PRM: Photometric Stereo based Large Reconstruction Model

Supplementary Material

A. BRDF Parameterization

In Sec. 3.1 we introduce the D, F and G term of the specular component of BRDF property. We implement the Cook-Torrance BRDF model [5]. The basic specular albedo $F_0 = (m * \boldsymbol{a} + (1 - m) * 0.04)$, where $\boldsymbol{a}$ is the albedo and m is the metalness. The Fresnel term (F) is defined as:

$$F = F_0 + (1 - F_0)(1 - (\boldsymbol{h} \cdot \boldsymbol{\omega}_o))^5, \tag{13}$$

where $\boldsymbol{h}$ is the half-way vector between ω_o and viewing direction $\boldsymbol{\omega}_i$. The normal distribution function D is Trowbridge-Reitz GGX distribution as

$$D(h) = \frac{\alpha^2}{\pi \left((\mathbf{n} \cdot \mathbf{h})^2(\alpha^2 - 1) + 1\right)^2}, \tag{14}$$

where $\alpha = \rho^2$, $\mathbf{n}$ is the surface normal. The geometry term G is the Schlick-GGX Geometry function:

$$G(\boldsymbol{n}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i, k) = G_{\text{sub}}(\boldsymbol{n}, \boldsymbol{\omega}_o, k) G_{\text{sub}}(\boldsymbol{n}, \boldsymbol{\omega}_i, k), \tag{15}$$

where G_{sub} is given by:

$$G_{\text{sub}}(\boldsymbol{n}, \boldsymbol{\omega}, k) = \frac{\boldsymbol{n} \cdot \boldsymbol{\omega}}{(\boldsymbol{n} \cdot \boldsymbol{\omega})(1 - k) + k}, \tag{16}$$

where k is a parameter related to the roughness ρ, often approximated as $k = \frac{\rho^4}{2}$.

B. Optimization and Additional Model Details

Optimization Details. We used Adam [24] as our optimizer. In the first stage, the learning rate was set to 4×10^{-5}. In the second stage, the learning rate was set to 4×10^{-6} for finetuning. We used 32 NVIDIA A800 GPUs in the first stage for nerf training with a batch size of 256 for 100K steps, taking about 7 days. In the second stage, We used 32 NVIDIA A800 GPUs to finetune the model from the first stage with a batch size of 256 for 30K steps, taking about 3 days.

Network architecture. Our network architecture is similar to that of InstantMesh [48], consisting of a pre-trained DINO that encodes images into image tokens, and an image-to-triplane transformer decoder that projects these 2D image tokens onto a 3D triplane using cross-attention. Furthermore, three MLPs are utilized, taking interpolated triplane features as input and outputting albedo, SDF, deformation, and weights. These outputs are required by FlexiCube for mesh extraction and subsequent rendering. The details of the network is shown in Figure 7. Our final model is a large transformer with 16 attention layers, with feature dimension 1024. The size of triplane is $64 \times 64 \times 3$ with 80 channels. The grid size for FlexiCube was set to 128. The resolution of input images was 512.

C. Training Strategy

Camera Augmentation. Previous LRMs typically prepare training data by rendering images with fixed Fields-Of-View (FOVs) and camera distances, making the models sensitive to changes in these variables during inference. Since we adopt a real-time rendering method and mesh rasterization for fast online rendering, we can readily adjust the FOVs and camera distances during training. This training strategy enhances our model's robustness to variations in camera embeddings. We provide some examples in the following section.

Random materials and lighting. During inference, one option for 3D mesh reconstruction is to leverage a multi-view diffusion model to generate multi-view images. However, these images may exhibit inconsistencies in materials or lighting. To ensure our model remains robust to these inconsistencies, we randomly change the materials and lighting when rendering each view during training. Alternatively, the lighting and materials of the input images are consistent. Our model need to handle this scenario. Therefore, we establish a threshold to ensure that the rendered multi-view images potentially share the same materials and lighting. Specifically, when rendering each view, there is a 50% probability that the materials and lighting will change. This arrangement means that each view may feature different materials or lighting. If no changes are made, all views are rendered with consistent materials and lighting.

