mathematics

MDPI

Article

Joint Optimization-Based Texture and Geometry Enhancement Method for Single-Image-Based 3D Content Creation

Jisun Park [1], Moonhyeon Kim [2], Jaesung Kim [2], Wongyeom Kim [2] and Kyungeun Cho [3,*]

[1] NUI/NUX Platform Research Center, Dongguk University-Seoul, 30 Pildongro-1-gil, Jung-gu, Seoul 04620, Republic of Korea; jisun@dongguk.edu
[2] Department of Computer Science and Artificial Intelligence, Dongguk University-Seoul, 30 Pildongro 1-gil, Jung-gu, Seoul 04620, Republic of Korea; bir0301@dongguk.edu (M.K.); kjs9504@dongguk.edu (J.K.); dnjsrua59@dgu.ac.kr (W.K.)
[3] Division of AI Software Convergence, Dongguk University-Seoul, 30 Pildongro 1-gil, Jung-gu, Seoul 04620, Republic of Korea
* Correspondence: cke@dongguk.edu; Tel.: +82-2-2260-3834

Abstract: Recent studies have explored the generation of three-dimensional (3D) meshes from single images. A key challenge in this area is the difficulty of improving both the generalization and detail simultaneously in 3D mesh generation. To address this issue, existing methods utilize fixed-resolution mesh features to train networks for generalization. This approach is capable of generating the overall 3D shape without limitations on object categories. However, the generated shape often exhibits a blurred surface and suffers from suboptimal texture resolution due to the fixed-resolution mesh features. In this study, we propose a joint optimization method that enhances geometry and texture by integrating generalized 3D mesh generation with adjustable mesh resolution. Specifically, we apply an inverse-rendering-based remeshing technique that enables the estimation of complex-shaped mesh estimations without relying on fixed-resolution structures. After remeshing, we enhance the texture to improve the detailed quality of the remeshed mesh via a texture enhancement diffusion model. By separating the tasks of generalization, detailed geometry estimation, and texture enhancement and adapting different target features for each specific network, the proposed joint optimization method effectively addresses the characteristics of individual objects, resulting in increased surface detail and the generation of high-quality textures. Experimental results on the Google Scanned Objects and ShapeNet datasets demonstrate that the proposed method significantly improves the accuracy of 3D geometry and texture estimation, as evaluated by the PSNR, SSIM, LPIPS, and CD metrics.

Keywords: 3D content creation; image-based 3D generation; joint optimization; texture enhancement; geometry enhancement

MSC: 68T07; 68U10; 68T45

check for updates

Citation: Park, J.; Kim, M.; Kim, J.; Kim, W.; Cho, K. Joint Optimization-Based Texture and Geometry Enhancement Method for Single-Image-Based 3D Content Creation. *Mathematics* **2024**, *12*, 3369. https://doi.org/10.3390/math12213369

Academic Editor: Junlin Hu

Received: 19 September 2024
Revised: 17 October 2024
Accepted: 25 October 2024
Published: 28 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Recently, the generation of three-dimensional (3D) meshes from single images has become a prominent topic in computer vision and graphics. High-quality 3D mesh models are central to 3D vision and graphics applications, and 3D editing, rendering, and simulation tools have been specifically optimized for this purpose. Traditionally, 3D mesh assets have been manually created by expert artists or generated using photogrammetry systems [1]. However, recent approaches—such as neural-network-based methods, for example, neural radiance fields (NeRFs) [2]—offer simpler end-to-end pipelines through 3D optimization. Nevertheless, NeRF-based methods require additional postprocessing to convert the predicted 3D rendered images into meshes [3,4]. To address this issue, research on generating 3D meshes from single images is ongoing [5–9]. However, this field is challenging because of the need to represent complex 3D shapes as meshes. 3D data often

include discontinuities, such as irregular triangular surface structures, which complicate shape generalization and learning. Traditional methods frequently employ fixed-resolution meshes during training, leading to significant limitations in accurately capturing complex shapes. These approaches often involve trade-offs between detail and computational efficiency, which affect the overall quality of the generated 3D models. To address these issues, many single-image-based 3D reconstruction methods have focused on generalizing 3D shape estimation. However, they often rely on rigid 3D mesh network structures, which have clear limitations in capturing the level of detail necessary for high-quality 3D models.

To overcome these challenges, we propose a joint optimization method to enhance both the geometry and texture of 3D meshes derived from the generalization of 3D shape estimations from a single image. The proposed method integrates generalized 3D mesh generation with the capability of dynamically adjusting mesh resolution. This integration enables a more flexible and precise representation of complex shapes.

The proposed method is divided into three main parts: (1) coarse 3D mesh reconstruction, (2) geometry enhancement, and (3) texture enhancement. In the first part, a coarse 3D mesh is generated on the basis of a single image via a pretrained large reconstruction model (LRM). Although this mesh is not detailed, it maintains the overall shape of the object and addresses the problem of shape distortion that can occur when a single image is predicted. Building on this coarse 3D mesh, we perform joint optimization-based geometry and texture enhancement through inverse rendering. For geometric enhancement, we extract sparse mesh vertices and normal map images from the coarse 3D mesh and perform continuous remeshing via inverse rendering to generate a dense vertex mesh. Next, texture enhancement is applied to the dense vertex mesh to obtain high-quality texture details, ensuring a comprehensive improvement in both the geometry and texture. To improve the geometric quality of the coarse 3D texture mesh, we use a diffusion model to enhance the normal maps.

The main contributions of this study are as follows:

1. We propose a joint optimization-based approach for texture and geometry enhancement using an LRM model to maintain the overall 3D geometry while achieving a high-quality textured mesh.
2. We integrate it with an inverse-rendering-based continuous remeshing technique that enables continuous and complex-shaped mesh estimation without relying on fixed-resolution meshes.
3. We propose combining it with inverse texture mapping and applying rendering image enhancement via a diffusion model.

The remainder of this paper is organized as follows: Section 2 reviews related work in the fields of 3D diffusion, large reconstruction models, and inverse rendering. Section 3 details the proposed joint optimization-based texture and geometry enhancement method, including its key components and innovations. In Section 4, the experimental results are analyzed to demonstrate the effectiveness of the proposed method. Section 5 concludes the paper, discusses the proposed method in the context of experimental findings, and elaborates on its limitations.

2. Related Work

2.1. 2D Images to 3D Diffusion Model

Subsequent to the advent of 2D diffusion models, such as stable diffusion [10] and DALL-E [11], it has become possible to utilize these models to train NeRFs [2] and develop 3D generation models. Reconstructing 3D scenes and objects from a single image is a critical task for 3D vision. Various methods have been explored to achieve improved 3D representations. Explicit representation techniques include point cloud data [12–14], 3D voxel grids [15,16], multiplane images [17,18], meshes [19,20], and 3D Gaussians [21]. The implicit representation methods include signed distance fields (SDF) [22,23] and neural fields [24–29], which have been studied to increase the reconstruction accuracy.

