can be used in interactive systems such as augmented reality (AR) and virtual reality (VR) to provide effective ways to access and learn about cultural heritage. One of the widely adopted reconstruction techniques is close-range photogrammetry. However, scanned models need to be processed and optimized before they can be used in interactive systems, which requires a series of retopology and baking work to reduce the size of models while maintaining visual fidelity. Nevertheless, manual retopology and baking are complex processes. An efficient optimization workflow is essential for the use of cultural heritage assets in interactive systems. This paper presents an optimization workflow for retopology and texture baking using free and open-source software. Evaluations show that the workflow demonstrates its strengths in its high efficiency, versatility, learnability, and low cost. This work contributes insights to researchers and practitioners in the field of cultural heritage.**

***Index Terms*—digital heritage; interactive system; VR; AR; 3D modelling; retopology; texture baking; photogrammetry**

I. Introduction

The digitization of museum objects, monuments, and historical sites has become an important trend in the field of cultural heritage [1]. Digital technologies are increasingly used for cultural heritage protection. Digitization and the generation of photorealistic 3D models allow for free and continuous worldwide access to monuments, despite distances and different time zones [2]. Recent research in the field of cultural heritage digitization are developing rapidly in drone-based 3D measurement and techniques, augmented and virtual reality visualization, and virtual museums. However, there are still some factors that hinder the effective use of 3D digitized assets for the design and development of interactive systems. Lack of technical expertise and support is one of the biggest obstacles to the digitization of cultural heritage [3]. This demonstrates the need for an efficient workflow to overcome these barriers and advance the development of digital heritage.

High-precision scanning techniques are often used to scan and reconstruct digital assets in 3D. The scanned models often have a large number of polygons (see Fig. 1b), which makes it necessary for the scanned models to be optimized to low-poly models that retain high-precision model details (see Fig. 1c and 1g). Post-processing and optimization is important for the use of reconstructed models in interactive systems, such as virtual museums and multimedia guide systems. Traditional manual topology often requires strong expertise in 3D modelling to manually edit the vertices and edges, which is time-consuming and labor-intensive. Meanwhile, the baking of texture maps determines the fidelity of visual experiences and has significant influence on users' perceived presence when interacting with the virtual objects. Therefore, this paper proposes a retopology and texture baking workflow using open-source software *Instant Meshes*, *Blender*, and *Materialize* as the main tools. This low-cost solution greatly improves the efficiency and reduces the demand of technical expertise in the optimization of scanned cultural heritage assets.

This paper is based on the author's earlier work at the NVIDIA Joint-Lab on Mixed Reality, University of Nottingham Ningbo China. This work receives financial support from the Xi'an Jiaotong-Liverpool University (RDF-20-02-47 and TDF20/21-R22-142).

978-1-6654-8810-5/22/$31.00 ©2022 IEEE
DOI 10.1109/COMPSAC54236.2022.00290

II. Related Work

A. Digitization of Cultural Heritage

Museum is not only a place for collecting and statically displaying cultural artifacts. It serves an important mission as a platform for communication, academic research and education. Compared with the traditional museum communication approach through physical objects and static labels, more interactive communication and storytelling methods based on modern digital media technologies such as virtual reality, augmented reality, cloud computing and the Internet of Things are integrated into today's museum practices.

The digital conservation of cultural heritage through technical means is a beneficial supplement to traditional protection methods. Through 3D scanning, reconstruction, modeling and interactive system development, digital assets provide new possibilities for the communication of cultural heritage. Digital cultural heritage assets can be used for preservation, dissemination, and promotion of historical and cultural heritage worldwide. Previous research has investigated extensively on approaches to reconstructing digital assets [4], [5]. However, the digital assets should be used for more than static display. Visitors' learning of cultural heritage can benefit from active experimentation and concrete experience in the physical and sociocultural contexts [6], [7]. Effective storytelling through interactions is the important next step.

B. Photogrammetry and Optimization

3D data can be acquired in a number of ways, with photogrammetry being one of the main methods widely used in the field of cultural heritage. Photogrammetry is an effective technique for processing image data, providing accurate, metric, and detailed 3D information at different scales, and estimating the accuracy and reliability of unknown parameters based on measured image correspondence [8]. It is useful for recording cultural heritage over a wide range of scales, from landscapes to small objects. Current commercial software such as *RealityCapture*, *Agisoft PhotoScan* and *Autodesk ReCap* supports the creation of high-quality 3D models from $\tilde{2}0$ images that cover details of an object from different perspectives.

