3D-aware Image Generation using 2D Diffusion Models (*Supplementary Material*)

Table I: Network and training details

Network	#Params	Steps	Training time
ImageNet $\mathcal{G}_u$	422M	1M	$\sim$ 11.1 Days
ImageNet $\mathcal{G}_c$	422M	500K	$\sim$ 5.73 Days
ImageNet $\mathcal{G}_{sr}$	105M	200K	$\sim$ 0.54 Days
SDIP Dogs $\mathcal{G}_u$	105M	500K	$\sim$ 1.29 Days
SDIP Dogs $\mathcal{G}_c$	105M	300K	$\sim$ 0.82 Days
SDIP Dogs $\mathcal{G}_{sr}$	105M	200K	$\sim$ 0.54 Days
SDIP Elephants $\mathcal{G}_u$	105M	300K	$\sim$ 0.77 Days
SDIP Elephants $\mathcal{G}_c$	105M	300K	$\sim$ 0.82 Days
SDIP Elephants $\mathcal{G}_{sr}$	105M	200K	$\sim$ 0.54 Days
LSUN Horses $\mathcal{G}_u$	105M	300K	$\sim$ 0.77 Days
LSUN Horses $\mathcal{G}_c$	105M	300K	$\sim$ 0.82 Days
LSUN Horses $\mathcal{G}_{sr}$	105M	200K	$\sim$ 0.54 Days

A. More Training Details

The network architecture of our method is adopted from ADM [2] with necessary modifications to support our task. Specifically, for the unconditional diffusion model $\mathcal{G}_u$, the first convolution layer's input channel and the last convolution layer's output channel are enlarged from 3 to 4 to incorporate the depth map. For the conditional diffusion model $\mathcal{G}_c$, the fist layer is enlarged to 10 channels to incorporate additional conditions (4 channels for the noisy RGBD image, 3 for the warped texture, 1 for the warped depth, 1 for the texture mask, and 1 for the depth). For super-resolution model $\mathcal{G}_{sr}$, the fist layer is enlarged to 8 channels to include the low-resolution image as condition. The parameters added for injecting the conditions are initialized as zero to remove the impact of conditions at the beginning of conditional model fine-tuning.

All our models are trained with FP16 precision using 8 NVIDIA Tesla V100 GPU with 32GB memory FP16 training. We use the Adam [3] optimizer with a learning rate of $1e-4$ and β set to $(0.9, 0.999)$ for all the datasets. The batchsize is set to 64. Exponential Moving Average (EMA) is enabled with smoothing rate of 0.9999 to boost the performance. More information about the networks and training process can be found in Table I.

B. More Implementation Details

B.1. Condition Aggregation Details

In Sec. 4.3 of the main paper, we introduced our *aggregated conditioning* strategy and here we present more details regarding aggregation weight computation.

To sample from $q_i(\Gamma(\mathbf{x}, \boldsymbol{\pi}_n)|\Pi(\Gamma(\mathbf{x}, \boldsymbol{\pi}_0), \boldsymbol{\pi}_n), \cdots)$, our condition aggregation collects information from previous images by performing a weighted sum across all warped versions of them:

$$\mathbf{C}_n = \sum_{i=0}^{n-1} \mathbf{W}_{(i,n)} \Pi(\mathbf{I}_i, \boldsymbol{\pi}_n) \Big/ \sum_{i=0}^{n-1} \mathbf{W}_{(i,n)}, \qquad \text{(I)}$$

where $\mathbf{W}_{(i,n)}$ is the weight map. The weight is calculated for each pixel (x, y). Let $\mathbf{p}^{x,y}$ and $\mathbf{n}^{x,y}$ be the pixel's spatial position and normal in the world coordinate space and $d^{x,y}$ be its distance to the nearest masked pixel on image plane, we set

$$\mathbf{W}^{x,y}_{(i,n)} = \phi\Big(\frac{\boldsymbol{o}_i - \boldsymbol{p}^{x,y}}{\|\boldsymbol{o}_i - \boldsymbol{p}^{x,y}\|} \cdot \mathbf{n}^{x,y}\Big)\psi(d^{x,y}), \qquad \text{(II)}$$

where $\boldsymbol{o}_i$ is the camera center of the i-th view. $\phi(\cdot)$ and $\psi(\cdot)$ are scalar functions for balancing the weights of the two terms, which are empirically set as $\phi(a) = \exp(-20 * \arccos(a))$ and $\psi(b) = b$. The first term assign each pixel the least distorted information [1] and the second term suppresses the contribution of pixels near occlusion boundaries.