DreamFusion [30] and Magic3D [31] have demonstrated that 2D generative models can be adapted for 3D generation. These studies introduced score distillation sampling to generate 3D shapes from text by using 2D diffusion models to supervise and optimize the 3D representations. Although these optimization-based methods can generate 3D object scenes from single-image inputs, they encounter issues related to 2D diffusion models, which may produce redundant and inconsistent content. For example, objects may be nearly invisible or obscured at certain angles. This finding indicates that 3D objects can differ from different viewpoints or under different lighting conditions, revealing the limitations of 2D lifting methods when creating 3D models from a single viewpoint because of potential discrepancies and ambiguities between scenes.

To address these challenges, models such as MVdream [32] and Zero123 [33] have been proposed for generalized 3D content creation. MVdream [32] enhances the generalization capabilities of 2D diffusion models and the consistency of 3D rendering by learning from both 2D and 3D data. The Zero123 [33] model introduces a novel approach that uses relative camera poses as conditions to generate 3D shapes from 2D images, enabling consistent shape inference from various viewpoints to estimate stable 3D models. Models such as viewset diffusion [34] and SyncDreamer [35] utilize attention layers to generate consistent multiangle color images. Conversely, Zero123 [33] creates a model that generates images from different viewpoints within a 3D scene using a single image and camera matrix (R, T). Whereas previous diffusion models relied solely on single images to create 3D synthetic views, this approach fine-tuned the diffusion model on the basis of viewpoint conditions, significantly enhancing its ability to predict high-resolution textures and geometric details. Zero123++ [36], an advanced version of Zero123 [33], subsequently demonstrated improved performance by tiling six images in a 3×2 layout, fixing the azimuth and elevation, and training the diffusion model on these camera poses.

Additionally, some studies have been conducted on 3D generation in the form of 3D point clouds by combining techniques such as diffusion models, neural rendering, and Gaussian splatting [37]. Although these studies are suitable for generating large-scale scenes, they tend to lack detail because of their point cloud nature, making them unsuitable for creating high-quality 3D content.

Even though these methods integrate 2D diffusion models and 3D data and can generate 3D scenes from single images, their primary focus remains on rendering image estimation or 3D point clouds. Consequently, they often lack the ability to fully represent geometric information, and the quality of 3D mesh results may be compromised because of incomplete texture information (e.g., shape, depth, and texture) during postprocessing.

2.2. Large 3D Reconstruction Model

LRMs [8,9] have been designed to predict 3D models of objects quickly and accurately from a single input image. These models use a transformer-based encoder–decoder architecture to effectively learn 3D representations of objects from a single image. The image features generated by the encoder are transformed into vectors with spatial and positional information via the transformer–decoder structure. These vectors are then converted into a triplane feature map by upsampling. A multilayer perceptron (MLP) decodes this feature map to predict color and density, enabling volume rendering from arbitrary viewpoints. LRMs require only multiview supervision, rendering them applicable to training with 3D, video, and image datasets. These models analyze global and local image characteristics to transform them into 3D shapes, demonstrating a high generalization capability for multiview outputs and generating detailed 3D outputs from limited data, thus improving the 3D generation quality.

Several studies have been conducted on LRMs to create generalized models for reconstructing 3D models from images. Instant3D [8] fine-tunes a 2D text-to-image diffusion model to generate consistent four-view images and then uses the LRM model to lift these images into 3D. However, owing to the use of relatively low-resolution triplanes in the feed-forward reconstructor, the generated textures tend to be blurry compared with the

input images. CRM [6] generates six orthographic images from a single input image. A different diffusion model was used to create a canonical coordinate map (CCM). The six images and CCMs were sent to the CRM, which reconstructed the textured meshes. This approach effectively leverages the spatial relationship between the input image and output triplane, reducing training costs compared with those of previous transformer-based methods. However, as noted in a subsequent study [38], performance degrades if the input image has a large elevation or different fields of view. LGM [39] and GRM [40] have been proposed as models that perform LRM-based 3D generation via Gaussian models. They replace the traditional triplane NeRFs with 3D Gaussian representations. Although this method improves rendering efficiency in 3D reconstruction, it faces limitations in accurately extracting high-precision geometric details and surface textures. InstantMesh [7] combines the strengths of off-the-shelf multiview diffusion and LRM-based sparse-view reconstruction models to generate various 3D assets within 10 s. However, the transformer-based triplane decoder in the LRM creates 64 $\times$ 64 triplanes, which can cause bottlenecks in high-quality 3D modeling. Although using Flexicube can reduce mesh surface artifacts and improve smoothness, its performance decreases when modeling small and thin structures. TripoSR [5] significantly enhances data processing, model design, and training techniques on the basis of the LRM network architecture. It improves technology using a transformer-based architecture and multiple axes, demonstrating high computational efficiency and fast reconstruction capabilities. It performs well with datasets similar to the Objaverse dataset; however, it shows a lack of imagination ability, reduced geometric quality, and texture on the back side when handling freestyle input images.

Current single-image-based 3D reconstruction methods focus on generalizing 3D shape estimation and are often based on fixed 3D mesh network structures. This study proposes a method for enhancing geometry and texture through joint optimization combined with generalized 3D mesh generation.

2.3. Inverse Rendering

Inverse rendering begins with the rendered image, compares it to some form of the ground-truth image, and updates the parameters of the scene elements with the loss that compares the images. Inverse rendering-based 3D reconstruction methods have been conducted for 3D object and scene generation.

The objective of inverse rendering is to decompose an image into its geometric structures, material properties, and lighting conditions. Considering the inherent ambiguity between observed images and their fundamental attributes, various methods with different constraints have been proposed. For example, some methods capture images using fixed lighting while rotating objects [41], whereas others use moving cameras with simultaneous lighting [42]. Inverse rendering models combined with neural representations simulate the interaction of light with neural volumes and diverse material properties to model the scene and estimate lighting and material parameters during the optimization process. Neural reflectance fields [43] represent a scene as a field of volumetric density, surface normals, and bidirectional reflectance distribution functions (BRDFs) with known point light. PhySG [44] assumes the visibility of all light sources without shadow simulation and represents lighting and scene BRDFs using spherical Gaussians for acceleration. NeRV [45] and InvRender [46] extend this approach to arbitrarily known lighting conditions and train additional MLPs to model the lighting visibility. TensoIR [47] adopts an efficient TensoRF [48] representation to compute the visibility and indirect lighting through ray tracing, although it is limited to the object level. To model surface geometry via point clouds, fuzzy metaballs (FMs) [49,50] offer an excellent method for rendering depth via 3D Gaussians, along with order-independent transparency and approximate intersections. GS-IR [51] leverages 3D Gaussian splatting (3DGS) to achieve novel photorealistic view synthesis and relighting results.

However, these methods require multiple images for 3D generation. To address this issue, recent approaches focus on large 3D reconstruction models that use only a single

image. However, current large 3D reconstruction models have limitations that result in 3D models with reduced surface and texture details, which restricts their ability to create high-quality 3D content. Therefore, we propose a joint optimization method with 3D LRM for texture and geometry enhancement. The overall 3D shape based on a single image is generated by integrating a 3D LRM model, and the surface and texture details of the generated 3D model can be enhanced through the proposed joint optimization method, which includes continuous remeshing and a diffusion model.