Photogrammetry begins with the acquisition of 2D images that cover different perspectives of the target object, followed by camera calibration and orientation, 3D point cloud generation and alignment, structuring and meshing, and texture mapping to create a 3D model. This technique has been widely adopted in the digitization of museum assets. For example, Google Arts Culture has worked with the Palace of Versailles, one of the most visited museums in the world, on the digitization of its collections. 130,000 photographs were taken by cameras and drones, collecting a total of 4 terabytes of data and over 15 billion pixels of texture [9]. This data is used to build digital and virtual museums that are deployed on a variety of hardware devices and platforms such as smartphones and websites.

Photogrammetry provides a fast and low-cost approach to reconstructing high-quality 3D models of cultural heritage [4]. Previous work has explore effective workflow for reconstructing cultural heritage assets based on photogrammetry approach [5]. However, these models often have millions of polygons and require high-performance workstations if visual effects are to be processed in real-time. Real-time interaction and rendering are important for the communication and storytelling of cultural heritage assets. To optimize the high-quality models for it to be used in system development, especially for mobile devices with limited computing power such as [10], retopology and map baking are needed.

Manual retopology is one way to reduce the number of polygons and to optimize complex models. However, the disadvantage of this approach is that the process is tedious and time-consuming. The manual editing process consists of complex tasks that require significant human efforts. Minor changes could affect the subsequent work and use of the model. In addition, it is time consuming as it requires abundant repetitive work. Meanwhile, it requires considerable expertise in 3D software to carry out retopology and baking work. These have hindered efficiency in post-processing work and in turn the design of interactive cultural experience. An efficient and beginner friendly optimization workflow for photogrammetry scanned assets is of great significance.

III. Optimization Workflow for Interactive System Development

Here we summarize our practices and present an optimization workflow to process photogrammetry scanned cultural heritage assets (see Fig. 2). We use a model captured from the Metropolitan Museum of Art as an example to illustrate the workflow. The model was created in *RealityCapture* in 604 seconds, using 75 images taken with an iPhone X. We use three pieces of free and open-source software for the optimization: *Instant Meshes*, *Blender*, and *Materialize*.

A. Automatic Retopology with Instant Meshes

Instant Meshes[1] is an interactive meshing application developed by Jakob et al. [11], which can be used in Windows, Mac OS X and Linux systems. The aim of this step is to reduce the polygon counts and to reduce the size of the model.

Photogrammetry scanned models usually have irregular edges, which can be fixed with the configuration details in *Instant Meshes* (see Fig. 3). Models imported to *Instant Meshes* can be retopologized into isotropic triangular or quad-dominant meshes. *Instant Meshes* optimize the edge orientations and vertex positions in the output mesh, by applying a unified local smoothing operator [11]. The algorithm implemented in the software linearly with the input size, thus can support a fast and efficient processing of input surface representations with various types and sizes.

The following five steps are taken to obtain a retopologized mesh: (1) Import the scanned high surface count model into *Instant Meshes*; (2) Set the `Target vertex count`. Fig. 1c shows a model with 42.50 K target vertex count; (3) Click

