Accelerating 3D Scene Development for the Metaverse: Lessons from Photogrammetry and Manual Modeling

Viviana Pentangelo, Dario Di Dario, Vincenzo De Martino, Marco Dello Buono, Stefano Lambiase
Software Engineering (SeSa) Lab, Department of Computer Science, University of Salerno, Salerno, Italy
E-mail: {vpentangelo, ddidario, vdemartino, slambiase}@unisa.it

***Abstract*—The metaverse, a 3D immersive digital environment, is gaining significant interest due to its ability to connect people globally engaging and immersively, thanks to recent technological advancements. Developing high-quality 3D models is crucial for achieving realism and immersivity in the metaverse. However, this process is complex and resource-intensive, demanding specialized skills and substantial time. The emergence of novel automation tools and technologies, such as photogrammetry, which uses computer vision algorithms to reconstruct 3D models from 2D images, is beginning to address such challenges. Our research focused on analyzing the current state of such technologies in automating 3D scenes for the metaverse, comparing them to traditional manual modeling techniques. We conducted an experiment in which we built the same 3D scene using two techniques: a manual approach with Blender and a photogrammetry approach exploiting the Polycam tool on a mobile device. Our results have provided insights into the main strengths and limitations of using 3D automation techniques. The photogrammetry approach has significantly sped up the entire process, producing textures and models that accurately replicate real objects. However, it cannot wholly replace manual modeling approaches, without which it is impossible to obtain complete and efficient models. Lessons learned will serve as a foundation to guide developers in developing 3D scenes for the metaverse.**

***Index Terms*—Metaverse, Photogrammetry, 3D Scene Generation, Virtual Environments**

I. Introduction

In recent years, interest in the expansive domain of the metaverse has been steadily increasing, finding its relevance across diverse contexts and applications [19]. The metaverse is an immersive three-dimensional digital environment where users interact in real-time with both the surroundings and other users via their avatars, which serve as their digital proxies [33], [34]. This surge in interest has been fueled by rapid technological advancements, the growing digitization of various daily activities—from work to entertainment—and the significant impact of the COVID-19 pandemic, which underscored the necessity for digital solutions [19], [29]. Consequently, researchers have begun designing and developing metaverses for various purposes, including entertainment [23], software development [15], medical simulations [18], and educational applications [14].

The development of high-quality 3D models is crucial for the metaverse, as these models form the foundational elements of the virtual environments and objects that users interact with. The degree of immersivity—the extent to which users feel present within the digital world—relies heavily on the realism and intricacy of these 3D models, allowing a fluid social interaction with other avatars [24], [26]. Detailed and lifelike models contribute to a more engaging and believable experience, allowing users to interact with the virtual space in ways that mimic real-world interactions. However, developing such 3D models and the scenes they compose for integration into virtual worlds like the metaverse is a complex task [9], [10], [20]. One of the primary challenges is achieving the high levels of detail and realism needed to enhance immersivity [11], which demands substantial computational resources and time. Moreover, many professional tools (e.g., Blender and Autodesk) are difficult to learn and use due to their numerous functionalities [10], [20]. Additionally, creating these 3D models requires specialized skills that take time to acquire [10]. Despite the support of these tools, the considerable effort involved in developing 3D models and scenes often results in outcomes that lack realism and quality.

Given the previously mentioned difficulties, various solutions have been proposed to automate the process of creating 3D models and scenes. Specifically, research has focused on utilizing tools, such as Artificial Intelligence (AI), to generate scenes and models from textual descriptions. Many of these approaches involve using AI to identify and position pre-existing models in a scene based on a textual description [16], [21]. However, these methods do not create models from scratch; they find models similar to the users' requests. In response to this limitation, recent advancements have led to the development of techniques—like **photogrammetry**—based on object recognition and Computer Vision (CV) modeling through pictures of the actual object to model [4], [9]. These new approaches aim to enhance the automation of 3D model creation, potentially generating models that are more accurate and tailored to the specific needs. Nevertheless, the research in such a field is still in its early stage, and no clear methodology or process has yet been proposed or evaluated for the automatic creation and insertion of 3D scenes into the metaverse.

This work aims to contribute to the body of knowledge on the use of computer vision techinques, specifically photogrammetry, for 3D scene reconstruction for the metaverse. More in detail, the objective is to conduct a comparative analysis of two approaches to scene creation: manual and photogrammetry-based. The goal is to evaluate the effectiveness of the auto-

mated computer vision-based approach—leveraged using the Polycam tool [30]—in supporting the development of metaverse environments. In addition, this work seeks to provide useful lesson learned based on the results of our experiment to assist professionals in deciding whether and in which scenarios to rely on such automated techniques. All the material produced by our experiment is made available at our online appendix [27].

II. Background

The concept of the metaverse, introduced in Neal Stephenson's 1992 novel "Snow Crash" [34], has evolved into a tangible reality. Originating from Extended Reality (XR) and Virtual Reality (VR), the metaverse is now a three-dimensional virtual world where users interact through customizable avatars [33]. Key characteristics include persistent actions, realistic experiences, interoperability, and scalable architecture [12], [22]. It operates in real-time, allowing simultaneous interactions among thousands or millions of users. Creating realistic virtual environments tailored to various applications is crucial for enhancing the immersive user experience in the metaverse [12].

