# Supplementary Materials for Wonder3D: Single Image to 3D using Cross-Domain Diffusion

Xiaoxiao Long[1,3*], Yuan-Chen Guo[2*‡], Cheng Lin[1†], Yuan Liu[1], Zhiyang Dou[1]
Lingjie Liu[4], Yuexin Ma[5], Song-Hai Zhang[2], Marc Habermann[6], Christian Theobalt[6], Wenping Wang[7†]

[1]The University of Hong Kong [2] Tsinghua University [3]VAST
[4]University of Pennsylvania [5]Shanghai Tech University [6]MPI Informatik [7]Texas A&M University
[*]Core contributions [†]Corresponding authors [‡]Intern at VAST
https://www.xxlong.site/Wonder3D/

## 1. Method Details.

### 1.1. Coordinate System

In practice, the target object is assumed to be placed along the gravity direction. **1) *Canonical coordinate system.*** Some prior works (e.g. MVDream and SyncDreamer) adopt a shared canonical system for all objects, whose axis $Z_c$ shares the same direction with gravity (Fig. 1 (a)). **2) *Input view related system.*** Wonder3D adopts an independent coordinate system for each object that is related to the input view. Its $Z_v$ and $X_v$ axes are aligned with the UV dimension of 2D input image space, and its $Y_v$ axis is vertical to the 2D image plane and passes through the center of ROI (Region of Interests) (Fig. 1 (b)). **3) *Camera poses.*** Wonder3D outputs 6 views $\{v_i, i = 0, ..., 5\}$ that are sampled at the $X_vOY_v$ plane of the input-view related system with a fixed radius, where the front view $v_0$ is initialized as input view and the other views are sampled with pre-defined azimuth degrees (see Fig. 1 (b)).

### 1.2. Textured Mesh Extraction

**Optimization Objectives.** With the obtained normal maps $G_{0:N}$ and color images $H_{0:N}$, we first leverage segmentation models to segment the object masks $M_{0:N}$ from the normal maps or color images. Specifically, we perform the optimization by randomly sampling a batch of pixels and their corresponding rays in world space $P = \{g_k, h_k, m_k, \mathbf{v}_k\}$, where $g_k$ is normal value of the $k_{th}$ sampled pixel, $h_k$ is color value of the $k_{th}$ pixel, $m_k \in \{0, 1\}$ is mask value of the $k_{th}$ pixel, and $\mathbf{v}_k$ is the direction of the corresponding sampled $k_{th}$ ray, from all views at each iteration.

The overall objective function is defined as

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{normal} + \mathcal{L}_{rgb} + \mathcal{L}_{mask} \\ + \mathcal{R}_{eik} + \mathcal{R}_{sparse} + \mathcal{R}_{smooth}\end{aligned} \tag{1}$$

where $\mathcal{L}_{normal}$ denotes the normal loss term that has been discussed in the main manuscript. $\mathcal{L}_{rgb}$ denotes a MSE loss term that calculates the errors between rendered colors $\hat{h}_k$ and generated colors $h_k$:

$$\mathcal{L}_{rgb} = \sum_k \left(h_k - \hat{h}_k\right)^2, \tag{2}$$

$\mathcal{L}_{mask}$ denotes a binary cross-entropy loss term that calculating errors between the rendered mask $\hat{m}_k$ and the generated mask $m_k$:

$$\mathcal{L}_{mask} = \sum_k \text{BCE}\left(m_k, \hat{m}_k\right), \tag{3}$$

$\mathcal{R}_{eik}$ denotes eikonal regularization term that encourages the magnitude of the SDF gradients to be unit length, $p$ is a 3D position in the world space, $f_\theta$ is the signed distance function, and $\nabla$ denotes the second-order gradient operator :

$$\mathcal{R}_{eik} = \sum_p \left(|\nabla f_\theta(p)| - 1\right)^2 \tag{4}$$

$\mathcal{R}_{sparse}$ denotes a sparsity regularization term that avoids floaters of SDF, and $\tau$ is a parameter to rescale the SDF values:

$$\mathcal{R}_{sparse} = \sum_p \exp(-\tau \cdot f_\theta(p)) \tag{5}$$

$\mathcal{R}_{smooth}$ denotes a 3D smoothness regularization term that enforces the SDF gradients to be smooth in 3D space, and $\Delta p$ is a small random shift relative to the 3D position $p$:

$$\mathcal{R}_{smooth} = \sum_p \left(|\nabla f_\theta(p)| - |\nabla f_\theta(p + \Delta p)|\right)^2 \tag{6}$$