[1] https://github.com/wjakob/instant-meshes

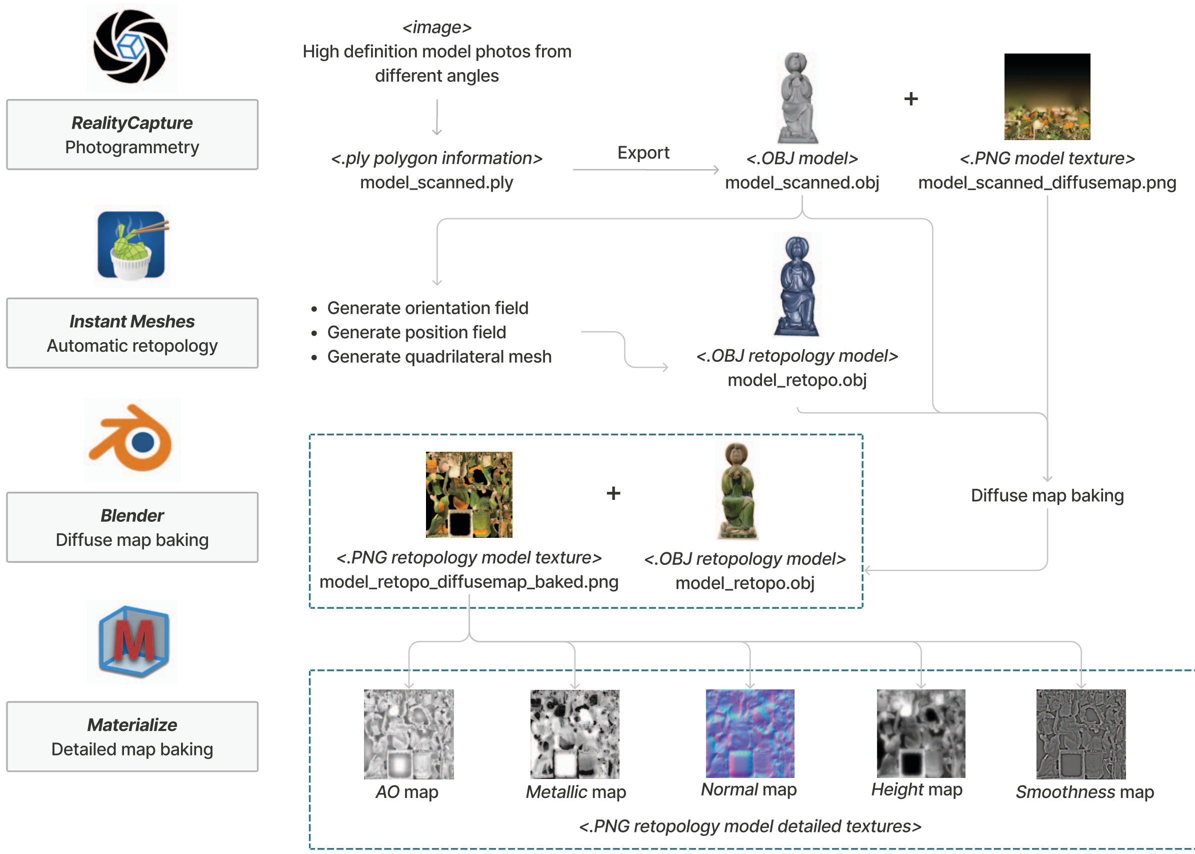


Fig. 2: Optimized retopology workflow for photogrammetry scanned cultural heritage assets.

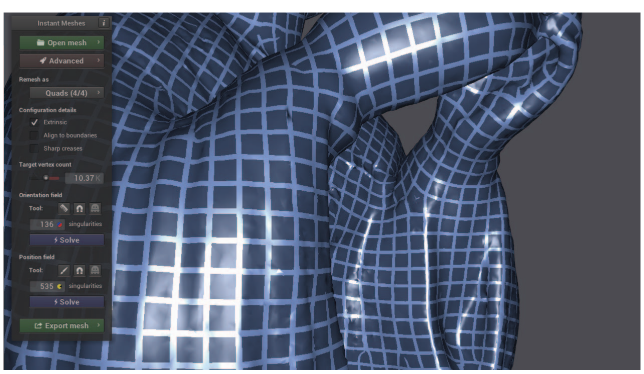


Fig. 3: *Instant Meshes* editing menus and the retopologized mesh.

on the `Solve` button under the `Orientation field` and wait for it to finish; (4) Click on the `Solve` button under the `Position field` and wait for it to finish; (5) Click on `Export mesh`, then `Extract mesh` and `Save`. The model should be manually named with the `.OBJ` suffix.

There are also some optional steps. For example, after solving the orientation field and the position field, one can select the first tools in the `Orientation field` (orientation comb) and `Position field` (edge brush) to make local adjustments, usually for the sharp edges. When exporting the mesh, one can check the `Pure quad mesh` under `Mesh settings` to generates a pure quadrilateral model, and to smooth the surface of the mesh by adjusting the `Smoothing iterations`. Fig. 1b-c shows a comparison of the original scanned model with 165,964 vertices and a size of 29.6 MB, and the retopologized model in isotropic triangular and quadrilateral mesh with 42,505 vertices and a size of 5 MB.

B. *Diffuse Texture Baking with Blender*

Blender[2] is one of the most popular 3D computer graphics software, which is free and open-source. It also supports Microsoft Windows, Mac OS X and Linux systems. The purpose of this step is to bake the color and pattern details from the scanned high-poly model to the retopologized low-poly model.