3. Joint-Optimization-Based Texture and Geometry Enhancement Methods

3.1. Overview of the Proposed Joint Optimization Method

We propose a joint optimization method that enhances geometry and texture by integrating generalized 3D mesh generation with adjustable mesh resolution. Figure 1 shows an overview of the proposed method, which comprises three main parts: coarse 3D mesh reconstruction, geometric enhancement, and texture enhancement. In response to the input of a single image, a coarsely textured 3D mesh is generated, followed by enhancements to both the shape and texture based on the generated mesh. Each stage takes approximately 10 s of processing time. Initially, a coarse 3D mesh is generated from a single image using InstantMesh [7], which is a pretrained LRM that retains the overall shape and addresses shape distortion despite the lack of fine details. In this process, multiview RGB and normal maps are generated via multiview diffusion methods such as zero123 [33]. These are used not only to create a coarse 3D mesh but also to aid in geometry and texture enhancement, which reduced distortions in the 3D shape during joint optimization. The generated coarse mesh is then refined through a joint-optimization-based geometry and texture enhancement via inverse rendering techniques. For geometry enhancement, sparse mesh vertices and normal map images are extracted from the coarse mesh, and continuous remeshing is performed to generate a dense vertex mesh. To further improve the geometric quality, a diffusion model is employed to enhance the normal maps. Finally, texture enhancement is applied to the dense vertex mesh to achieve high-quality texture details, ensuring comprehensive improvements in both the geometry and texture. In the following subsections, we explain the joint optimization-based geometry and texture enhancement in detail.

3.2. Geometry Enhancement for Joint Optimization

To further enhance the coarse mesh with rich details, instead of directly manipulating the mesh vertices, we propose improving and editing the initial mesh using normal maps as an intermediate representation (Figure 2). First, a coarse mesh is rendered into normal maps. The normal-based diffusion model is subsequently leveraged to enhance the rendered normals with intricate details. The refined normals subsequently serve as supervision to optimize the mesh, which yields a refined mesh with rich details.

3.2.1. Normal Map Enhancement

We propose using normal maps as an intermediate representation instead of directly manipulating the mesh vertices to enrich the coarse mesh with finer details. The proposed approach first renders the coarse mesh into normal maps and subsequently applies a normal-based diffusion model to enhance these maps with intricate details. The refined normal maps are then used for supervision to optimize and improve the mesh, resulting in a highly detailed final mesh. We employ ControlNet [52] to enrich the details of the rendered normal maps. We propose a more efficient approach that leverages preexisting priors from a 2D diffusion model trained in the depth domain. The proposed method involves finetuning a diffusion model [53] on normal images, after which a pretrained ControlNet-Tile network is applied. We fine-tune the stable diffusion model trained on depth data, training it with the text “normal map” and normal map training images. This

is denoted as φ from the depth domain to refine the outputs. Given a rendered normal map (n_i) for the i-th view, the result from the network is represented as

$$\hat{n_i} = \varphi(n_i, y_{texture}) \tag{1}$$

where $y_{texture}$ is the input text condition that can be omitted using the guess mode suggested by ControlNet [52]. Binary-based alpha images are generated from the enhanced multi-view normal maps, marking object regions as 255 and the background as 0 for each viewpoint. The alpha images are essential for calculating the loss only within object-containing regions during normal map-based continuous remeshing. The refined multi view normal map images serve as the ground truth for each viewpoint in continuous remeshing. When vertex and face values of the 3D mesh are provided through inverse rendering, the loss is computed between the rendered normal map images and the ground truth normal map images.

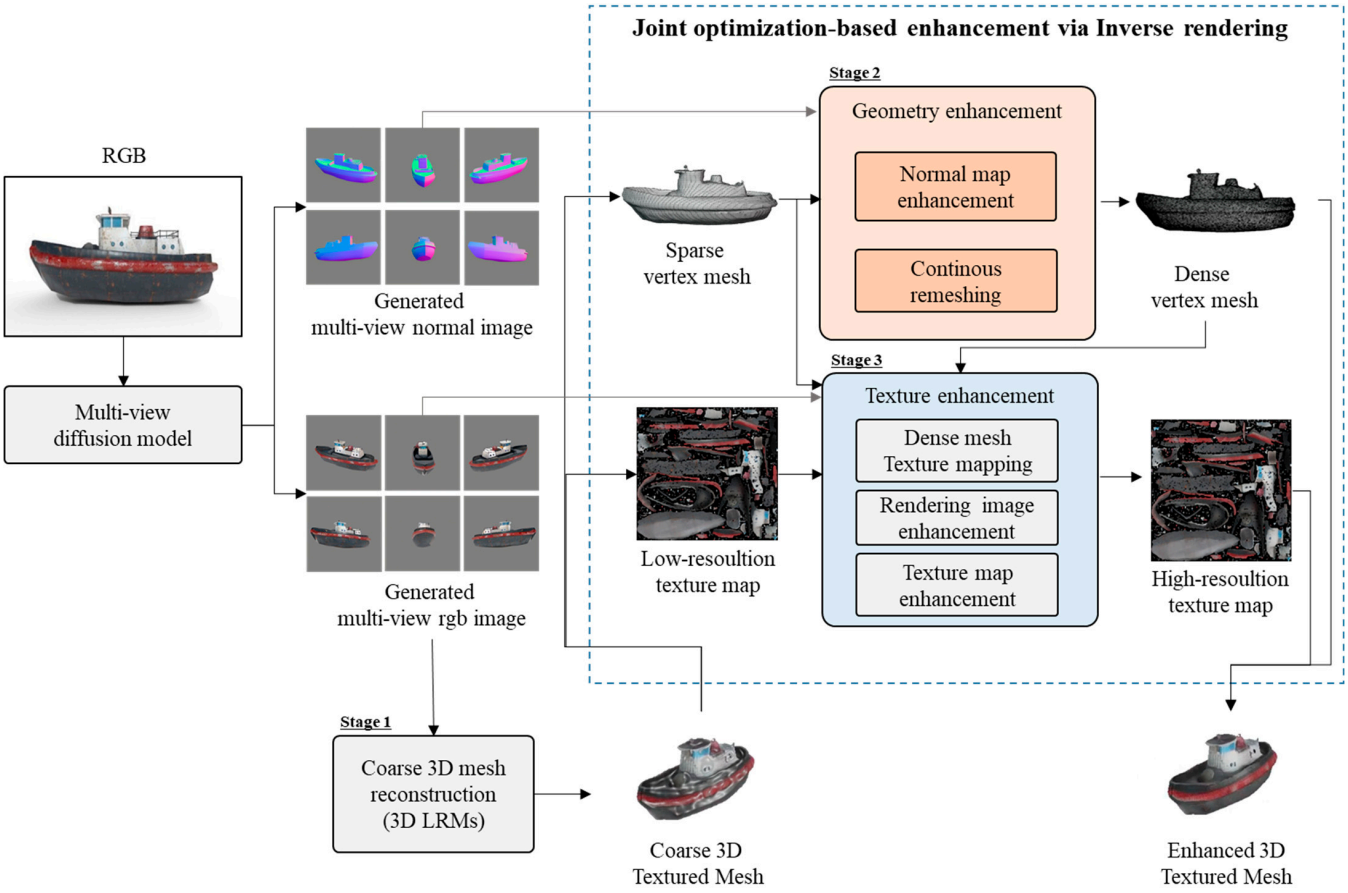


Figure 1. Overview of the proposed joint optimization-based texture and geometry enhancement method for single-image-based 3D mesh generation (stage 1: coarse 3D mesh reconstruction using 3D LRMs, stage 2: geometry enhancement, stage 3: texture enhancement).