## 2. More Results.

### 2.1. Novel View Synthesis

To further evaluate the generalization and robust performance of Wonder3D, we conduct visual comparisons with

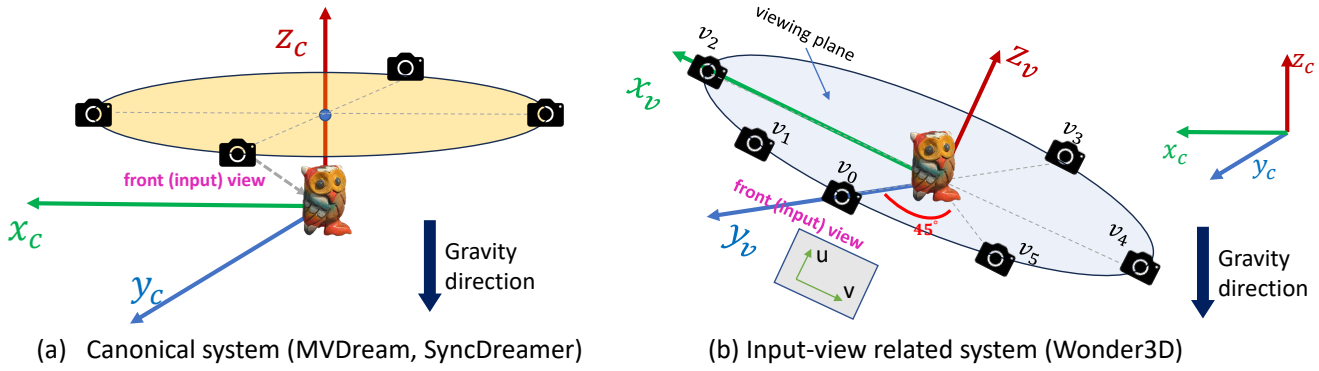


Figure 1. Illustration of the coordinate systems and camera poses.

the most recent work Zero123++ [2] and SyncDreamer [1] on the images with various styles, as shown in Fig. 7. As you can see, our method keeps correct geometry and robust performance on images with varying styles.

### 2.2. Single-view Reconstruction

We present more reconstruction results in Figure 3 and Figure 5. The readers may refer to supplementary videos for 360° visualization.

## 3. More Discussions.

### 3.1. Geometric Representation.

In addition to normal maps, various 2D representations encode geometric details, such as depth maps. However, Wonder3D opts for normals as the geometry representation instead of depth, guided by two considerations: 1) **Global Consistency**: Normal maps, defined in the same system, maintain global consistency across different viewpoints. The normal values for a 3D point in the world system remain consistent in any view. In contrast, the depth values for the same 3D point in the world system vary across different viewpoints. 2) **Scale-Invariance**: Inferring depth from a single image poses an ill-posed problem due to scale ambiguity. The size of an object in captured images is influenced by both the location of the capturing camera and the true shape scale, introducing ambiguity. On the contrary, normals remain scale-invariant, avoiding such ambiguity.

To compare the two geometry representations, we additionally train Wonder3D using depth instead of normals. The accompanying Figure 8 illustrates the reconstruction results using depth representation or normal representation. It is evident that due to the generated multi-view depth having less accurate multi-view consistency, the reconstruction results are noisier and less accurate.

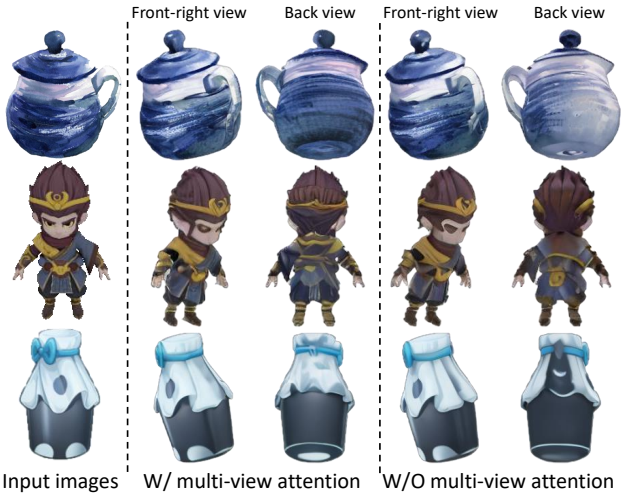


Figure 2. Ablation study on multi-view attention.