C. More Experimental Results

C.1. More Visual Results

In Fig. I and Fig. II, we show more multiview results of 128^2 resolution for each of the four datasets we tested. More uncurated results with camera pose randomly drawn from Gaussian distribution ($\sigma = 0.3$ for yaw and 0.15 for pitch) are presented in Fig. III and Fig. IV.

C.2. More 360° Results

In Fig. V, we show more cases where our method successfully generates the results under a 360° camera trajectory.

C.3. More 256^2 Results

In Fig. VI, we show more 256^2 multiview results generated with the diffusion-based image upsampler.

C.4. Shape Extraction

To extact the 3D shape of a generated instance, we employ *tsdf-fusion* [5] to fuse the generated multiview depth maps into a voxel grid and obtain the surface mesh using *marching cubes* [4]. Some examples are visualized in Fig. VII.

C.5. Failure Cases

In Fig. VIII, we demonstrate some typical failure cases of our method. First, our unconditional generation model $\mathcal{G}_u$ sometimes failed to model complex structures, which can be attributed to both limited model capacity and limited data for some object categories in ImageNet. Second, our model may generate severely-mismatched color and depth maps along occlusion boundaries which cannot be handled our texture erosion strategy. Under such situations, the conditional model will fail to generate proper contents under novel views. For the 360° generation, the sequence may not converge and the results can be completely out of domain.

D. Discussions

D.1. Discussion about concurrent work

3DGP [6] is a concurrent work to ours. Their code and models are released after our submission (released on *May 6th*). Here, we add the results comparison with their method. Quantitatively speaking, our method achieves better generation quality as their FID-10K on ImageNet-128 is **20.6** while ours is **9.45**. Besides, our method does not suffer their "background sticking and no 360° generation", "flat geometry", and "GAN mode collapse" issues (see text in their webpage).

D.2. More discussion of limitations

Our method suffers from degraded image quality for large views, as mentioned in Sec. 5.3 of the main paper. There are at least two causes for this issue: domain drift and data bias. The errors on the generated depth will lead to distortions in the warped novel views, which will be accumulated and amplified in the iterative view sampling process and hence gradually drive the sample away from the real image distribution. Poor results will be generated when severe domain drift happens. Additionally, the object poses in the training datasets are usually not distributed uniformly. Side and rear views are often much less than frontal ones, rendering side- and rear-view image generation difficult.

Also, the quality of the back-views when generating under 360° is sometimes unsatisfactory (*e.g.*, copy of the frontal view). Methods trained on single images may have the ambiguity for certain objects with symmetry. Adding view conditioning may help alleviating this issue. Note that in our case, lacking back-view training images is also an important reason for the inferior back-views.

References

[1] Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler, and Michael Cohen. Unstructured lumigraph rendering. In *Annual Conference on Computer Graphics and Interactive Techniques*, pages 425–432, 2001. 1

[2] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. 1

[4] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM SIGGRAPH*, 21(4):163–169, 1987. 2

[5] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136. Ieee, 2011. 2

[6] Ivan Skorokhodov, Aliaksandr Siarohin, Yinghao Xu, Jian Ren, Hsin-Ying Lee, Peter Wonka, and Sergey Tulyakov. 3d generation on imagenet. In *International Conference on Learning Representations*, 2023. 2

Figure I: More 128^2 multiview results on ImageNet.