D. Example of images with varying materials and illumination

In this section, we present examples of rendered images with varying materials and illumination along with intermediate shading variables, including specular lighting, diffuse lighting, albedo maps and environment maps, as shown in Figure 8. The red box highlights how varying roughness levels influence the specular lighting maps, affecting their frequency. Specifically, lower roughness (right) results in specular lighting of higher frequency.

E. Application Visualization

Since our method can reconstruct high-quality meshes with predicted albedo, it facilitates downstream applications such as relighting and material editing. We showcase some

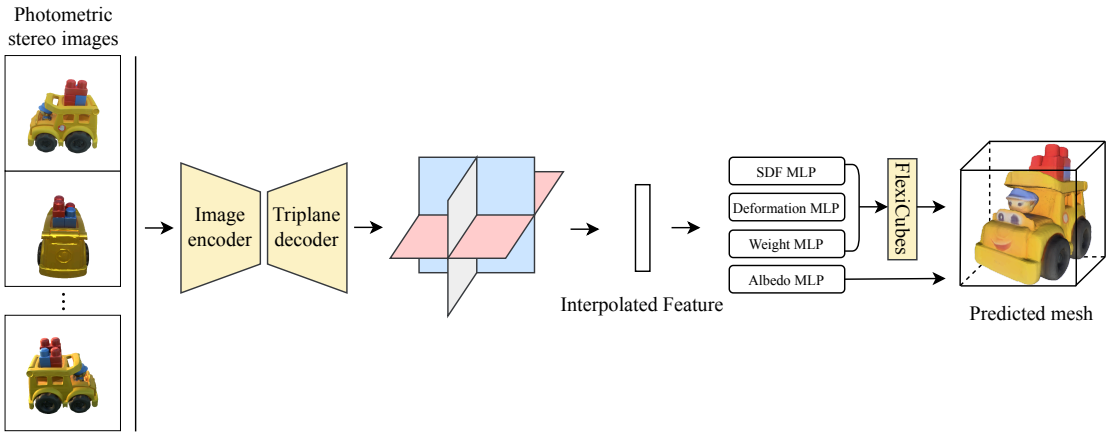

Figure 7. The details of network architecture.

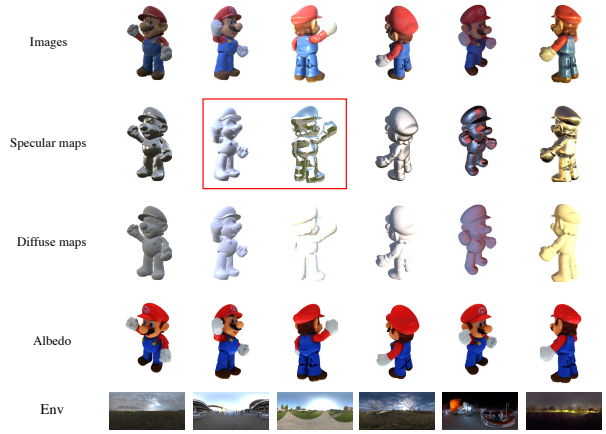

Figure 8. Examples of rendered images with varying materials and illumination, along with specular, diffuse lighting maps and albedo maps. The red box highlights how varying roughness levels influence the specular lighting maps, affecting their frequency. Specifically, lower roughness (right) results in specular lighting of higher frequency.

examples in Figure 9.

F. ROBUSTNESS Evaluation

Robustness to Camera Embedding. PRM exhibits robustness to variations in camera embedding. We compared PRM with InstantMesh by altering the camera embedding (i.e., FOV and camera radius) during inference. The results, shown in Figure 11, demonstrate that PRM maintains strong robustness to changes in camera embedding, whereas the performance of InstantMesh declines significantly when camera embedding varies.