Traditional manual methods for the creation of 3D scenes for the metaverse require expertise in 3D modeling software [5] to manually recreate an object's mesh by manipulating its vertices, faces, and polygons [10]. Unlike such approaches, CV techniques aim to automate this process and eliminate the need for technical skills. Scientific literature has documented significant efforts in the field of computer vision aimed at reconstructing 3D scenes and virtual environments. Aharchi et al. [2] have outlined methods for creating 3D models from 2D images, categorizing them as *active* and *passive* techniques. Active methods involve using tools such as **laser scanners** to measure how light is reflected or changed when directed onto an object. In contrast, passive methods like **photogrammetry** that rely on images taken by regular cameras to analyze visual information like shadows, edges, and color changes to gather the 3D shapes of the object [4].

III. Related Work

This section discusses existing work that compared active and passive methods with manual measurements. Finally, we provide studies on the use of photogrammetry in the metaverse. Due to space constraints, we focus on the most recent and relevant studies across various domains.

Emmanuel P. Baltsavias [3] was among the first to compare 3D reconstruction techniques. This study highlighted the integral role of photogrammetry in the development of laser scanning technology, noting that many laser scanning service providers have backgrounds in photogrammetry. This suggests that laser scanning should be viewed as a complementary tool to photogrammetry rather than a competitor. Thoeni et al. [36] conducted a comparative study using multiple digital cameras and a terrestrial laser scanner (TLS) to reconstruct a rock wall. They aimed to determine the limitations of different camera types and establish the minimum camera requirements to achieve results comparable to those of TLS. Built et al. [7] compared terrestrial photogrammetry, UAV photogrammetry, and UAV video for rockfall monitoring. They found that UAVs are more cost-effective and provide similar resolution coverage to LiDAR or aerial photogrammetry, especially in smaller, rugged areas. They also overcome visibility, parking, and vegetation limitations by accessing difficult areas. Kadobayashi et al. [7] found that combining laser scanning and photogrammetry is most effective for digitally recording cultural heritage, with the best approach depending on the specific situation.

Beyond comparisons among 3D reconstruction techniques, other studies have compared these techniques with manual measurements. For instance, Anubhab et al. [25] demonstrated that photogrammetric methods for collecting anthropometric data are a reliable substitute for manual measurements across diverse populations. Randles et al. [32] assessed the relative accuracy of hands-on and photogrammetric measurement techniques used in vehicle accident reconstruction. They found both methods to be effective in measuring vehicle points. Finally, Düppe et al. [13] evaluated facial anthropometric measurements obtained through 3D photogrammetry and direct measurements using a caliper, finding no significant differences between the two methods.

Abramov et al. [1] employ drone photogrammetry to generate detailed 3D models of rural settlements, optimized for metaverse platforms like Voxels and Mona, promoting rural revival and a virtual economy. Prasetyadi et al. [31] use terrestrial photogrammetry instruments, e.g., Polycam, to create lifelike avatars and environments for the Science and Technology Region (KST) within the metaverse. Finally, Gledhill et al. [17] introduce a methodology to optimize photogrammetry data for metaverse virtual environments. Their focus is on enhancing mesh and textures to produce accurate, realistic representations of physical objects that perform well on low-powered VR devices.

The research community has invested considerable effort in identifying the best techniques for various domains and comparing the reliability of 3D techniques to manual measurements. However, no studies have analyzed photogrammetry from the perspective of efficiency and usability for metaverse applications, which require real-time rendering with realism to enhance user experience. To our knowledge, no analyses provide insights on applying these technologies to metaverse development, nor their advantages and limitations compared to traditional manual methods for real-time 3D scene modeling.

IV. Experiment Design

This research aims to conduct a comparative analysis of two distinct methodologies for constructing 3D environments intended for integration within the metaverse: a predominantly manual approach and a semi-automated technique supported by CV. Our ultimate goal is to develop comprehensive guidelines detailing (1) the optimal conditions for deploying each method and (2) the effective utilization of automated processes in 3D virtual environment creation.

To reach our objective, we executed an empirical study to compare the above-mentioned techniques by replicating a real-world indoor setting in three dimensions. The experiment required us to recreate the same indoor environment using two approaches: the first through manual modeling and texturing of each element and the second via computer vision technologies, notably leveraging the Polycam [30] software to transform 2D photographs into 3D models. The choice of Polycam was guided by its ability to demonstrate efficiency and user-friendliness as a photogrammetry application, capable of generating digital casts with a good balance between accuracy and processing speed compared to its competitors [8]. Moreover, Polycam is an open-access tool, downloadable by all users on their smartphones, which can therefore easily promote greater approachability in 3D scene reconstruction. Afterwards, we employed a suite of metrics specifically tailored for assessing 3D models to evaluate the results from both methodologies.

A. The Two Techniques

In this study, we focused on evaluating two distinct techniques for a common task: reconstructing a real-world scene within a 3D virtual environment.