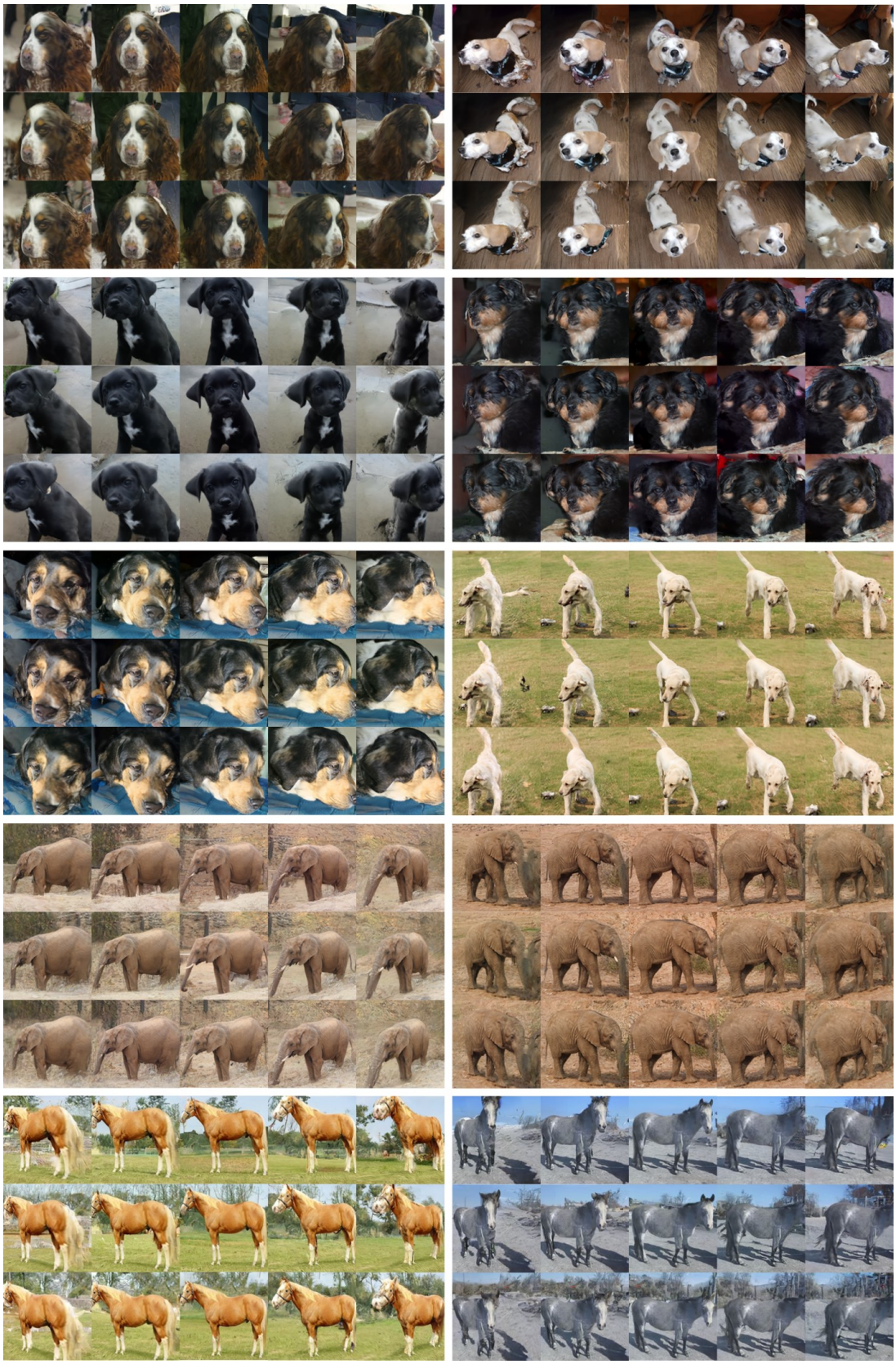

Figure II: More 128^2 multiview results for SDIP Dogs, SDIP Elephants, and LSUN Horses.