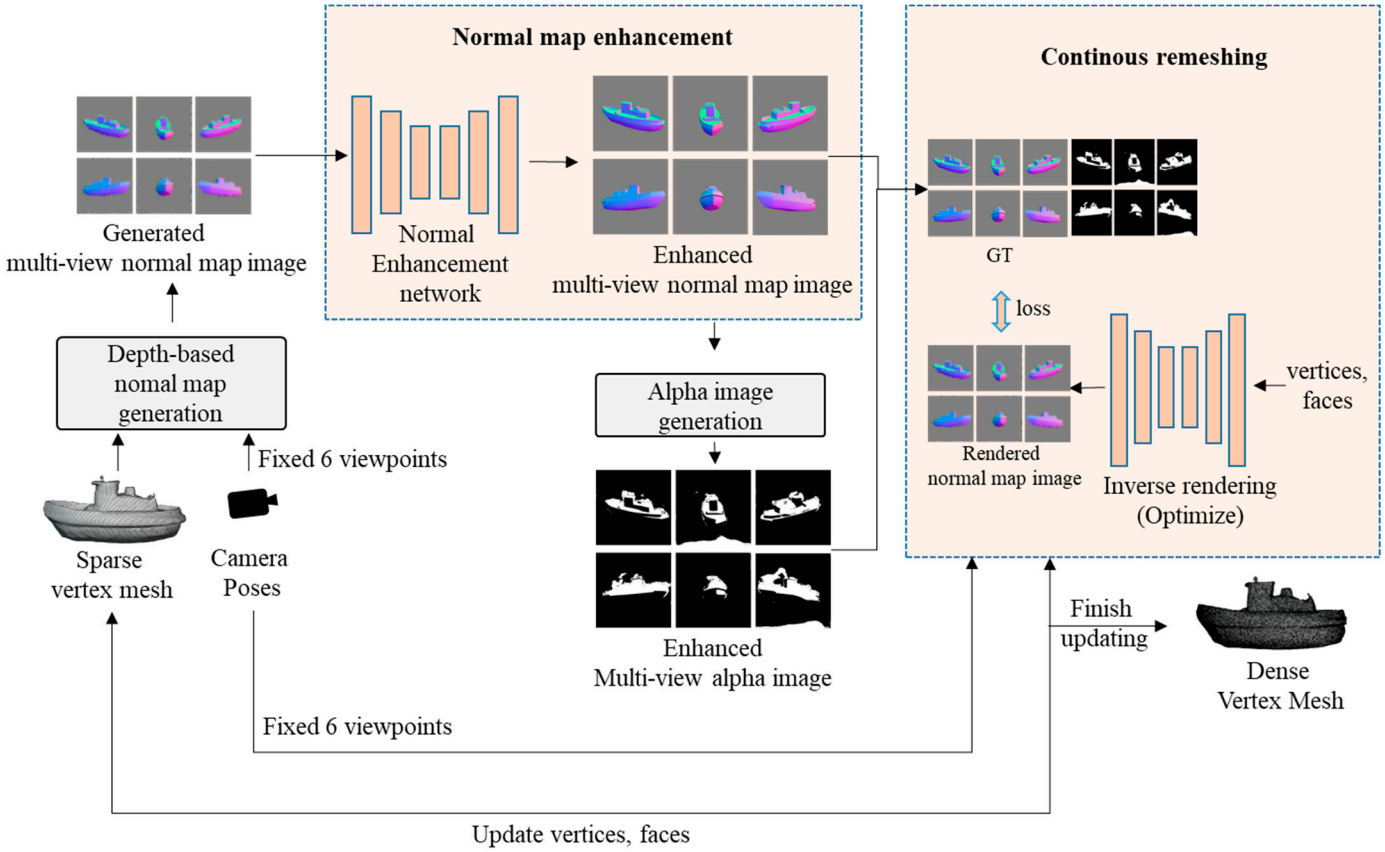


Figure 2. Proposed geometry enhancement for the joint optimization process.

3.2.2. Continuous Remeshing

Inspired by [54], this approach processes a vertex optimization through continuous remeshing using the multivew enhanced normal map images. The mesh is optimized by manipulating its vertices (V) and faces (F); the optimization is guided by enhanced normal maps $\hat{n}_i$. During optimization, differentiable rendering is used to render normal maps from the current mesh, denoted as $R_n(V, F, \pi_i)$, where π_i represents the camera information for the i-th rendering. The goal is to minimize the Mean Squared Error (MSE) loss difference between the rendered and enhanced normals, which is expressed as

$$L_{remeshing} = \sum_i \|\hat{n}_i - R_n(V, F, \pi_i)\|_1^1 \quad (2)$$

During each iteration, vertex positions are updated on the basis of the gradients from the backward loss process, followed by remeshing operations, such as edge splitting, merging, and flipping, as proposed in [54]. During the optimization process, the vertices and faces of the sparse vertex mesh are progressively updated, leading to the formation of a dense vertex mesh upon completion. The dense vertex mesh represents the final outcome of the remeshing procedure, wherein the shape is optimized using enhanced normal map images, thereby improving the level of detail in the 3D mesh.

3.3. Texture Enhancement for Joint Optimization

As shown in Figure 3, the generated dense vertex mesh must be initialized through dense mesh texture mapping using sparse vertices and texture maps. Thereafter, the low-resolution dense mesh texture map is enhanced to a high-resolution texture map by using an RGB enhancement model and a texture enhancement model with a renderer. The details of each component are as follows.