### 3.2. Multi-view Consistency.

We conducted an analysis of the effectiveness of the multi-view attention mechanism, as illustrated in Figure 2. Our findings show that the multi-view attention greatly enhances the 3D consistency of the generated multi-view images, particularly for the rear views. In the absence of the multi-view attention, the color images of the rear views exhibited unrealistic predictions.

### 3.3. Generalization.

To demonstrate the generalization capability of our method, we conducted evaluations using diverse image styles, including sketches, cartoons, and images of animals, as shown in Figure 5 and Figure 3. Despite variations in lighting effects and geometric complexities among these images, our method consistently generated multi-view normal maps and color images, ultimately yielding high-quality geometries.

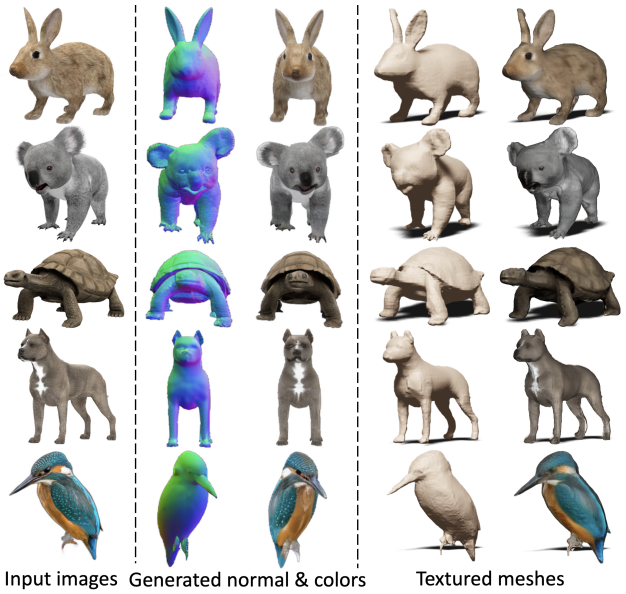


Figure 3. The qualitative results of *Wonder3D* on various animal objects.

Figure 4. Combine Wonder3D and Houdini to create lego-style objects.

## 4. Potential Applications

### 4.1. 3D Printing.

Our method *Wonder3D* presents strong generalization ability, making it a powerful tool for the application of 3D printing. As shown in Figure 6, *Wonder3D* can faithfully lift the 2D images into real 3D objects via 3D printing machines.

### 4.2. Creating Lego-style objects.

The high-quality textured meshes generated by *Wonder3D* can be further processed into the lego-style objects via Houdini. The lego-style objects prove the potentials that *Wonder3D* can be used as customized 3D assets creation tool in the open-world game Minecraft.