- **Manual technique:** The first method, termed the "manual," involves a traditional approach where scene elements are crafted and textured manually utilizing Blender [5], a widely used open-source 3D modeling software [35], [37]. This choice was motivated by Blender's accessibility and substantial community support. Specifically, this process involved directly modeling all elements from the real scene within the software to create accurate 3D representations and associated materials for integration into the virtual environment.
- **Photogrammetry technique:** The second method, referred to as "Photogrammetry technique" leverages the Polycam [30] software to generate 3D models from photographs of the environment intended for reproduction. This approach begins with photographing the target environment, followed by using Polycam to create both the models and materials. These components were then imported into Blender, where they were assembled to reconstruct the scene.

B. Experiment Procedure

The experimental activity central to our study involved the digital reconstruction of a real-world scene within a 3D environment, specifically a living room owned by one of the study's authors (depicted in Figure 1, Picture A), using the two methodologies described earlier. The decision to replicate the living room was driven by two primary factors: (1) in line with Dionisio's emphasis on realism as a vital attribute of the metaverse [11], we aimed to recreate a commonplace yet intricate environment that is familiar in everyday human life; (2) having complete access and control over the physical space allowed us to adjust the complexity of the scene by modifying its elements as needed. The objective of this reconstruction was to produce a textured, functional 3D replica of the living room, where "functional" implies that the model is capable of being rendered and navigated in real time.

Prior to initiating the experimentation, a preliminary feasibility analysis was performed through a pilot study in a kitchen environment facilitated by one of the co-authors. The primary objective of this pilot study was to equip the research team with detailed knowledge of the tools and methodologies involved. Additionally, this initial exploration helped refine the experimental procedures for the main study, which are detailed in the subsequent section of this document.

In terms of procedure, concretely, the authors first organized the living room to contain some objects to make it seem as accurate as possible. Specifically, the set of objects consisted of: sofa, lamp, curtains, painting, coffee table with plant, radiator, air conditioner, fireplace with decorations, wall-mounted wardrobe with TV, potted plant, and remote control. As a second step, one of the co-authors (hereafter referred to as the *experimenter*) took pictures of the room; such were used as references for creating the living room using the two techniques and computing metrics (described in the continuation of the paper). After this, the experimenter used both techniques—starting from the manual one—to create the living room models and imported them into Blender. The entire process required multiple sessions of two hour each conducted on consecutive days.

The decision to assign the experimental tasks to a specific author was strategic rather than arbitrary. This author possesses considerable expertise in 3D modeling, acquired through formal education—completing both an introductory and an advanced course at a university—and personal projects. Engaging this author as a reference point enabled us to evaluate our methodology through a case study that closely mirrors a real-world scenario.

As a result of the above-mentioned procedure, two distinct 3D representations of the same living room scene were created within Blender. This approach allowed us to leverage Blender's capabilities to initially collect metrics related to model complexity and detail. Following the modeling phase, both scenes were exported to Unity3D, a robust game development platform, to assess their functionality. This step was crucial for evaluating how well each model performed in a dynamic environment, particularly focusing on real-time rendering capabilities and interactive potential. The functionality assessment in Unity3D helped determine the practicality of each modeling approach in creating realistic, immersive, and interactive 3D environments suitable for applications such as virtual reality experiences and simulations in the metaverse.

C. Metrics for Evaluation

In evaluating the two modeling techniques used in our study, we applied a set of metrics designed to capture both the time investment and quality of the resulting 3D scenes. The primary metric was the duration required to create each scene using the manual and photogrammetry techniques. This measurement provided insights into the actual time savings offered by each method, allowing us to weigh these against the quality of the

final 3D models produced. Moreover, in terms of quality, we used three additional metrics:

Number of Polygons — This metric quantifies the complexity of the 3D models. A higher number of polygons—i.e., the number of triangular faces composing the mesh—generally indicates a more detailed model, which could enhance realism but might also affect performance, especially in real-time applications [6]. This measure is relevant for the study because the number of polygons is closely related to the functionality of the scene and its operability in real-time—a crucial feature of the metaverse [11]. Indeed, each polygon requires computational time to determine how it should be rendered on screen, including texture rendering and lighting operations [6]. Therefore, a higher number implies a greater need for hardware and software resources to run the simulation smoothly with the created scene.

To compute this metrics, we compared the results obtained from the two techniques regarding *number of triangles*. It is important to specify that the experimenter did not receive any directives regarding the number of polygons to achieve during manual modeling and that the common goal of both techniques was to obtain a result as visually close to the real reference as possible.