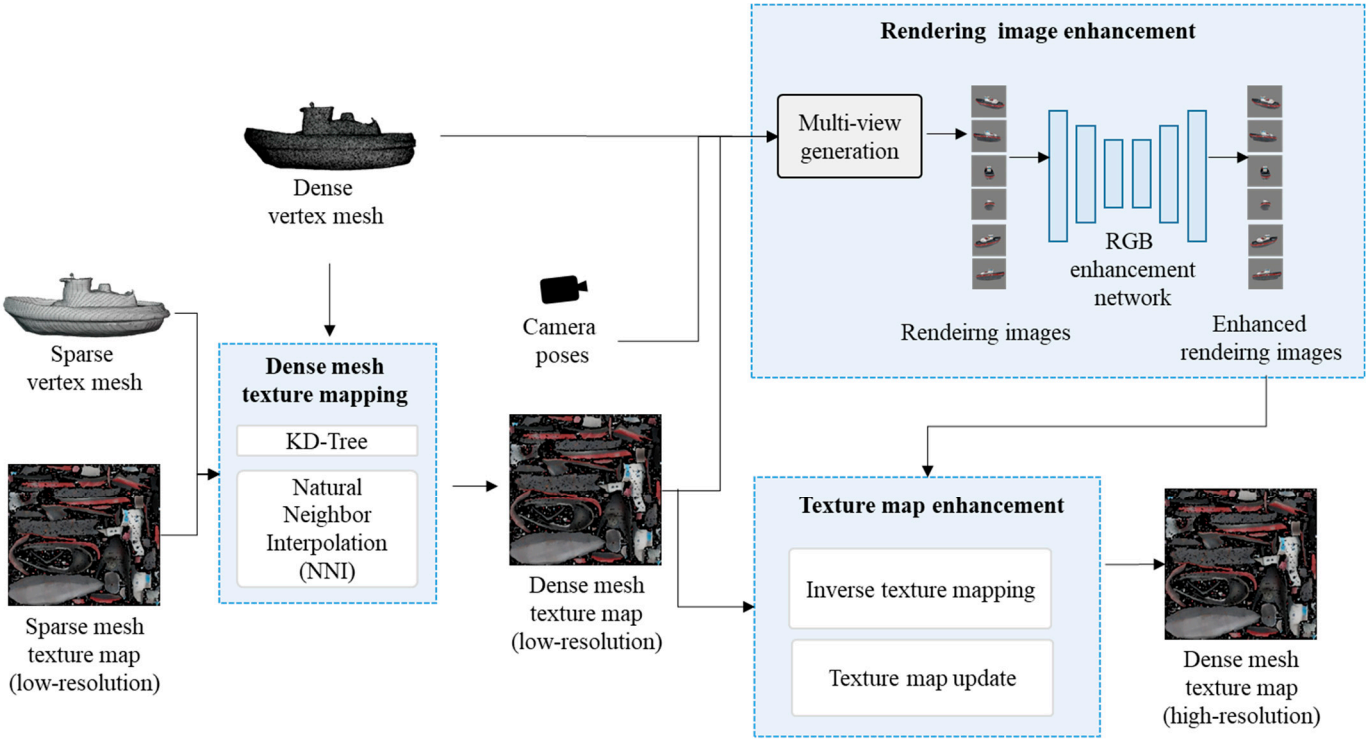


Figure 3. Proposed texture enhancement for the joint optimization process.

3.3.1. Dense Mesh Texture Mapping

Dense mesh texture mapping using a kd-tree and natural neighbor interpolation (NNI) involves transferring the texture from the source to the target mesh by leveraging geometric relationships. First, a kd-tree is built using the 3D coordinates of the vertices in the source mesh to enable efficient nearest-neighbor searches. For each vertex in the target mesh, the vertex closest to the source mesh is determined using the kd-tree. The NNI is then applied to interpolate the texture values (e.g., colors or UV coordinates) based on neighboring vertices from the source mesh, ensuring smooth transitions and natural texture mapping across the target mesh. Finally, the interpolated texture values are projected onto the target mesh using optional postprocessing steps, such as smoothing, to enhance the quality of the mapping. This approach enables accurate texture transfer between meshes while maintaining geometric and visual consistency.

3.3.2. Rendering Image Enhancement

For rendering image enhancement, the rendered images are captured from six fixed viewpoints: front, back, left, right, top, and bottom. These six rendered images are then enhanced using a multiview conditional diffusion model based on a stable diffusion backbone individually. Cross-attention modules are employed to capture global features that remain consistent regardless of the viewpoint, allowing for a more robust understanding of the object. The denoising U-Net extracts local features from the multiview inputs, which are integrated into the self-attention module to encode 3D correspondences. The 2D self-attention layers of stable diffusion are extended to 3D to facilitate interactions between views, and the camera poses are encoded using a two-layer MLP and added to the time embedding.

3.3.3. Texture Enhancement

To enhance specific parts of a texture map via image rendering, we apply projective texture mapping. First, the enhanced rendering images captured from the fixed six viewpoints (front, back, left, right, top, and bottom) are to be projected onto the 3D scene. Inverse texture mapping is used to determine which texture coordinates correspond to the

enhanced image pixels, utilizing the projection matrix from rendering. The texture map is updated by applying the enhanced pixels to the correct coordinates, blending new and original textures if needed. Finally, the object is rerendered with the updated texture map to verify the changes and make any additional adjustments.

4. Experiments

4.1. Experimental Settings

4.1.1. Dataset

We quantitatively evaluated the performance of the proposed model using two public datasets: Google Scanned Objects (GSO) [55] and ShapeNet [56]. The GSO dataset contains approximately 1000 objects, from which we randomly selected 50 objects for the evaluation set. For ShapeNet, we curated 25 boat objects and 25 car objects by specifically organizing a list of items tagged as "ship" and "warship" for the boat data. We selected 50 objects from the ShapeNet dataset because boats and cars share a common 3D shape but differ in surface details. The ShapeNet dataset, consisting of 50 objects categorized as boats and cars, allows us to verify whether the surface and texture improve, while the GSO dataset, composed of 50 randomly selected object types, demonstrates that the algorithm can generate various object types. We created two image evaluation sets for GSO and ShapeNet to assess the 2D visual quality of the generated 3D meshes. Specifically, the camera elevation was fixed at 0° for each object. For the eight generated views, the azimuth was set to 45-degree intervals, resulting in eight images. Each image was rendered with a resolution of 1024×1024 pixels.

4.1.2. Evaluation Metrics

We selected five metrics to compare the quality of 2D and 3D geometries. For the 2D evaluation, we compared eight images rendered from the ground truth and generated 3D meshes at the same angles using the PSNR, SSIM [57], and LPIPS [58] as metrics. The peak signal-to-noise ratio (PSNR) is used to evaluate the amount of image quality loss between two images by normalizing the error between their pixels. A smaller error indicates greater similarity between two images. The structural similarity index measure (SSIM) evaluates the correlation between two images (x, y) in terms of three aspects: luminance, contrast, and structure (with $\alpha = \beta = \gamma = 1$). A higher SSIM value indicates better image quality, and it is used to assess the degree of quality loss between two images. Learned perceptual image patch similarity (LPIPS) uses the results from a pretrained neural network to evaluate the similarity between two images. The two images are passed through a VGG network, and feature values from intermediate layers are extracted. The similarity of these features is then measured as the evaluation metric. For a 3D evaluation comparison, we adopted the chamfer distance (CD). CD was adopted for 3D evaluation and measures the similarity of 3D shapes. It assesses the similarity between points sampled from a 3D model based on the distance between the nearest points. A lower value indicates greater similarity. We aligned the coordinate system of the generated mesh with the ground truth and repositioned and rescaled all the meshes, including the ground truth, to fit within a cube of size $[-1, 1]^3$. We then uniformly sampled 16k points from the surface of the meshes in this standardized environment for evaluation.

4.2. Experimental Results

4.2.1. Results of the Joint Optimization-Based Texture and Geometry Enhancement

Figure 4 shows the results of the proposed joint optimization-based texture and geometry enhancement. In the first and second rows, where no geometry optimization is applied, the ship and car lack detailed features such as the ship's window and the car's wheel. However, with the application of geometry optimization, as shown on the right, the details of the window and wheel are enhanced. Additionally, in the third and fourth rows, where texture enhancement is applied, the generated texture closely resembles the original input image, demonstrating a more similar color and enhanced texture details.