Robustness to image appearance. PRM is robust to the image appearance. When handling specular surfaces, we

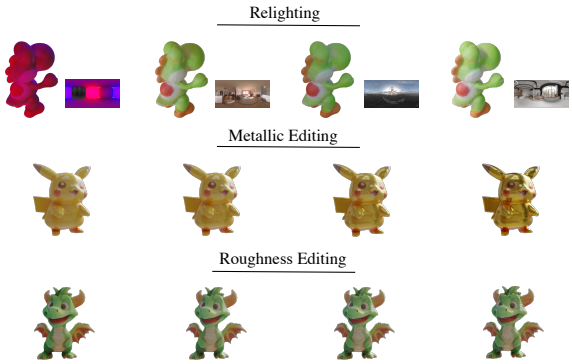

Figure 9. Application visualization. We show relighting and materials editing here.

can achieve correct geometry reconstruction. More visualization results can be found in Figure 12.

Robustness to spatially-varying materials. PRM is robust to the objects with spatially-varying materials for both synthetic and real-captured images. More visualization results can be found in Figure 17.

G. The effect of the number of camera views

We demonstrate the importance of varying camera poses for rendering multi-view images with varying materials and illumination as input. The number of input views is increased from 1 to 8. The qualitative results are illustrated in Figure 13. The quantitative results, including Chamfer Distance (CD) and F-Score, are depicted in Figure 10. When more images rendered under different camera views are inputted, we achieve better results; using 4 or 6 views provides the optimal balance between effectiveness and efficiency.

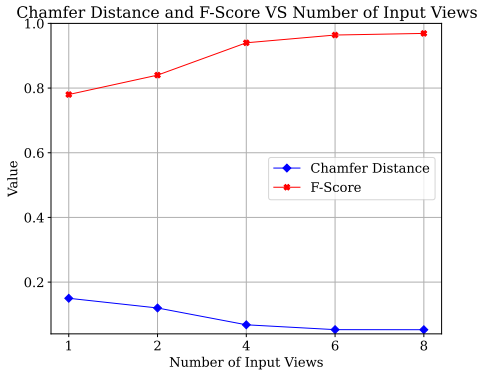

Figure 10. Ablation study of the effect of number of input views.

H. Failure Cases

Although effective, the performance of PRM is constrained by the quality of the multi-view images generated by the multi-view diffusion model when performing single image to 3D tasks. We illustrate a failure case in Figure 14. The lack of depth information in the input image leads to undesirable multi-view image generation, resulting in a reconstructed 3D mesh that lacks accuracy. A potential solution is to use the estimated depth to guide the multi-view images generation. We show an example of the estimated depth by DepthAnythingV2 [51] in Figure 15. Our method cannot handle multi-view images with background as shown in Figure 18, since we used images with while background as input during training as previous methods do. However, we can easily obtain images with while background by pretrained segmentation model.

I. More visualization results

We show more visualization results of PRM in Figure 16.