Structural Similarity Index (SSIM) — This metric was first proposed by Wang et al. [38] and is based on the idea of measuring the quality of a 3D model by confronting it with the original object. We used the SSI to compare the visual similarity between the original photographs of the living room and the resulting 3D models. Specifically, the SSIM extracts three features from two images for comparison, i.e., luminance, contrast, and structure. This index helped us assess how accurately each technique managed to capture the real-world appearance and textures in the virtual environment. The metric ranges from -1 to 1, where -1 indicates maximum dissimilarity and 1 indicates maximum similarity. The SSIM was calculated as follow: by taking three photos of the real room from three different angles, ensuring that all objects were captured in at least one image, we generated three renders from the same angles for each scene. We attempted to precisely match the angle of each real photo in the renders of the 3D scenes as if we were taking the same photo inside the virtual room. Subsequently, for each render, we calculated the SSIM with the corresponding real image. We chose this methodology for calculating the metric on the entire room scene rather than individual models because we were interested in evaluating the realism of the room as a whole, rather than the realism of individual models.

GPU Usage — The GPU usage percentage for running the real-time simulation of the produced scenes. This measurement is useful for determining the functionality and operability of the results in actual metaverse contexts, i.e., 3D environments rendered in real-time that can be freely explored by the user [11], [33].

To measure this, we exported the two generated scenes to Unity3D, a graphics engine particularly used for metaverse applications [14], [28]. We set up the two scenes and positioned a player with a first-person view that could be freely controlled using a mouse and keyboard. We ran the simulation for 5 minutes on three different machines, each with a different model of NVIDIA GPU, with varying computing power: an NVIDIA GeForce RTX 3060, an NVIDIA GeForce MX130, and an NVIDIA GeForce GT 730. During the simulation, we measured the average GPU usage and average frames-per-second (FPS) in each session.

By employing the above-mentioned metrics, we aimed to comprehensively assess the balance between time efficiency and quality in the creation of 3D environments, facilitating a deeper understanding of the trade-offs involved in choosing between manual and photogrammetry modeling techniques.

D. Threats to Validity

One significant threat in this experiment stems from the dependency on a single participant's skill level and familiarity with the modeling tools used, namely Blender and Polycam. This reliance raises concerns about the reproducibility of the experiment since the results may be different from those of another individual with different levels of expertise. For example, if the participant is more proficient with Blender than Polycam, the results might be inherently biased in favor of manual modeling over photogrammetry generation. Furthermore, consistency in task execution is crucial. The same co-author conducted all modeling tasks under both techniques, which could introduce personal bias or variability in the application of each method. Such factors could skew the comparative analysis, favoring one method due to subjective preferences or familiarity rather than objective performance. Nevertheless, since the objective of this paper is to present and explore an automated method rather than undoubtedly assessing its superiority, we judged this choice as a good compromise in this part of the research.

The study is potentially compromised by its limited generalizability. The experimental design involves reconstructing a specific indoor environment—a living room—using two distinct methodologies. While informative, the results might not generalize to other types of environments, such as outdoor scenes or different architectural styles. This limitation restricts the applicability of the study's conclusions to similar settings and might not hold true for varied scenarios. Moreover, because the experiment utilizes particular software tools, the findings are somewhat tied to the capabilities and limitations of these tools at the time of the study. Technological advancements or different software choices could yield different outcomes, affecting the broader applicability of the results.

A potential limitation lies in whether the metrics used comprehensively capture the intended constructs. The study employs metrics such as the time taken, number of polygons, SSIM, and GPU usage to evaluate the quality and efficiency of the modeling techniques. However, these metrics may only partially encompass all relevant aspects of practical utility, such as the user experience in interactive applications or the computational performance in real-time rendering scenarios.

For instance, a model with fewer polygons might be less realistic, thus offering a poor user experience despite higher performance or lower computational demand. Therefore, relying solely on these metrics might not provide a complete picture of each method's effectiveness and practicality.

V. Analysis of the Results

The experiment's output consisted of two scenes resembling the indoor environment used as a reference, one for each technique. All the produced material is available in our online appendix [27]. Figure 1 shows a comparison of the models for the same object with the two techniques.

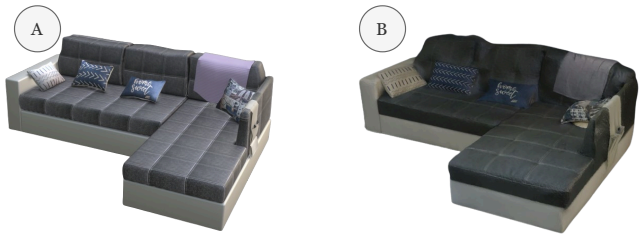

Fig. 1. Visual result of the two models resembling the sofa; A for the manual technique and B for the photogrammetry one.

Regarding the manual technique, each model was created manually in Blender, using photos of the room as a visual reference. Such photos were also used for the creation of the textures. The time for completing the scene, from accessing the room to take the photos to the completion of the 3D project, was **10 hours and 30 minutes**. This required five work sessions, each of two consecutive hours.

Regarding the photogrammetry technique, each model was generated by Polycam using its smartphone application. From 50 to 150 images were taken from all possible angles for each model. Subsequently, Blender was used only to compose the final scene. The time for completion, measured from scanning the room to finishing the 3D project, was **2 hours and 40 minutes**. This means that, in the context of our study, the photogrammetry technique reduced the time by nearly fourfold compared to the manual technique.