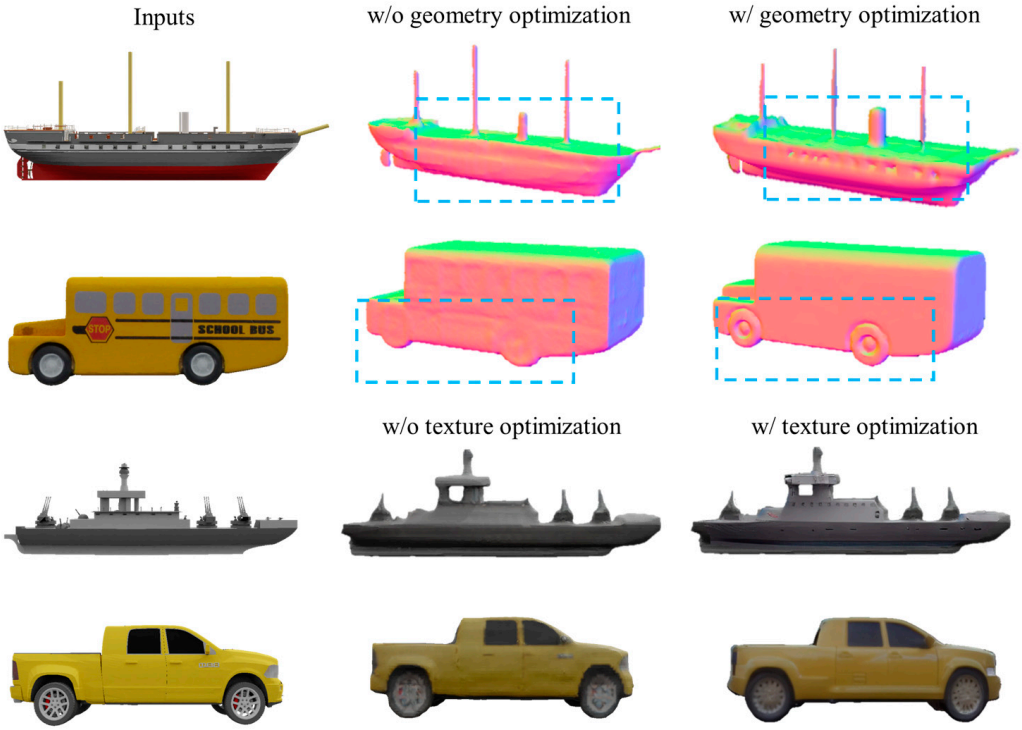


Figure 4. Qualitative results of the proposed joint optimization method.

4.2.2. Quantitative and Qualitative Results on the ShapeNet [55] Dataset

Table 1 presents the evaluation of the 3D estimation using the ShapeNet [55] dataset. The proposed method achieved the highest PSNR, SSIM, LPIPS, and CD, significantly outperforming the other methods. This demonstrates the superior performance of the proposed approach in accurately estimating and reconstructing 3D objects, with improved image quality and geometric accuracy compared with existing techniques. The high PSNR and SSIM values indicate better fidelity and structural preservation in the reconstructed 3D models with enhanced texture fidelity, whereas the low CD scores reflect a more precise 3D geometric correspondence.

Table 1. Quantitative results on the ShapeNet [55] dataset.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	CD ↓
TripoSR [5]	20.499	0.906	0.094	0.00556
CRM [6]	20.412	0.914	0.097	0.01276
InstantMesh [7]	21.278	0.923	0.089	0.00867
Real3D [59]	20.717	0.910	0.094	0.00448
Ours	21.289	0.933	0.088	0.00430

Figure 5 provides visual comparisons between the performances of related 3D generation models and the proposed method on the ShapeNet [55] dataset for the experiments. These comparisons highlight the effectiveness of the proposed approach in generating more accurate 3D mesh models, as evidenced by the finer details, improved surface representation, and enhanced texture fidelity derived from the input images.



Figure 5. Qualitative comparison results on the ShapeNet dataset: (**a**) TripoSR [5], (**b**) CRM [6], (**c**) InstantMesh [7], (**d**) Real3D [59], and (**e**) Ous.

4.2.3. Quantitative and Qualitative Results on the GSO [54] Dataset

Table 2 presents the evaluation of the 3D semantic estimation using the GSO [54] dataset. This demonstrates the superior performance of the proposed approach in accurately estimating and reconstructing 3D objects, with improved image quality and geometric accuracy compared with existing techniques. This confirms that the approach is applicable to both the ShapeNet [55] and GSO [54] datasets.

Table 2. Quantitative results on the GSO dataset.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	CD ↓
TripoSR [5]	20.341	0.874	0.142	0.01821
CRM [6]	19.665	0.874	0.150	0.02726
InstantMesh [7]	20.120	0.880	0.142	0.02015
Real3D [59]	19.967	0.867	0.145	0.02222
Ours	20.423	0.882	0.141	0.01713

Figure 6 provides a visual comparison of the performances of TripoSR [5], CRM [6], InstantMesh [7], and Real3D [59] and the performance of the proposed method on the GSO dataset. The results show that, on the GSO [54] dataset, the proposed method generates 3D meshes with superior detail and color accuracy compared with the other approaches.

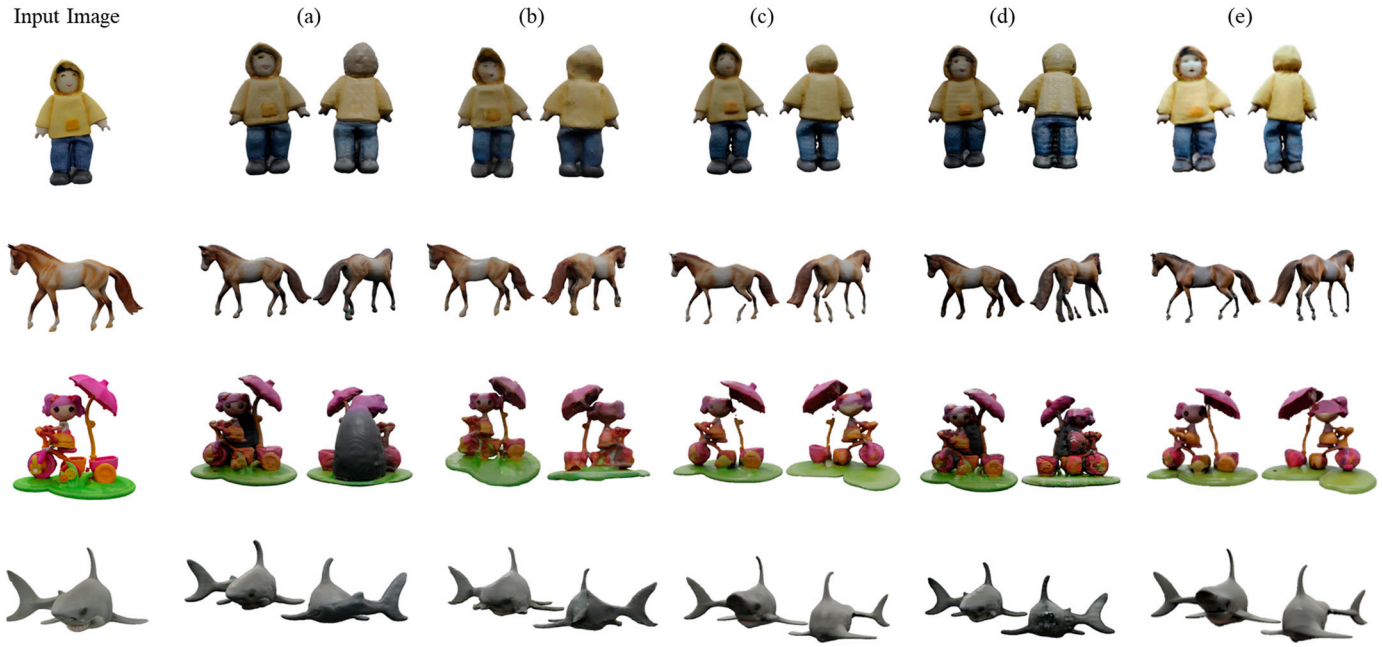


Figure 6. Qualitative comparison results on the GSO dataset: (**a**) TripoSR [5], (**b**) CRM [6], (**c**) InstantMesh [7], (**d**) Real3D [59], and (**e**) Ous.