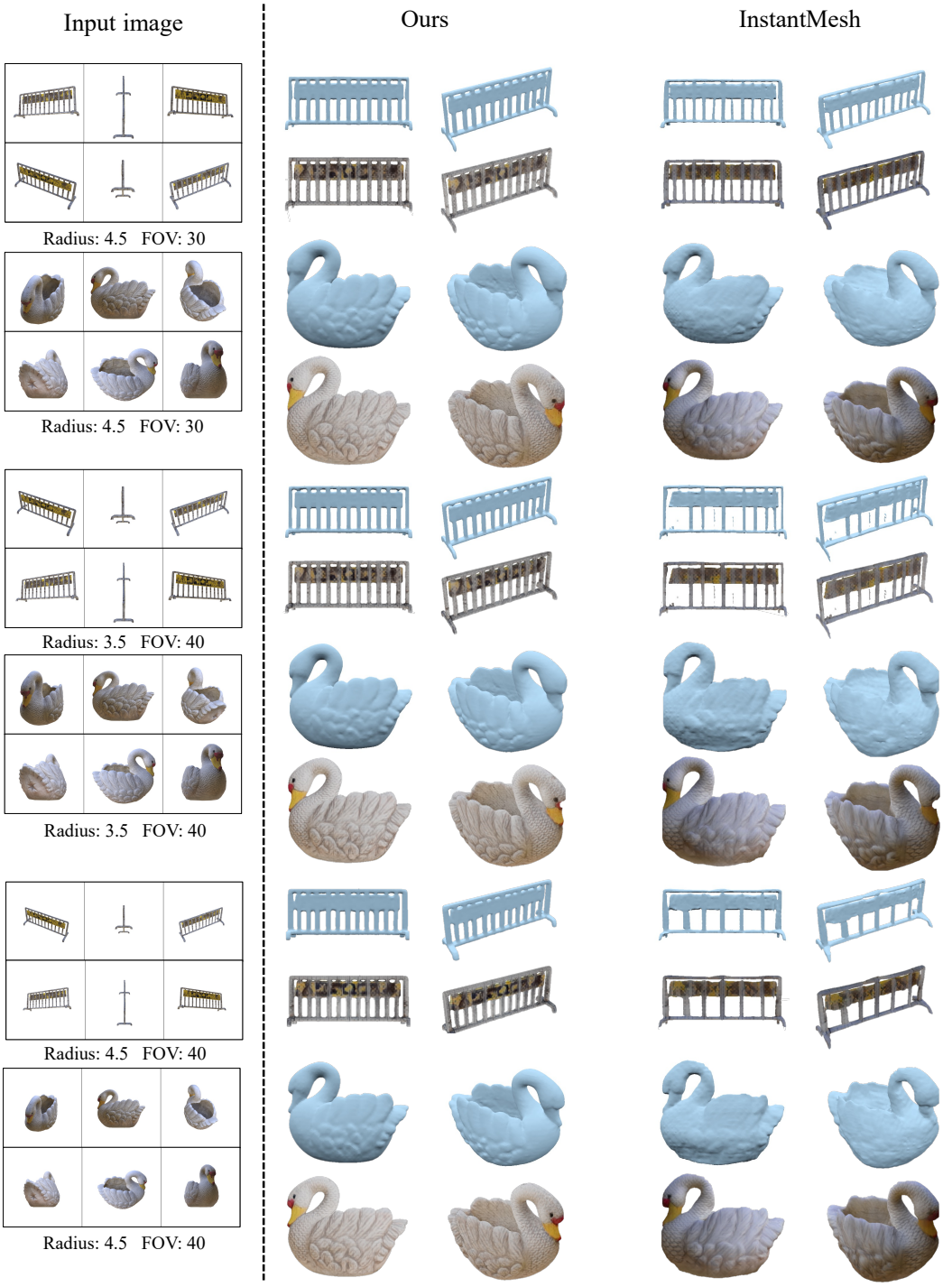

Figure 11. Comparison with InstantMesh when changing FOVs and camera radius: PRM demonstrates robustness to variations in camera embedding. Conversely, InstantMesh struggles when the radius and FOV differ from those used during training.