The first evaluation of the models was the polygon count for each, i.e., the number of triangular faces composing the mesh. This is pertinent as the polygon count directly influences the scene's functionality and real-time operability, which is a critical aspect of the metaverse [11].

The total number of polygons obtained from the manual technique was **56,113**, while the number obtained by the photogrammetry one was **470,902**. Each model's polygon number is shown in Table I. In this case, the difference between the two techniques is evident, as the photogrammetry technique has a total number of triangles that is more than eight times greater than that of the manual technique. However, more information can be obtained by visually examining the mesh topology, i.e., the arrangement of vertices on the surface. Figure 2 shows the two topologies for the sofa model. The differences between the two results lie primarily in the complexity and density of the polygons. The meshes resulting from manual modeling operations are simpler and cleaner but capture fewer details of the real objects' shape. The meshes derived from Polycam scanning, faithfully structured from images of the actual objects, captured much finer details, resulting in significantly higher polygonal complexity. Moreover, the meshes derived from the photogrammetry technique are not entirely complete, as the sides that could not be photographed are missing.

TABLE I
Objects' number of triangles for each technique.

Model	Triangles for Manual Tech.	Triangles for Photogrammetry Tech.
Sofa	2,433	37,339
Lamp	1,758	23,468
Coffee Table	2,162	62,304
Radiator	2,392	59,467
Painting	2,450	3,291
Air Cond.	96	34,211
Curtain	1,210	30,800
Potted Plant	14,780	10,000
Fireplace	4,160	142,237
Wardrobe	15,430	35,155
Remote control	9,242	32,630
Total	**56,113**	**470,902**

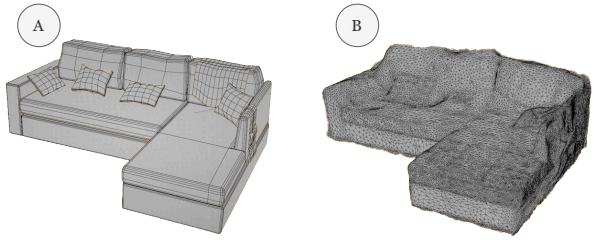

Fig. 2. Topology of the two models resembling the sofa; A for the manual technique and B for the photogrammetry one.

Following the preliminary analysis of the obtained models, we were interested in gaining more accurate insights regarding the realism and verisimilitude of the models compared to the actual reference scene. Therefore, we calculated the SSIM [38]. Figure 3 shows an example set of images from the same angle for the metric calculation.

The SSIM values obtained for each angle are summarized in Table II. Additionally, Figure 2 shows the heatmap generated for Angle 1 for both techniques. For all three angles, although slightly, the results indicate a greater similarity between the real images and the results from the photogrammetry technique. Despite the manual technique's results being visually cleaner and featuring complete object meshes with a lower polygon count, the SSIM results regarding similarity to the real environment are interesting. This suggests that despite the technical inaccuracies and scanning limitations of some elements of the room, the CV results contribute to a more

realistic outcome, as it can reproduce both models that closely match the real object's proportions and more realistic textures.

TABLE II
SSIM VALUES FOR EACH TECHNIQUE.

Picture's Angle	SSIM for Manual Tech.	SSIM for Photogrammetry Tech.
Angle 1	0.7009	0.72
Angle 2	0.696	0.7439
Angle 3	0.6037	0.6892

As the final step in evaluating the results of the two simulations, we were interested in assessing the functionality of the generated scenes in a real-time, freely navigable context. Therefore, we exported both scenes to Unity3D, a graphics engine particularly used for metaverse applications [14], [28].

Table III presents the measurement results for each of the GPUs. The results of the simulations did not show significant differences in GPU usage between the two techniques. In fact, the main difference was in the performance of one GPU compared to another, rather than significant variations between the scenes generated by the two techniques. The only notable variation was with the NVIDIA GeForce MX130, which reached its capacity limit only with the CV-generated scene, although it also showed high usage for the manual scene. However, this result is encouraging from the perspective of using auto-generation techniques for metaverse scenes: the latest GPU models can adequately handle even a significant difference in the number of polygons between two scenes, opening up possibilities for using CV-generated models, despite their complexity, in real-time rendered applications.

TABLE III
MEASUREMENT RESULTS OF UNITY3D SIMULATIONS.

GPU	Mean GPU usage for Manual Tech.	Mean GPU usage for Photogrammetry Tech.
RTX 3060	34.4%	38.6%
MX130	86%	99%
GT 730	99%	99%
GPU	**Mean FPS for Manual Tech.**	**Mean FPS for Photogrammetry Tech.**
RTX 3060	60	60
MX130	28	27
GT 730	24	25

Summary of the results

The photogrammetry technique recreated the scene in 2 hours and 40 minutes, compared to 10 hours and 30 minutes manually. However, the auto-generated models had eight times more triangles. SSIM showed slightly higher similarity for CV-generated renders with the real environment across all three angles. Real-time simulation in Unity3D showed no significant differences in hardware resource usage between the two scenes.

VI. DISCUSSION AND IMPLICATIONS

The experiment results offer insights into the current automation level in 3D scene generation for the metaverse using photogrammetry. This section discusses the strengths and weaknesses of each technique and shares lessons learned for quickly creating functional 3D scenes based on our findings.