The steps below are followed in *Blender*: (1) Import the retopologized low-poly model into Blender and rename it to *low-poly*, clear the sharp edges under `Edit Mode` to avoid interference in texture baking; (2) Import the scanned high-poly model, rename it to *high-poly* and keep the position of both models unchanged to keep them in the same coordinate;

[2]https://www.blender.org/

(3) Select the *low-poly* model, go to `UV Editing` view and select `Smart UV Project` under UV Mapping (press `U` under `Edit Mode`) to obtain the UV coordinate information; (4) Under `Shading` view, create a new image texture. The larger the size, the more detailed the texture. This is the base texture of the retopologized *low-poly* model; (5) Switch to `Rendering` view to bake the texture. Select `Cycles` for the `Render Engine` and `CPU` as the `Device` (see Fig. 4a). Under `Bake` menu, select `Diffuse` for the `Bake Mode` and check only `Color` for the `Contributions` (see Fig. 4b); (6) Make a copy of the *low-poly* model to define the scope of the texture baking. Rename it to *cage*. Under `Rendering` view, navigate to `Bake` menu, check `Selected to Active` and `Cage`, select *cage* for the `Cage Object` placeholder. Define a `Cage Extrusion` to an appropriate distance at around 0.8 m (see Fig. 4b). This is to leave some space for texture baking; (7) Under `Layout` view, first select the scanned high-poly model and then the retopologized *low-poly* model, click on `Bake` under `Render Properties` to bake the texture; (8) Under `Shader Editor`, connect the low-poly model and texture (see Fig. 4c), the texture is baked to the retopologized model.

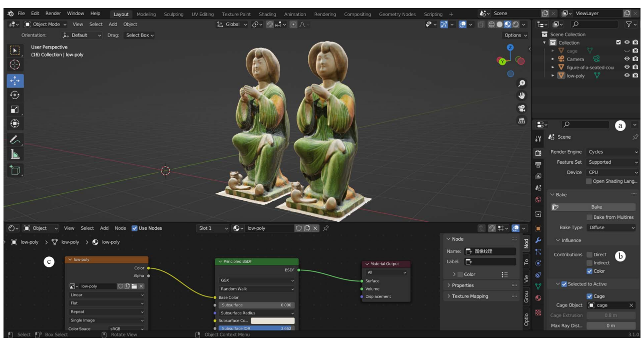


Fig. 4: *Blender* interface showing (a) Cycles render and diffuse map baking; (b) Cage extrusion settings for baking; (c) Shader settings.

C. Detailed Texture Baking with Materialize

Materialize[3] is a standalone tool that supports the generation of textures from a single image. It can be downloaded on a Microsoft Windows machine and used in *Unity* and *Unreal* game engines. This step aims to generate detailed textures for the retopologized model, including the normal, metallic, smoothness, height, and occlusion textures.

After the map baking in *Blender*, the color information in the scanned *high-poly* model is mapped to the retopologized *low-poly* model. In order to improve the visual fidelity of the model and the efficiency of graphics rendering, more detailed maps are needed. Fig. 5 demonstrates the effects of materials containing different combinations of detailed maps. *Normal map* is particularly important because it contains detailed information that simulates the lighting of bumps and dents of a surface. Other detailed maps will further enhance the visual effects. This provides a high level of details in the rendering of the model while taking a small amount of graphics rendering resources.

[3]http://www.boundingboxsoftware.com/materialize/

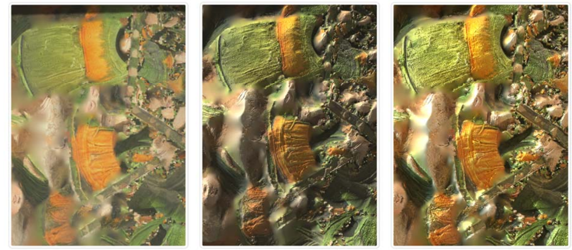


Fig. 5: Material effects with different texture information, simulated in *Materialize*. (a) *Diffuse* map only; (b) *Diffuse* and *Normal* maps; (c) *Diffuse*, *Height*, *Normal*, *Metallic*, *Smoothness* and *Ambient Occlusion (AO)* maps.