5. Conclusions

In this study, we introduced a joint optimization method to enhance the geometry and texture of a 3D mesh generated from a single image. The proposed approach, which combines inverse rendering with generalized 3D mesh generation, significantly improves the accuracy of 3D shape and texture estimation. By addressing the limitations of fixed-resolution meshes and adapting image features for specific networks, the proposed method demonstrated strong performance across various datasets. This advancement not only enhances the detail and texture quality but also provides a valuable contribution to the field of 3D reconstruction. In the experimental results, objects such as cars, buses, and boats, which do not have significant differences in width and height, tend to have their 3D models generated to closely resemble their actual shapes in the real world. However, for larger vessels such as ships, the width is significantly broader than the height, which can result in a reduced size ratio of the ship within the image when input to fit the width. This leads to decreased detail or distorted 3D shapes in the final results. Future research will focus on addressing the issue of detail reduction caused by large discrepancies in width and height ratios, as seen in 3D models of ships. Additionally, we plan to expand the proposed methods to larger and more diverse datasets, apply inverse rendering tailored to various lighting conditions and materials, and improve computational efficiency for broader applications.

Author Contributions: Conceptualization, J.P.; methodology, J.P.; software, J.P., M.K., J.K. and W.K.; validation, J.P., M.K., J.K. and W.K.; writing—original draft preparation, J.P., M.K. and J.K.; writing—review and editing, J.P.; visualization, J.P., M.K. and J.K.; supervision, K.C.; project administration, K.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Future Challenge Program through the Agency for Defense Development funded by the Defense Acquisition Program Administration.

Data Availability Statement: The datasets used in this study are publicly available. The Google Scanned Objects (GSO) dataset can be accessed at https://app.gazebosim.org/GoogleResearch/fuel/collections/Scanned%20Objects%20by%20Google%20Research, accessed on 11 September 2024, and the ShapeNet dataset is available at https://shapenet.org/, accessed on 11 September 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Furukawa, Y.; Ponce, J. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 1362–1376. [CrossRef] [PubMed]
2. Mildenhall, B.; Srinivasan, P.P.; Tancik, M.; Barron, J.T.; Ramamoorthi, R.; Ng, R. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* **2021**, *65*, 99–106. [CrossRef]
3. Rakotosaona, M.J.; Manhardt, F.; Arroyo, D.M.; Niemeyer, M.; Kundu, A.; Tombari, F. Nerfmeshing: Distilling neural radiance fields into geometricallyaccurate 3d meshes. In Proceedings of the 2024 International Conference on 3D Vision (3DV), Davos, Switzerland, 18–21 March 2024.
4. Yariv, L.; Hedman, P.; Reiser, C.; Verbin, D.; Srinivasan, P.P.; Szeliski, R.; Barron, J.T.; Mildenhall, B. Bakedsdf: Meshing neural sdfs for real-time view synthesis. *arXiv* **2023**, arXiv:2302.14859.
5. Tochilkin, D.; Pankratz, D.; Liu, Z.; Huang, Z.; Letts, A.; Li, Y.; Liang, D.; Laforte, C.; Jampani, V.; Cao, Y.-P. TripoSR: Fast 3D Object Reconstruction from a Single Image. *arXiv* **2024**, arXiv:2403.02151.
6. Wang, Z.; Wang, Y.; Chen, Y.; Xiang, C.; Chen, S.; Yu, D.; Li, C.; Su, H.; Zhu, J. CRM: Single Image to 3D Textured Mesh with Convolutional Reconstruction Model. *arXiv* **2024**, arXiv:2403.05034.
7. Xu, J.; Cheng, W.; Gao, Y.; Wang, X.; Gao, S.; Shan, Y. InstantMesh: Efficient 3D Mesh Generation from a Single Image with Sparse-view Large Reconstruction Models. *arXiv* **2024**, arXiv:2404.07191.
8. Li, J.; Tan, H.; Zhang, K.; Xu, Z.; Luan, F.; Xu, Y.; Hong, Y.; Sunkavalli, K.; Shakhnarovich, G.; Bi, S. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
9. Hong, Y.; Zhang, K.; Gu, J.; Bi, S.; Zhou, Y.; Liu, D.; Liu, F.; Sunkavalli, K.; Bui, T.; Tan, H. LRM: Large reconstruction model for single image to 3d. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
10. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, 18–24 June 2022; IEEE: New York, NY, USA, 2022; pp. 10674–10685.
11. Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; Sutskever, I. Zero-shot text-to-image generation. In Proceedings of the 38th International Conference on Machine Learning, Virtual, 18–24 July 2021; PMLR: New York, NY, USA, 2021; pp. 8821–8831.
12. Luo, S.; Hu, W. Diffusion probabilistic models for 3d point cloud generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 2837–2845.
13. Nichol, A.; Jun, H.; Dhariwal, P.; Mishkin, P.; Chen, M. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv* **2022**, arXiv:2212.08751.
14. Zhou, L.; Du, Y.; Wu, J. 3d shape generation and completion through point-voxel diffusion. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 5826–5835.
15. Choy, C.B.; Xu, D.; Gwak, J.; Chen, K.; Savarese, S. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Computer Vision–ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016, Proceedings, Part VIII 14*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 628–644.
16. Tulsiani, S.; Zhou, T.; Efros, A.A.; Malik, J. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2626–2634.
17. Mildenhall, B.; Srinivasan, P.P.; Ortiz-Cayon, R.; Kalantari, N.K.; Ramamoorthi, R.; Ng, R.; Kar, A.-H. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Trans. Graph. (TOG)* **2019**, *38*, 1–14. [CrossRef]
18. Tucker, R.; Snavely, N. Single-view view synthesis with multiplane images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 551–560.
19. Liu, Z.; Feng, Y.; Black, M.J.; Nowrouzezahrai, D.; Paull, L.; Liu, W. Meshdiffusion: Score-based generative 3d mesh modeling. In Proceedings of the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
20. Liu, S.; Li, T.; Chen, W.; Li, H. Soft rasterizer: A differentiable renderer for image based 3d reasoning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7708–7717.
21. Kerbl, B.; Kopanas, G.; Leimkühler, T.; Drettakis, G. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.* **2023**, *42*, 1–14. [CrossRef]
22. Cheng, Y.-C.; Lee, H.-Y.; Tulyakov, S.; Schwing, A.G.; Gui, L.-Y. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 4456–4465.
23. Chou, G.; Bahat, Y.; Heide, F. Diffusion-sdf: Conditional generative modeling of signed distance functions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 2262–2272.
24. Jun, H.; Nichol, A. Shap-e: Generating conditional 3d implicit functions. *arXiv* **2023**, arXiv:2305.02463.
25. Müller, N.; Siddiqui, Y.; Porzi, L.; Bulo, S.R.; Kontschieder, P.; Nießner, M. Diffrf: Rendering-guided 3d radiance field diffusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 4328–4338.