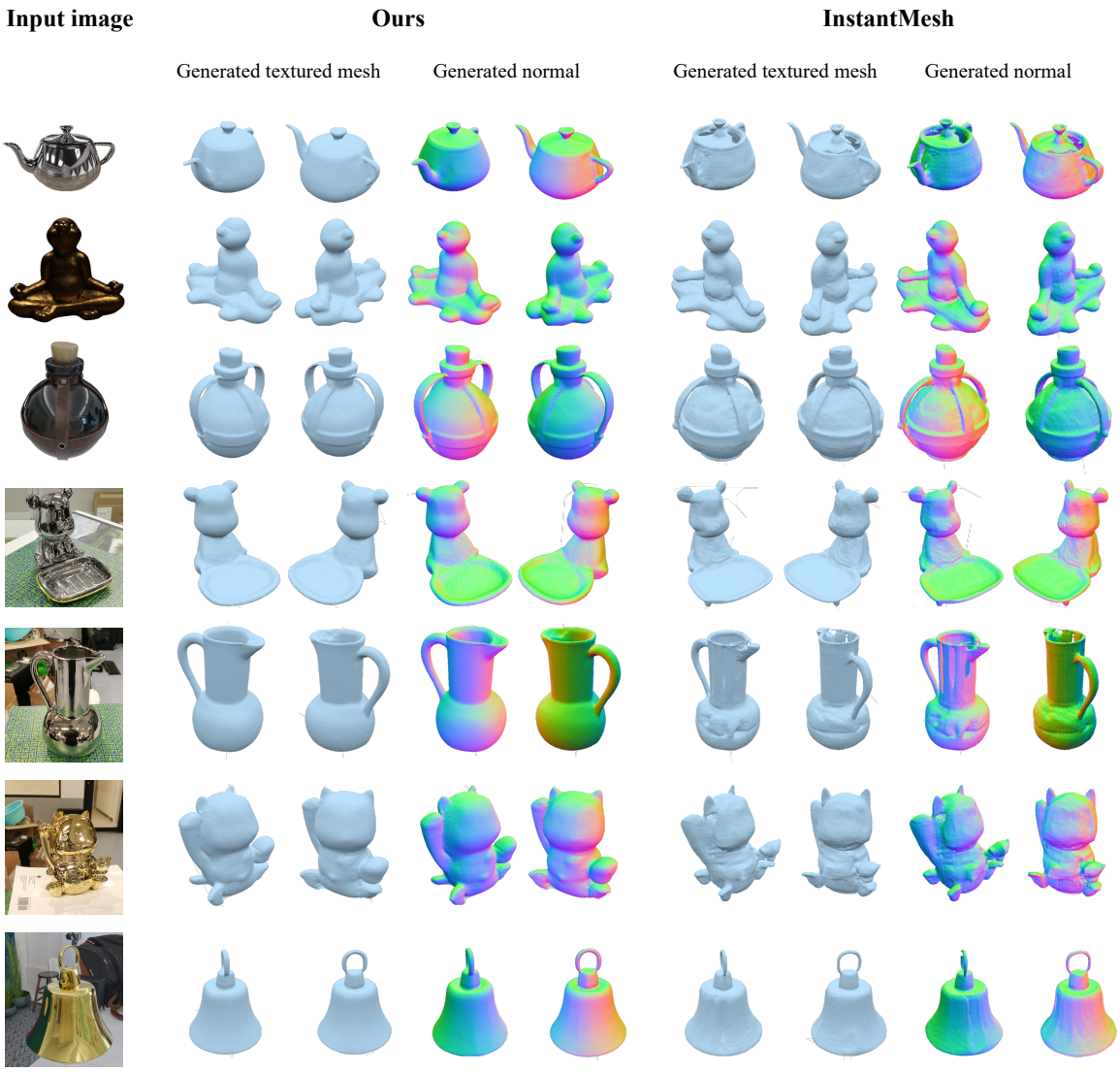

Figure 12. Single view reconstruction results using our method on input images with extreme conditions, such as specular highlights and shadows. Despite challenging lighting conditions, PRM successfully reconstructs the geometry and surface normals with high fidelity.