Texture baking in *Materialize* consists of the following steps: (1) Click `O` in the `Diffuse Texture` menu (see Fig. 6) to open and import the diffuse color texture saved in *Blender*. Under `Height Texture`, click `Create` to edit the settings and modify the texture details, and click `Set as Height map` to build the texture; (2) Repeat the previous step to generate the *Normal, Metallic, Smoothness,* and *AO map* in turn; (3) Select the `Preview` option in the menu of each texture to visualize the particular texture, and `Show Full Material` to see the combined rendered result; (4) Under the `Saving Options` menu, select `PNG` for the `File Format`. Click the `Save Project` button to export the texture maps.

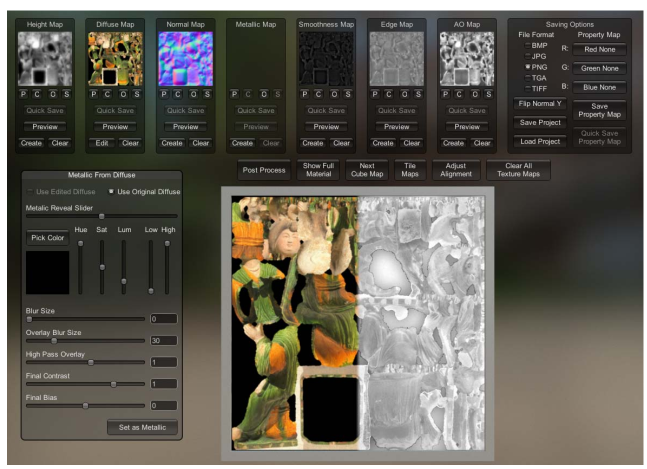


Fig. 6: *Materialize* editing menus and map baking results.

D. Review the Results in Unity

After the optimization of topology and texture maps, the results can be reviewed in *Unity* (see Fig. 7). The baked detailed maps can be applied to the placeholders in the material inspector. Our proposed approach supports the rendering of low-poly models with rich visual details and reduce the risk of delay in game play.

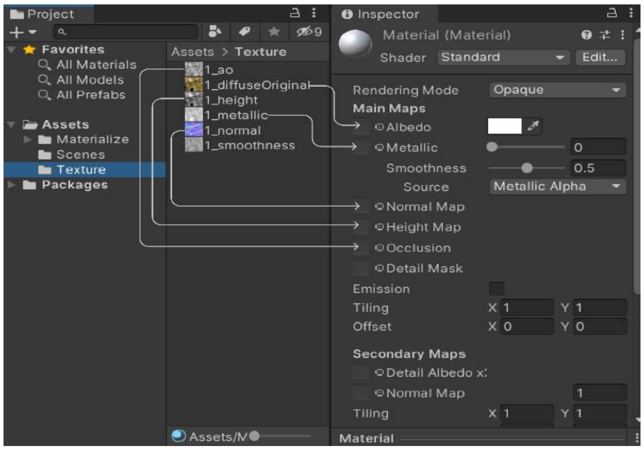


Fig. 7: *Unity* material inspector with placeholders for detailed maps.

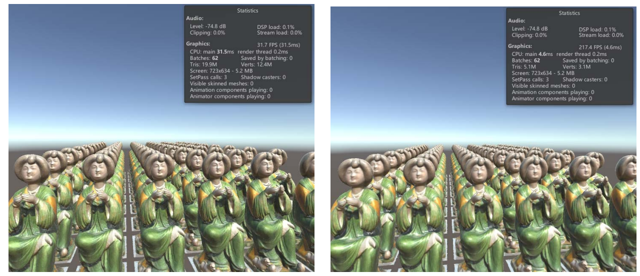
IV. Evaluation and Discussion

The design and development of interactive cultural heritage systems require 3D assets to be used in different platforms with different graphics rendering hardware. For mobile devices, high-poly models will consume large amount of computing resources for graphics rendering. The proposed workflow allows the adjustment of the size and quality of 3D models for them to be used in cross-platform and cross-device applications. Here, we describe a series of evaluations on the effectiveness of the workflow.

A. System Evaluation

We conducted a comparative study in two test scenes to evaluate the efficiency of the optimization workflow results. Both scenes are set up in the *Unity* default scene with a directional light and run on the same workstation. The workstation has an Intel i5 2.6 GHz CPU, NVIDIA RTX 3050 Laptop 4 GB graphics card, 16 GB of RAM, and 512 GB SSD.