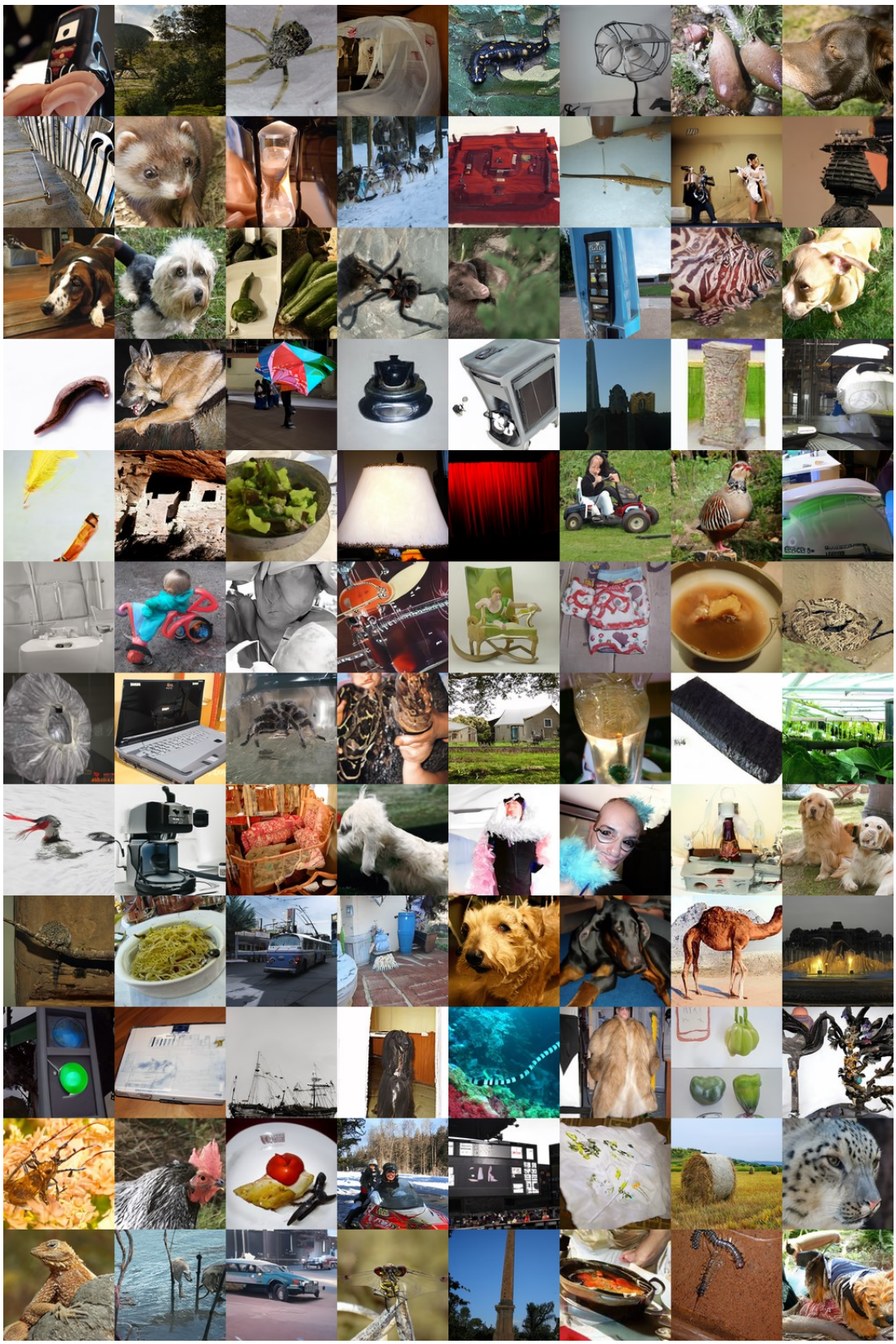

Figure III: More uncurated 128^2 single-view results for ImageNet. Note that the views are randomly sampled.

Figure IV: More uncurated 128^2 single-view results for SDIP Dogs, SDIP Elephants, and LSUN Horses. Note that the views are randomly sampled.

Figure V: More 360° generation results on ImageNet.

Figure VI: More 256^2 generation results.