A. Techniques' Comparison

As a first consideration, comparing the two techniques reveals that neither is superior in all aspects. Currently, two discussion points can be addressed: (1) the photogrammetry technique has significant advantages and can already be used today for creating realistic and functional metaverses; (2) this technique still has notable inaccuracies and technical limitations that require manual expertise to achieve the desired goal. To address this, we provide an overview of the strengths and weaknesses of each method.

Manual technique — Manually producing a 3D model using dedicated software such as Blender, refining its topology, and obtaining a complete and clean mesh is the best approach when designing 3D environments that need to be renderable in real-time and run on various devices with different hardware components and computing power. To date, the efficiency achieved by a model created and refined manually by an expert has yet to be surpassed by 3D model auto-generation techniques. However, choosing solely this approach when developing a vast, detailed, and realistic environment like the metaverse increases production costs and time, since the manual technique for a single scene took significantly longer compared to auto-generation with Polycam. When extending these results to larger and more complex environments, such timeframes can become unsustainable, highlighting the current need to integrate automation techniques to make the expansive and comprehensive concept of the metaverse a reality.

Photogrammetry technique — Using a 3D model auto-generation tool based on computer vision algorithms like Polycam allowed us to achieve tangible results in significantly less time. Although this characteristic represents the most obvious advantage, it is not the only one observed: the similarity to the real environment was also more faithful, as indicated by the SSIM calculation. An important strength of this technique, when aiming for realistic and credible scenes, is that it maps real objects on a 1:1 scale, effectively scanning them. The level of detail it captures is greater and more refined than a manually generated model based solely on 2D photo references. Another significant strength is the auto-generation of textures, which are also faithfully extracted and created from the real object. However, using this technique alone to fully develop a real-time renderable 3D environment is not yet feasible. The first evident limitation is the excessive polygonal complexity of the resulting models. Although this did not pose a real problem during the Unity simulation, this excessive numerical difference in polygons compared to manually obtained models can be a limitation

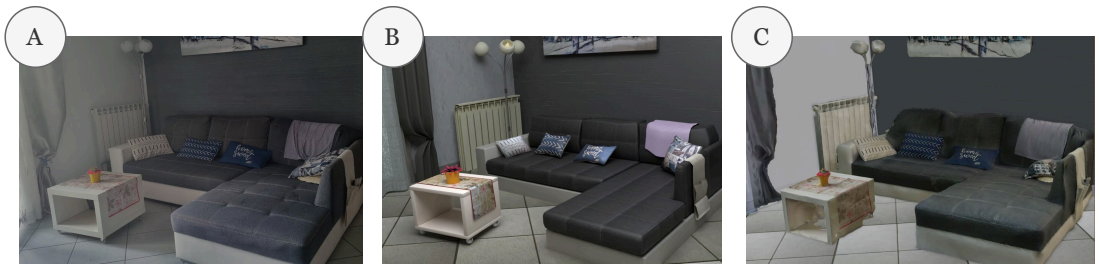

Fig. 3. Three images from the same angle for the SSIM calculation. A shows the image taken in the real scene, B the render of the scene created with the manual technique, and C the render of the scene created with the photogrammetry technique.

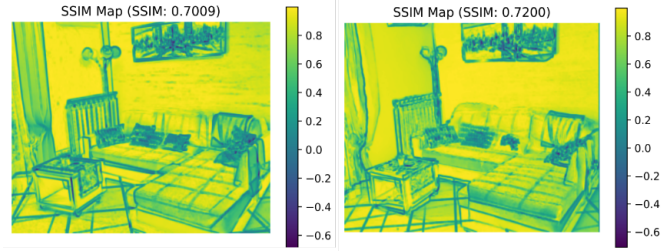

Fig. 4. Heatmaps resulting from the SSIM calculation for Angle 1. On the left, the overlay with the render from the manual technique; on the right, the one with the render from the photogrammetry technique.

when scaling up to larger and more complex environments. A second major limitation is that the models are incomplete and have significant inaccuracies and distortions in many areas. While this technique produced visually excellent results for small and detailed objects that could be easily photographed from many distinct points, it was weak and imprecise for large objects with simpler geometric shapes, where it was impossible to obtain a complete scan. The resulting object appears incomplete and highly deformed for some objects, such as the wall-mounted cabinet, the air conditioner, or the lamp. Such considerations suggest that the technique can easily support the creation of 3D scenes for the metaverse, but we are still far from being able to rely entirely on photogrammetry technologies.

B. Lessons Learned for Development of 3D Metaverse Scenes

Following the comparative analysis of the two techniques, we state our lessons learned and suggestions on how to combine them to obtain the best of both worlds specifically within the context of Metaverse scene development.