Following the workflow, we optimized the model of the *Figure of a Seated Court Lady* from 29.6 MB to 5 MB. We import 60 photogrammetry scanned models to the controlled scene and 60 retopologized models to the experimental scene. The Rendering Statistics in *Unity* provides metrics for real-time graphics rendering information in the scene (see Fig. 8). FPS indicates the number of frames *Unity* is able to draw per second. The higher the FPS value, the smoother the motions and the less likely for the application to be choppy and slow. The CPU metrics indicate the total amount of time taken to process and render one frame. The higher the values are, the longer it takes to render. The controlled scene with scanned models has 31.7 FPS and a CPU metric of 31.5 ms; the experimental scene with retopologized models has 217.4 FPS and a CPU metric of 4.6 ms. The results showed significant improvements in both the FPS metric and the CPU metric, proving the efficiency of the optimization workflow.



(a) Controlled scene with 60 scanned high-poly models. (b) Experimental scene with 60 retopologized low-poly models.

Fig. 8: *Unity* Rendering Statistics showing real-time graphics rendering information.

B. User Evaluation

One student (male, aged 24) with 2 years of 3D modelling experience followed this workflow and optimized over 20 models for VR system development within an hour. This is the expected result for experienced users following this workflow.

To examine if the proposed workflow is easy to learn, we recruited 9 participants to attend an online workshop (3 females and 6 males, aged from 21 to 27, M=23.78, SD=1.93). Six participants had a minimal understanding of 3D software, including importing and exporting files and rotating and scaling 3D views. Three had only some exposure to quadrilateral modeling software and operations. None of them had any experience with the software used in the workflow.

After welcoming each participant, we explained the purpose of the workshop and showed them an instruction manual with the main steps to follow. Participants were asked to follow the instructed steps and process as many models as possible within 20 minutes. Most of the participants completed more than two models, and only one participant completed one model. In the interview, we asked participants "Are there any aspects that you found particularly easy or challenging in the process?" They reported that overall, the workflow is clear and easy to follow for novice users like them. Specifically, all of them found that the steps to follow in *Instant Meshes* and *Materialize* were straightforward, but the baking process in *Blender* was comparatively more challenging. We anticipate that users could reach increased efficiency as they get more familiar with the software and workflow. Meanwhile, this shows rooms for improvement in the texture baking process of the workflow.

C. Implications

Photogrammetry scanning is of low cost and has wide application prospects [12]. It does not require expensive radar equipment for data collection, as demonstrated by our example, high-precision model reconstruction can be done with as little as a smartphone. With necessary guide on how to take photos of an object with horizontal and vertical overlaps, the data capture can be completed by a novice user in a short period. However, the model generated by photogrammetry often has a high surface count and the retopology and map baking processes are often tedious and time-consuming. The proposed workflow addresses largely on the challenge with the post-processing of photogrammetry scanned cultural heritage assets. Optimized low-poly models with detailed texture maps can be of effective use in interactive systems to better present and communicate cultural heritage with the public, such as augmented reality and virtual reality applications on mobile devices. We discuss main advantages of the proposed method.

High efficiency. The workflow involves the use of four software. The entire optimization process can be finished by one person, or distributed among a team to accelerate the efficiency of optimization. Compared to the original scanned models, the processed models demonstrate significant improvement in size reduction, and the graphics performance in game engines. Regardless of the equipment and methods used for 3D reconstruction, post-processing of the digital assets is an indispensable part. The proposed workflow is not limited to photogrammetry scanned digitization, but applicable to optimization of topology and map baking in general.

High versatility. Unlike black-box automatic retopology, *Instant Meshes* provides a fully controllable way of mesh editing. It supports an automatic generation of simplified meshes and also allows manual corrections. As a result, it saves significant efforts than manual topology, and produces more stable results than fully automatic approaches. Similarly, the steps in *Blender* and *Materialize* are also give instantly visible results and and allow reversible actions. The steps described in the proposed workflow can be used partially if only a simplified mesh or a baked diffuse map is needed.

Easy to learn. The workflow does not require highly skilled people to operate complex equipment or software. It requires minimum practice to quickly master the workflow, as indicated in our evaluations. In addition, the software used in the method are open-source, free of charge, and has a friendly community for novice users. Being beginner friendly is important to the sustainable development for the reconstruction and digital preservation of cultural heritage.