## References

[1] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2

[2] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model, 2023. 2

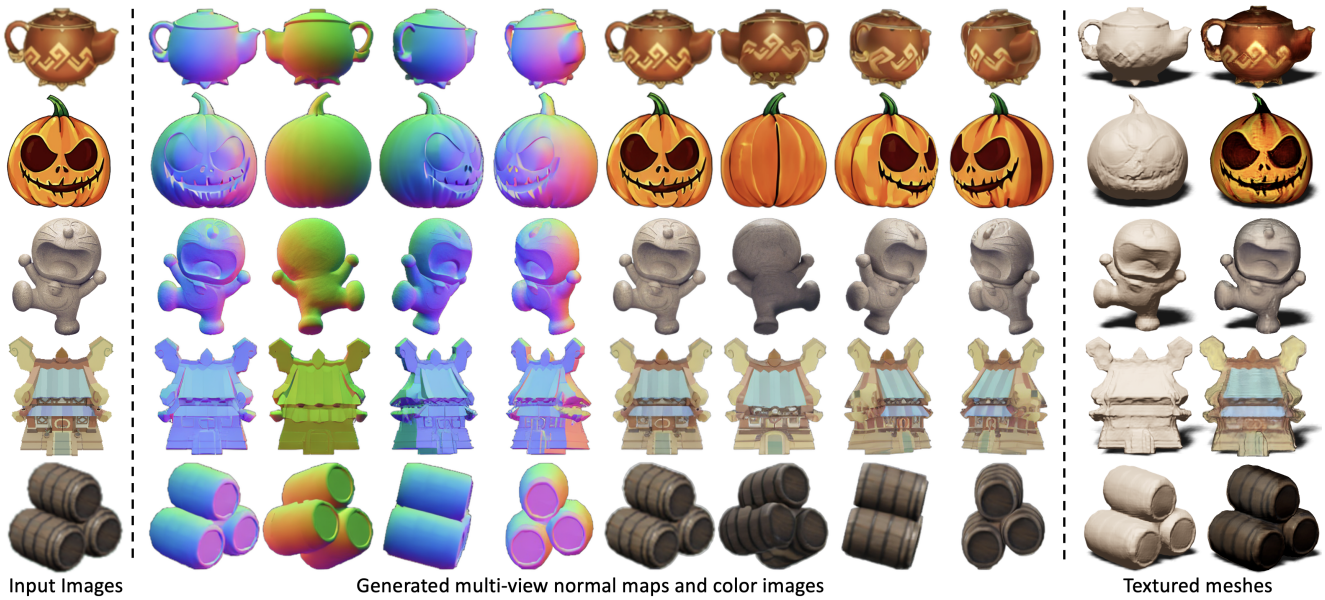


Figure 5. The qualitative results of *Wonder3D* on various styles of images.


Figure 6. The 3D printing results of Wonder3D.

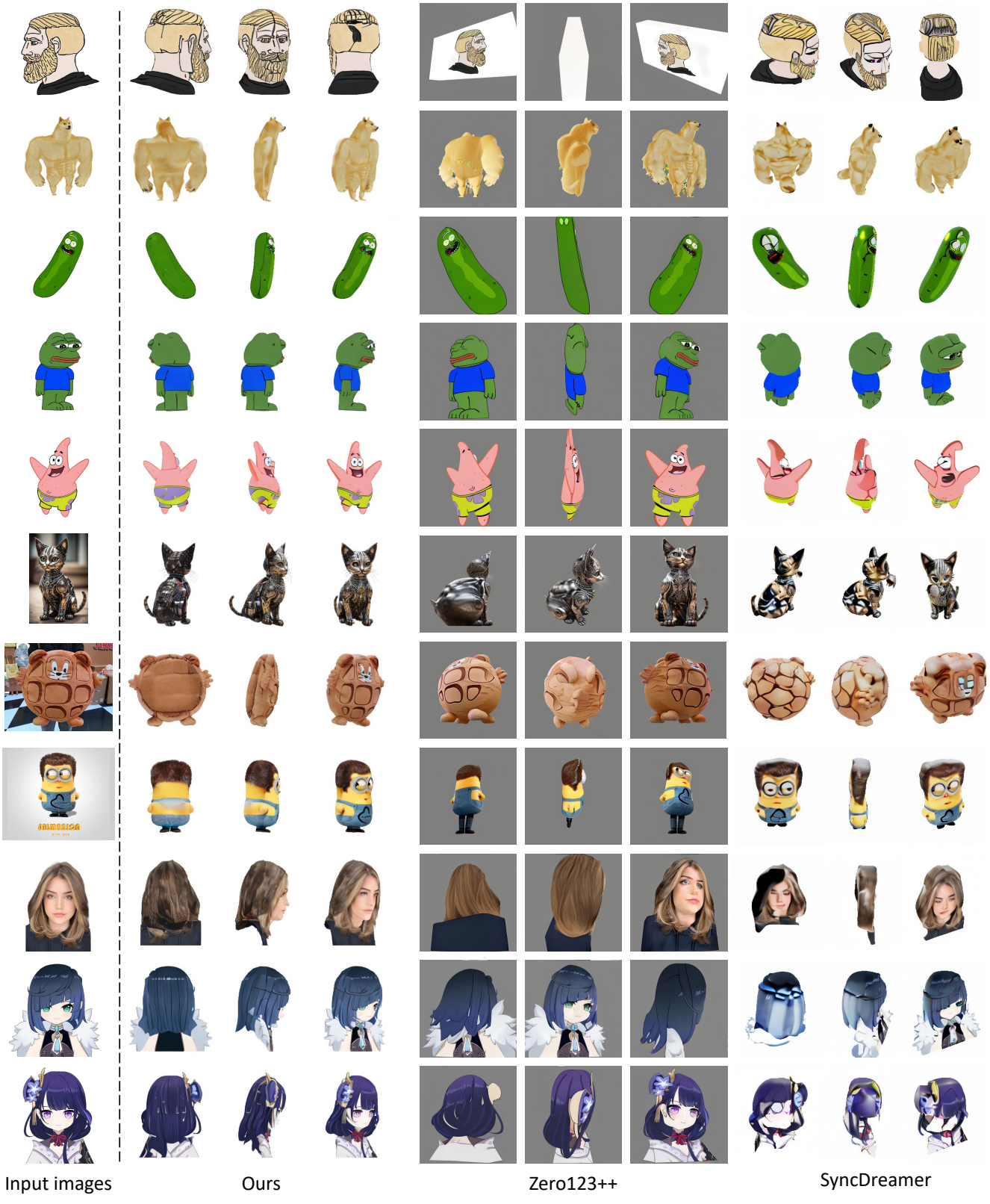


Figure 7. The qualitative comparisons with Zero123++ and SyncDreamer.