- First, *the use of a photogrammetry tool can be particularly useful for initial scanning of the overall structure and proportions of objects to be refined later*. This can solve the problem of obtaining accurate proportions of the actual object and overcome the limitations of having only 2D references when using a manual technique. **In the context of the Metaverse, this allows for rapid creation of large-scale, immersive environments that mirror real-world proportions accurately, enhancing user immersion and interaction.**
- Second, *manual modeling techniques are more efficient for objects with simple geometry*. For objects with very simple geometry that can easily be reduced to primitive 3D shapes, such as rectangular cabinets, various cylindrical objects, or, in general, objects with very simple lines, there is hardly any reason to rely on their 3D scanning. **This efficiency is crucial in the Metaverse where performance optimization is necessary to maintain real-time rendering and interaction speeds, providing a smoother user experience.**
- Third, *Photogrammetry techniques are particularly useful for generating models of small objects with a high level of detail*. Complementing the previous lesson learned, small objects with a high level of detail—such as the potted plant and the decorative items on the mantelpiece in the generated scene—are particularly suitable for being scanned and generated by photogrammetry algorithms rather than being completely developed manually. **In the Metaverse, these detailed objects contribute significantly to the realism and aesthetic appeal of the virtual environment, which is essential for user engagement and satisfaction.**
- Last, *integrating models generated by both techniques can be a valid way to balance level of detail and performance*. To obtain real-time 3D scenes that adequately balance performance with the level of detail and realism achieved, a good strategy is to combine models and outcomes from both techniques. **In the Metaverse, this balance is critical as it ensures that scenes are both visually compelling and performant, avoiding latency issues and enhancing the overall user experience.**

In summary, the photogrammetry offers significant advantages for constructing 3D environments. While it can not yet fully replace human expertise, our insights can help make creating 3D scenes for the metaverse more efficient.

VII. Conclusion and Future Work

This paper compares manual and CV-assisted techniques for creating 3D environments in the metaverse. Results indicate that manual modeling in Blender is effective for low-polygon, real-time rendering but is time-consuming, while CV-assisted tools like Polycam generate detailed models quickly, albeit with higher polygon counts and resource demands. The study

concludes that these methods complement each other, with CV techniques providing rapid model acquisition and manual methods refining models for performance optimization. A combined approach can enhance quality, reduce resource use and costs, and improve software sustainability, though CV methods cannot fully replace manual techniques.

The results obtained from this study can be extended from various perspectives for future works. First, our findings were derived from a single type of environment; in this regard, repeating the experiment in diverse settings could enhance the generalizability of the results. Additionally, deeper insights could be provided by including individuals with varying levels of expertise. Finally, further exploration could involve applying different software and devices for both techniques, broadening the applicability and robustness of the results. All the above-mentioned points are part of our future agenda.

REFERENCES

[1] N. Abramov, H. Lankegowda, S. Liu, L. Barazzetti, C. Beltracchi, and P. Ruttico. Metamorphosis: A digital approach to transforming communities through photogrammetry and metaverse. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48:1–8, 2024.

[2] M. Aharchi and M. Ait Kbir. A review on 3d reconstruction techniques from 2d images. In *Innovations in Smart Cities Applications Edition 3: The Proceedings of the 4th International Conference on Smart City Applications 4*, pages 510–522. Springer, 2020.

[3] E. P. Baltsavias. A comparison between photogrammetry and laser scanning. *ISPRS Journal of photogrammetry and Remote Sensing*, 54(2-3):83–94, 1999.

[4] J. Baqersad, P. Poozesh, C. Niezrecki, and P. Avitabile. Photogrammetry and optical methods in structural dynamics–a review. *Mechanical Systems and Signal Processing*, 86:17–34, 2017.

[5] Blender HQ Amsterdam. Blender. https://www.blender.org [Accessed: May 2024].

[6] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy. *Polygon mesh processing*. CRC press, 2010.

[7] F. Buill, M. A. Núñez-Andrés, N. Lantada, and A. Prades. Comparison of photogrammetric techniques for rockfalls monitoring. In *IOP Conference Series: Earth and Environmental Science*, volume 44, page 042023. IOP Publishing, 2016.

[8] M. M. Buzayan, A. H. Elkezza, S. F. Ahmad, N. M. Salleh, and I. Sivakumar. A comparative evaluation of photogrammetry software programs and conventional impression techniques for the fabrication of nasal maxillofacial prostheses. *The Journal of Prosthetic Dentistry*, 2023.

[9] A. Cannavò, A. D'Alessandro, D. Maglione, G. Marullo, C. Zhang, F. Lamberti, et al. Automatic generation of affective 3d virtual environments from 2d images. In *Proc. 15th International Conference on Computer Graphics Theory and Applications (GRAPP 2020)*, pages 113–124. SCITEPRESS, 2020.

[10] A. Cannavò, C. Demartini, L. Morra, and F. Lamberti. Immersive virtual reality-based interfaces for character animation. *IEEE Access*, 7:125463–125480, 2019.

[11] J. D. N. Dionisio, W. G. B. Iii, and R. Gilbert. 3d virtual worlds and the metaverse: Current status and future possibilities. *ACM Computing Surveys (CSUR)*, 45(3):1–38, 2013.

[12] J. D. N. Dionisio, W. G. B. Iii, and R. Gilbert. 3d virtual worlds and the metaverse: Current status and future possibilities. *ACM Computing Surveys (CSUR)*, 45(3):1–38, 2013.