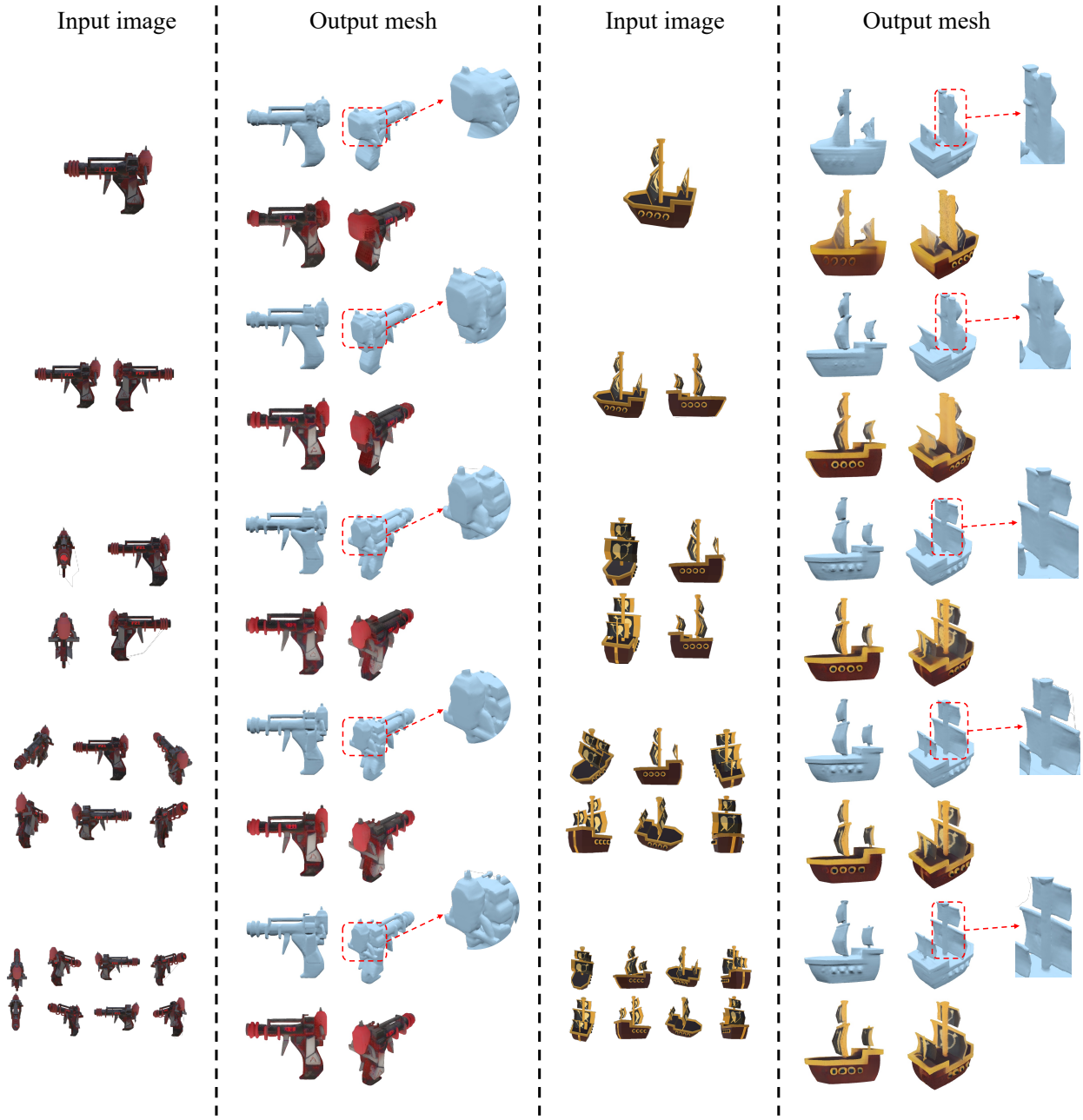

Figure 13. The effect of the number of input views. More views lead to better reconstruction result.