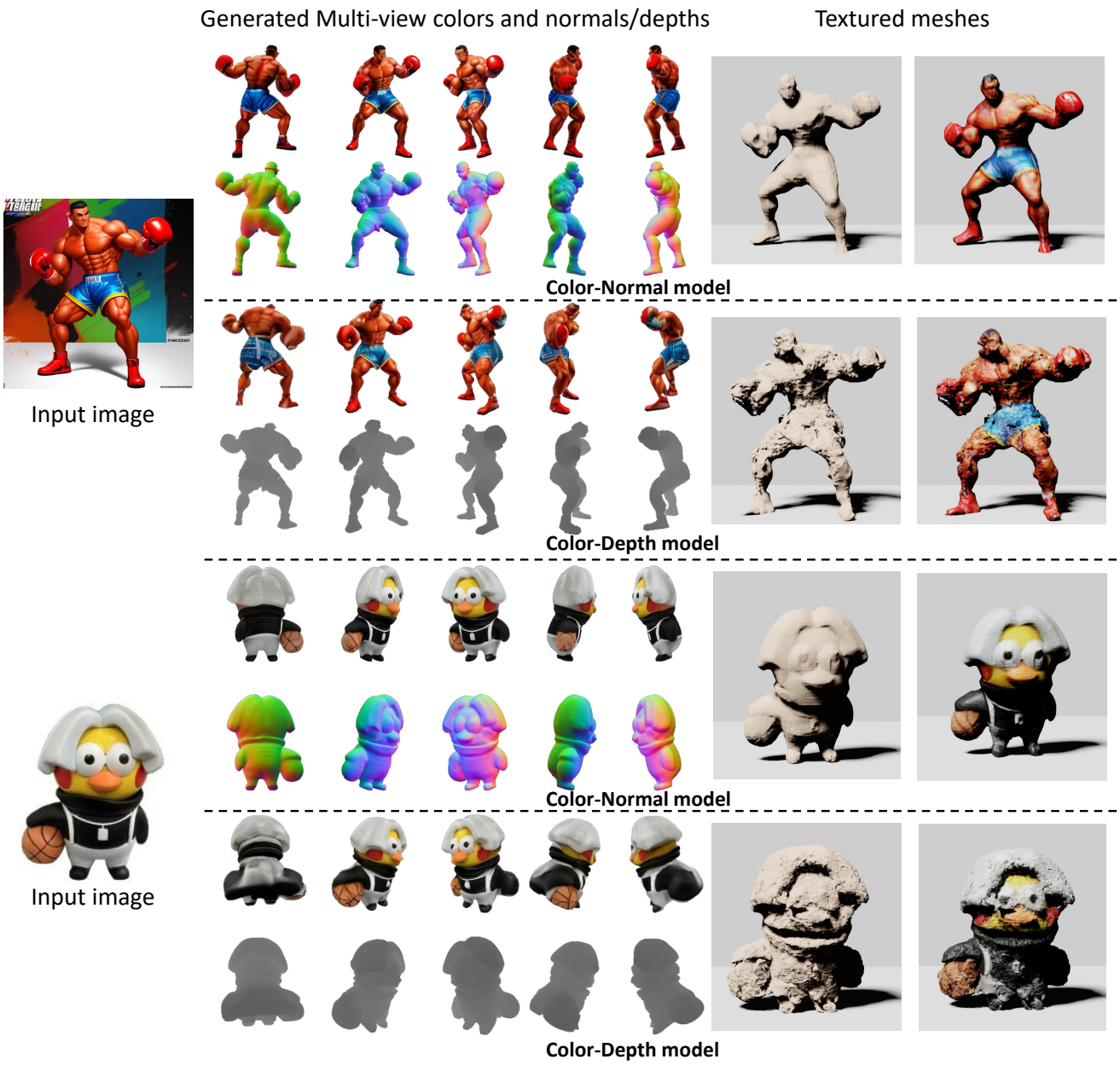


Figure 8. The results of Color-Normal model and Color-Depth m.