26. Zhang, B.; Tang, J.; Niessner, M.; Wonka, P. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Trans. Graph. (TOG)* **2023**, *42*, 1–16. [CrossRef]
27. Gupta, A.; Xiong, W.; Nie, Y.; Jones, I.; Oguz, B. 3dgen: Triplane latent diffusion for textured mesh generation. *arXiv* **2023**, arXiv:2303.05371.
28. Karnewar, A.; Mitra, N.J.; Vedaldi, A.; Novotny, D. Holofusion: Towards photo-realistic 3d generative modeling. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023.
29. Kim, S.W.; Brown, B.; Yin, K.; Kreis, K.; Schwarz, K.; Li, D.; Rombach, R.; Torralba, A.; Fidler, S. Neuralfield-ldm: Scene generation with hierarchical latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023.
30. Poole, B.; Jain, A.; Barron, J.T.; Mildenhall, B. Dreamfusion: Text-to-3d using 2d diffusion. In Proceedings of the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
31. Lin, C.-H.; Gao, J.; Tang, L.; Takikawa, T.; Zeng, X.; Huang, X.; Kreis, K.; Fidler, S.; Liu, M.-Y.; Lin, T.-Y. Magic3d: High-resolution text-to-3d content creation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 300–309.
32. Shi, Y.; Wang, P.; Ye, J.; Long, M.; Li, K.; Yang, X. Mvdream: Multi-view diffusion for 3d generation. *arXiv* **2023**, arXiv:2308.16512.
33. Liu, R.; Wu, R.; Van Hoorick, B.; Tokmakov, P.; Zakharov, S.; Vondrick, C. Zero-1-to-3: Zero-shot One Image to 3D Object. *arXiv* **2023**, arXiv:2303.11328.
34. Szymanowicz, S.; Rupprecht, C.; Vedaldi, A. Viewset diffusion:(0-) image-conditioned 3d generative models from 2d data. *arXiv* **2023**, arXiv:2306.07881.
35. Liu, Y.; Lin, C.; Zeng, Z.; Long, X.; Liu, L.; Komura, T.; Wang, W. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv* **2023**, arXiv:2309.03453.
36. Shi, R.; Chen, H.; Zhang, Z.; Liu, M.; Xu, C.; Wei, X.; Chen, L.; Zeng, C.; Su, H. Zero123++: A Single Image to Consistent Multi-view Diffusion Base Model. *arXiv* **2023**, arXiv:2310.15110.
37. Sohail, S.S.; Himeur, Y.; Kheddar, H.; Amira, A.; Fadli, F.; Atalla, S.; Copiaco, A.; Mansoor, W. Advancing 3D point cloud understanding through deep transfer learning: A comprehensive survey. *Inf. Fusion* **2024**, *113*, 102601. [CrossRef]
38. Wang, P.; Shi, Y. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv* **2023**, arXiv:2312.02201.
39. Tang, J.; Chen, Z.; Chen, X.; Wang, T.; Zeng, G.; Liu, Z. Lgm: Large multi-view Gaussian model for high-resolution 3d content creation. *arXiv* **2024**, arXiv:2402.05054.
40. Xu, Y.; Shi, Z.; Yifan, W.; Chen, H.; Yang, C.; Peng, S.; Shen, Y.; Wetzstein, G. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. *arXiv* **2024**, arXiv:2403.14621.
41. Dong, Y.; Chen, G.; Peers, P.; Zhang, J.; Tong, X. Appearance-from-motion: Recovering spatially varying surface reflectance under unknown lighting. *ACM Trans. Graph. (TOG)* **2014**, *33*, 1–12. [CrossRef]
42. Bi, S.; Xu, Z.; Sunkavalli, K.; Hašan, M.; Hold-Geoffroy, Y.; Kriegman, D.; Ramamoorthi, R. Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. In *Computer Vision–ECCV 2020, Proceedings of the 16th European Conference, Glasgow, UK, 23–28 August 2020, Proceedings, Part III 16*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 294–311.
43. Bi, S.; Xu, Z.; Srinivasan, P.; Mildenhall, B.; Sunkavalli, K.; Hašan, M.; Hold-Geoffroy, Y.; Kriegman, D.; Ramamoorthi, R. Neural reflectance fields for appearance acquisition. *arXiv* **2020**, arXiv:2008.03824.
44. Zhang, K.; Luan, F.; Wang, Q.; Bala, K.; Snavely, N. Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5453–5462.
45. Srinivasan, P.P.; Deng, B.; Zhang, X.; Tancik, M.; Mildenhall, B.; Barron, J.T. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 7495–7504.
46. Zhang, Y.; Sun, J.; He, X.; Fu, H.; Jia, R.; Zhou, X. Modeling indirect illumination for inverse rendering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 18643–18652.
47. Jin, H.; Liu, I.; Xu, P.; Zhang, X.; Han, S.; Bi, S.; Zhou, X.; Xu, Z.; Su, H. Tensoir: Tensorial inverse rendering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 165–174.
48. Chen, A.; Xu, Z.; Geiger, A.; Yu, J.; Su, H. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 333–350.
49. Keselman, L.; Hebert, M. Approximate differentiable rendering with algebraic surfaces. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 596–614.
50. Keselman, L.; Hebert, M. Flexible techniques for differentiable rendering with 3d gaussians. *arXiv* **2023**, arXiv:2308.14737.
51. Liang, Z.; Zhang, Q.; Feng, Y.; Shan, Y.; Jia, K. Gs-ir: 3d gaussian splatting for inverse rendering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024.
52. Zhang, L.; Rao, A.; Agrawala, M. Adding conditional control to text-to-image diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 3836–3847.
53. Qiu, L.; Chen, G.; Gu, X.; Zuo, Q.; Xu, M.; Wu, Y.; Yuan, W.; Dong, Z.; Bo, L.; Han, X. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 17–21 June 2024.

54. Palfinger, W. Continuous remeshing for inverse rendering. *Comput. Animat. Virtual Worlds* **2022**, *33*, e2101. [CrossRef]
55. Downs, L.; Francis, A.; Koenig, N.; Kinman, B.; Hickman, R.; Reymann, K.; McHugh, T.B.; Vanhoucke, V. Google scanned objects: A high quality dataset of 3d scanned household items. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; IEEE: New York, NY, USA, 2022; pp. 2553–2560.
56. Chang, A.X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. Shapenet: An information-rich 3d model repository. *arXiv* **2015**, arXiv:1512.03012.
57. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]
58. Zhang, R.; Isola, P.; Efros, A.A.; Shechtman, E.; Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 586–595.
59. Jiang, H.; Huang, Q.; Pavlakos, G. Real3D: Scaling Up Large Reconstruction Models with Real-World Images. *arXiv* **2024**, arXiv:2406.08479.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.