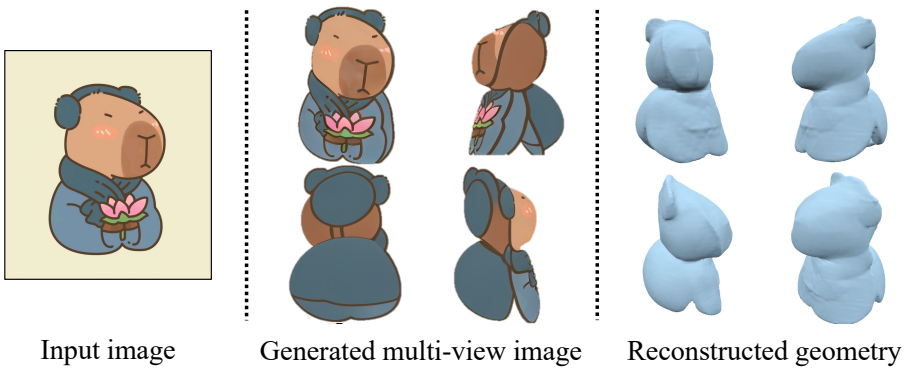

Figure 14. Illustration of a failure case.

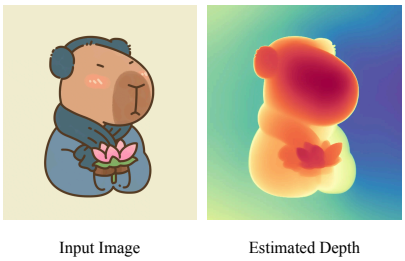

Figure 15. DepthAnythingV2 can estimate correct depth for image that lacks depth information, which may help multi-view diffusion model generate more reasonable multi-view images.

Figure 16. Visualization of more results of single view to 3D task.

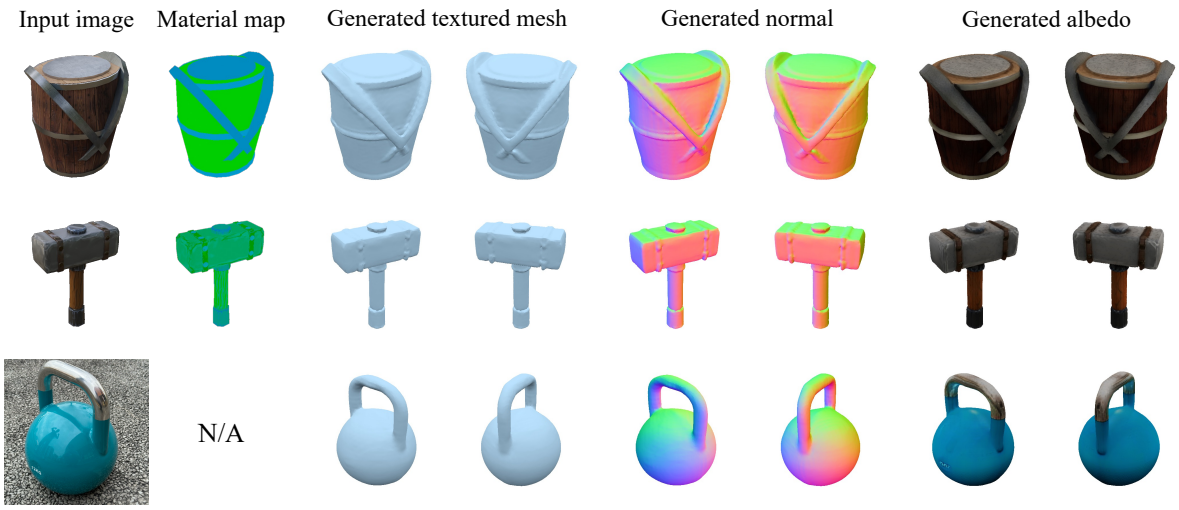

Figure 17. PRM can handle objects with spatially-varying materials for both synthetic and real-captured images.

Figure 18. Our method fails to handle images with natural background since we takes images with while background as input during training.