Low cost. The proposed workflow is of low cost in terms of the hardware, software, and labor. Despite the worldwide efforts to promote the digitization of cultural heritage in recent years, the lack of expensive equipment and technical expertise has prevented some countries and regions from promoting the digitization of cultural heritage. Interactive systems such as social interfaces for cultural exhibitions [13] and multiuser VR and AR systems [10] are based on efficient digitization of cultural heritage. Digital assets combined with the optimization workflow proposed in this paper offer a cost effective solution for interactive cultural experiences.

D. Limitations and Future Work

This optimization workflow is summarized during our research practice on VR and AR system design and development. There are some limitations. First, the sample of models in our practice is limited. Our optimization work was focused on relatively small-scale museum objects. In addition, due to the local pandemic control, the number and demographics of participants in the user evaluation are also limited. In the future work, we aim to expand the range of the digital assets and conduct in-depth evaluations. We will carry out optimization work for large-scale heritage site reconstructions scanned by drone photogrammetry and investigate the system performances and user evaluations. We believe these efforts are of great significance in advancing the development of digital heritage and promoting interactive cultural experiences.

V. Conclusion

We present an optimization workflow to simplify model meshes and generate detailed texture maps, providing insights into the effective use of cultural heritage assets in interactive system development. Our proposed workflow demonstrates high efficiency in performance, high versatility and visibility in the editing process, ease of learning for novice users, and low cost in hardware, software, and labor. Photogrammetry together with the proposed optimization workflow allow a rapid access to photo-realistic 3D models that are compatible with various types of system development. This research provides archaeologists, architects, museum practitioners and game developers with an efficient method to achieve interactive visualizations of 3D scanned cultural heritage assets.

References

[1] S. Münster and M. Terras, “The visual side of digital humanities: a survey on topics, researchers, and epistemic cultures,” *Digit. Scholarsh. Humanit.*, vol. 35, pp. 366–389, 2020.

[2] M. Milosz, J. Kesik, and J. Montusiewicz, “3d scanning and visualization of large monuments of timurid architecture in central asia – a methodical approach,” *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 14, pp. 1 – 31, 2021.

[3] S. Münster, “Digital heritage as a scholarly field—topics, researchers, and perspectives from a bibliometric point of view,” *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 12, pp. 1 – 27, 2019.

[4] E. Ch’ng, S. Cai, T. E. Zhang, and F.-T. Leow, “Crowdsourcing 3d cultural heritage: best practice for mass photogrammetry,” *Journal of Cultural Heritage Management and Sustainable Development*, 2019.

[5] H. Rahaman, E. Champion, and M. Bekele, “From photo to 3D to mixed reality: A complete workflow for cultural heritage visualisation and experience,” *Digital Applications in Archaeology and Cultural Heritage*, vol. 13, jun 2019.

[6] D. A. Kolb, “Experiential Learning: Experience as the Source of Learning and Development,” *Journal of Organizational Behavior*, 1984.

[7] J. H. Falk and L. D. Dierking, *Learning from museums: Visitor experiences and the making of meaning*. 2000.

[8] F. Remondino and A. Rizzi, “Reality-based 3d documentation of natural and cultural heritage sites—techniques, problems, and examples,” *Applied Geomatics*, vol. 2, pp. 85–100, 2010.

[9] S. Thompson, “Cultural heritage and spectacle: painted and digital panoramic re-presentations of versailles,” 2016.

[10] Y. Li, E. Ch’ng, S. Cai, and S. See, “Multiuser Interaction with Hybrid VR and AR for Cultural Heritage Objects,” in *Digital Heritage 2018*, (San Francisco, USA), IEEE, 2018.

[11] W. Jakob, M. Tarini, D. Panozzo, and O. Sorkine-Hornung, “Instant field-aligned meshes,” *ACM Transactions on Graphics (TOG)*, vol. 34, pp. 1 – 15, 2015.

[12] F. Remondino, “Heritage recording and 3d modeling with photogrammetry and 3d scanning,” *Remote sensing*, vol. 3, no. 6, pp. 1104–1138, 2011.

[13] T.-K. Chen, H.-P. Fang, Y. Tian, H.-L. Fang, Y.-J. Li, S.-H. Tseng, and S.-E. Miao, “Design and evaluation of social interfaces for cultural exhibitions of chinese shadow puppetry,” *2011 IEEE 35th Annual Computer Software and Applications Conference*, pp. 346–347, 2011.