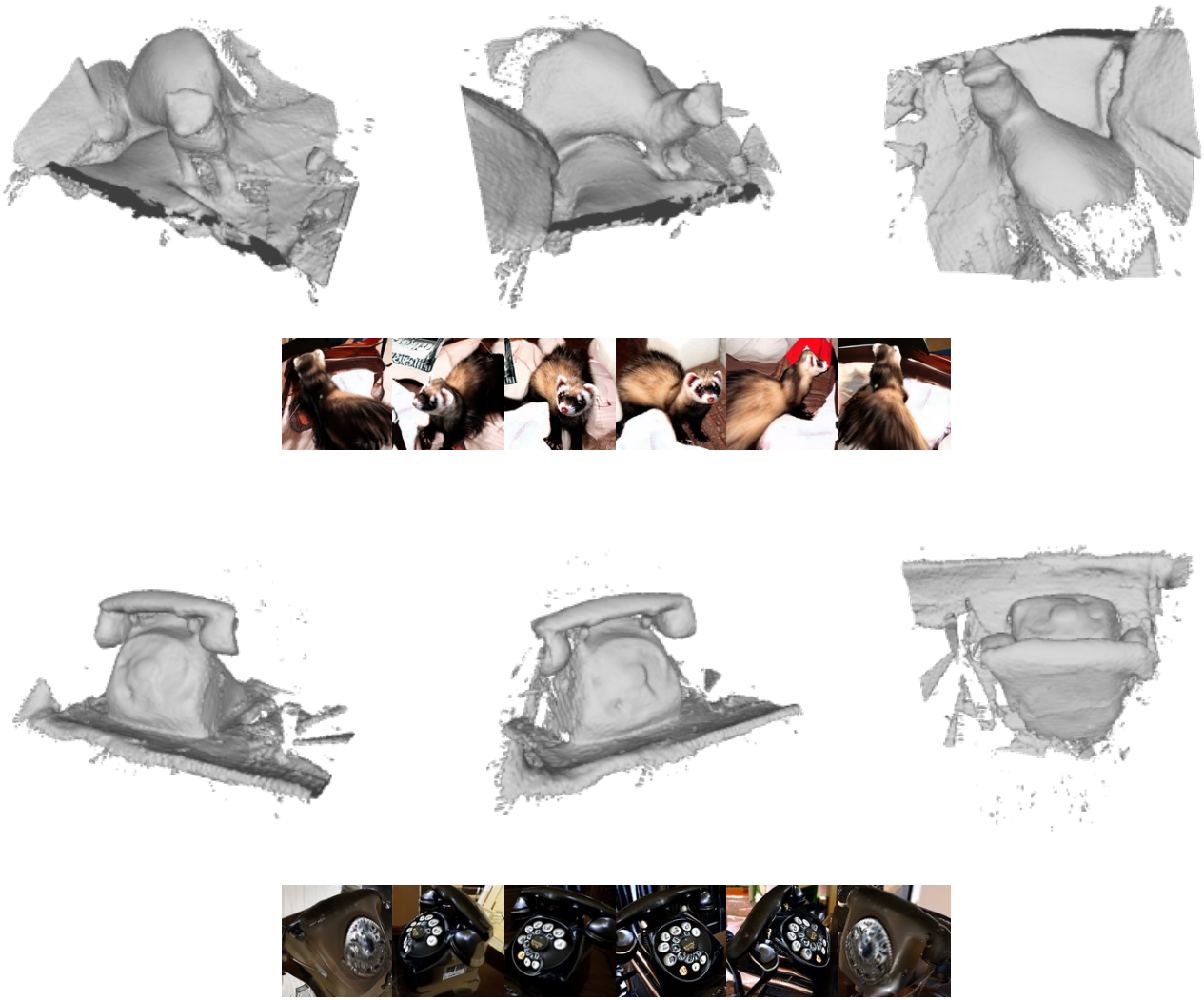

Figure VII: Shapes extracted using tsdf fusion and marching cubes.

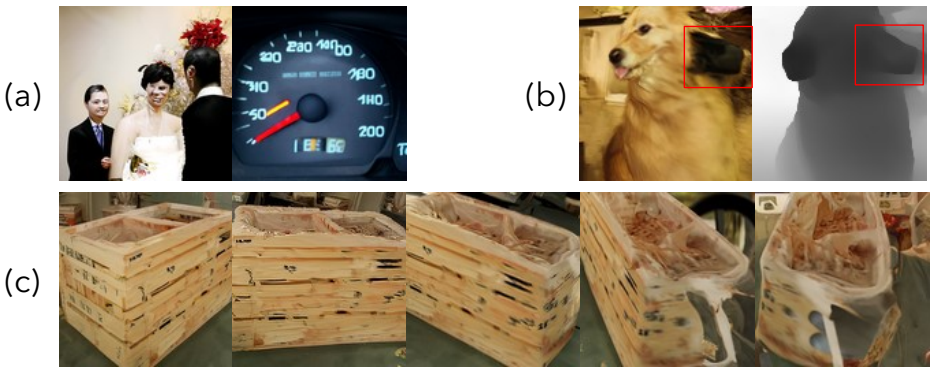

Figure VIII: Failure cases. **(a)** Some structures that are not well modeled by the network; **(b)** Severe mismatch between the generated color and depth map along occlusion boundary leads to poor novel view generation results. **(c)** Samples can be out of domain for very large views.