[13] K. Düppe, M. Becker, and B. Schönmeyr. Evaluation of facial anthropometry using three-dimensional photogrammetry and direct measuring techniques. *Journal of Craniofacial Surgery*, 29(5):1245–1251, 2018.

[14] D. et al. Metaverse for social good: A university campus prototype. In *Proceedings of the 29th ACM international conference on multimedia*, pages 153–161, 2021.

[15] F. Fernandes and C. Werner. A systematic literature review of the metaverse for software engineering education: Overview, challenges and opportunities. *PRESENCE: Washington, WA, USA*, 2022.

[16] M. A. Ghorab and A. Lakhfif. Text to 3d,2d scene generation systems, frameworks and approaches: a survey. *2022 4th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, pages 1–6, 2022.

[17] D. Gledhill and M. Novak. A novel methodology for the optimization of photogrammetry data of physical objects for use in metaverse virtual environments. In *2023 IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE)*, pages 40–45. IEEE, 2023.

[18] H. Koo. Training in lung cancer surgery through the metaverse, including extended reality, in the smart operating room of seoul national university bundang hospital, korea. *Journal of educational evaluation for health professions*, 18, 2021.

[19] L.-H. Lee, T. Braud, P. Zhou, L. Wang, D. Xu, Z. Lin, A. Kumar, C. Bermejo, and P. Hui. All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda. *arXiv preprint arXiv:2110.05352*, 2021.

[20] J. Lu, C. Li, C. Yin, and L. Ma. A new framework for automatic 3d scene construction from text description. In *2010 IEEE International Conference on Progress in Informatics and Computing*, volume 2, pages 964–968, 2010.

[21] R. Ma, A. G. Patil, M. Fisher, M. Li, S. Pirk, B.-S. Hua, S.-K. Yeung, X. Tong, L. J. Guibas, and H. Zhang. Language-driven synthesis of 3d scenes from scene databases. *ACM Transactions on Graphics (TOG)*, 37:1 – 16, 2018.

[22] S. Mystakidis. Metaverse. *Encyclopedia*, 2(1):486–497, 2022.

[23] X. Niu and W. Feng. Immersive entertainment environments-from theme parks to metaverse. In *International Conference on Human-Computer Interaction*, pages 392–403. Springer, 2022.

[24] K. L. Nowak and J. Fox. Avatars and computer-mediated communication: a review of the definitions, uses, and effects of digital representations. *Review of Communication Research*, 6:30–53, 2018.

[25] A. Pal, T. Patel, and K. Khro. A comparative study of the effectiveness of photogrammetric versus manual anthropometric measurements. *Work*, (Preprint):1–12.

[26] R. M. Paweroi and M. Köppen. 3d avatar animation optimization in metaverse by differential evolution algorithm. In *2023 International Conference on Intelligent Metaverse Technologies & Applications (iMETA)*, pages 1–7. IEEE, 2023.

[27] V. Pentangelo, D. Di Dario, V. De Martino, M. Dello Buono, and S. Lambiase. Accelerating 3d scene development for the metaverse: Lessons from photogrammetry and manual modeling. https://doi.org/10.6084/m9.figshare.25914097.

[28] V. Pentangelo, D. Di Dario, S. Lambiase, F. Ferrucci, C. Gravino, and F. Palomba. Senem: A software engineering-enabled educational metaverse. *Information and Software Technology*, 174:107512, 2024.

[29] S. Pokhrel and R. Chhetri. A literature review on impact of covid-19 pandemic on teaching and learning. *Higher education for the future*, 8(1):133–141, 2021.

[30] Polycam. Polycam. https://poly.cam [Accessed: May 2024].

[31] A. Prasetyadi, M. Y. Rezaldi, C. Trianggoro, A. Febriandirza, R. Suhud, A. Gunawan, and H. M. Ramdhan. Digital avatar sub-metaverse modeling using terrestrial photogrammetry techniques. In *2023 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, pages 221–225. IEEE, 2023.

[32] B. Randles, B. Jones, J. Welcher, T. Szabo, D. Elliott, and C. MacAdams. The accuracy of photogrammetry vs. hands-on measurement techniques used in accident reconstruction. Technical report, SAE Technical Paper, 2010.

[33] G. D. Ritterbusch and M. R. Teichmann. Defining the metaverse: A systematic literature review. *IEEE Access*, 2023.

[34] N. Stephenson. *Snow Crash*. Bantam Books, New York, NY, USA, 1992.

[35] The Pixelary. Behind the pixelary, 2018. https://blog.thepixelary.com/post/174286685782/blender-for-computer-vision-machine-learning [Accessed: May 2024].

[36] K. Thoeni, A. Giacomini, R. Murtagh, and E. Kniest. A comparison of multi-view 3d reconstruction of a rock wall using several cameras and a laser scanner. *The international archives of the photogrammetry, remote sensing and spatial information sciences*, 40:573–580, 2014.

[37] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid. Learning from synthetic humans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 109–117, 2017.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.