

A ARCHITECTURE DETAILS

We provide comprehensive information on the architectures of the networks employed. In Appendix A.1, we elaborate on the text-based video generation network (ModelScopeT2V Wang et al. (2023a)), including its transformation into a text-based multi-view image generator. In Appendix A.2, we discuss the image-based video generation network (I2VGen-XL Zhang et al. (2023)) and its conversion into an image-based multi-view image generator. Finally, in Appendix A.3, we present our large GaussianSplatting-based reconstruction model (LGM Tang et al. (2024)) and how it is utilized for noise reconstruction fine-tuning.

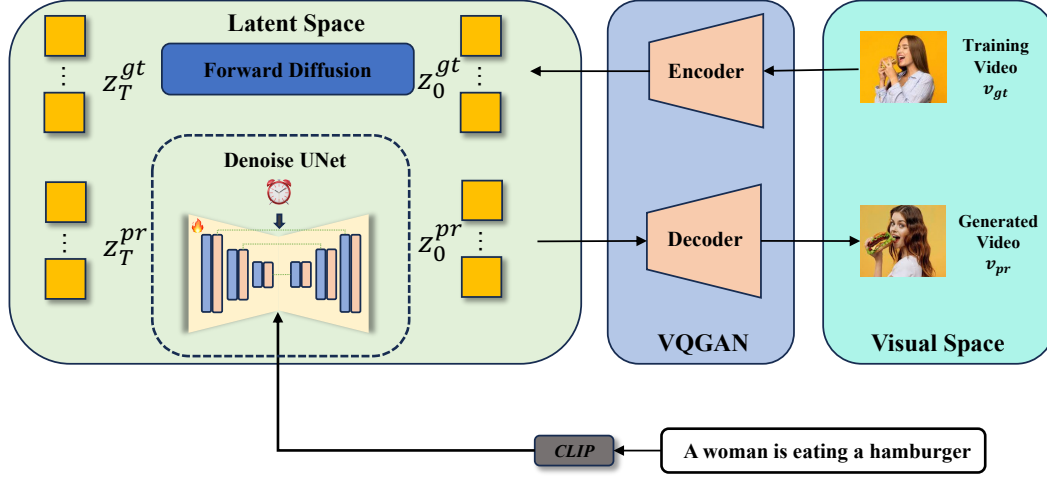


Figure 8: Overview of the architecture of ModelScopeT2V

A.1 MODELSCOPE T2V (TEXT-BASED VIDEO GENERATION)

The main paradigm of ModelScopeT2V is shown in Fig. 8. Overall, it comprises two main components: VQGAN and Denoise UNet. VQGAN aims to reconstruct the original video as precisely as possible. Given a training video v_{gt} , the encoder of VQGAN compresses it into the latent space as z_0^{gt} . Then, a quantizer is applied to z_0^{gt} . Note that the function of the quantizer is to find the closest quantized discrete vector of z_0^{gt} , and we denote the output from the quantizer as q_0^{gt} . Finally, the decoder of VQGAN takes q_0^{gt} as input and outputs the reconstruction v_0^{rec} . The overall objective of VQGAN is the weighted summation of the quantization loss between $L_q = f_q(z_0^{gt}, q_0^{gt})$, the reconstruction loss $L_r = f_r(v_0^{gt}, v_0^{rec})$, and an optional adversarial loss $L_{adv} = f_{adv}(v_0^{gt}, v_0^{rec})$.

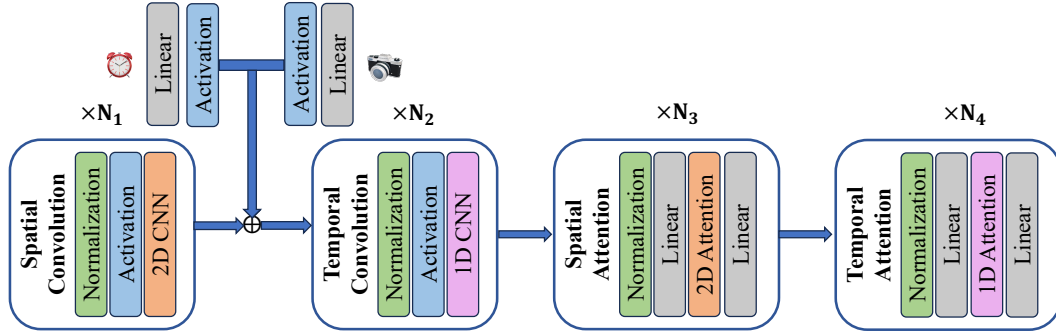


Figure 9: Building blocks of the Denoise UNet.

DenoiseUNet aims to recover z_0^{gt} from the noise-corrupted z_T^{gt} . The optimization objective is the denoise reconstruction loss $L_{dr} = f_{dr}(z_0^{gt}, z_0^{pr})$. z_0^{pr} is predicted by the network mapping function ϵ_θ as $z_0^{pr} = \epsilon_\theta(z_T^{gt}, y, t)$. As depicted in Fig. 9, DenoiseUNet is built with several Spatial-Temporal

convolutional and attention modules. In the original implementation of ModelScopeT2V Wang et al. (2023a), timesteps t are injected into the spatial convolution modules as residuals. To convert ModelScopeT2V Wang et al. (2023a) into a text-based multi-view generator, we pass camera poses through an activation layer and a linear layer. The output of these layers has the same feature dimension as the embeddings of timesteps, and we further add them together. Afterward, the added embeddings of timesteps and camera poses are projected to specific feature dimensions and added together with the output of the Spatial Convolution module. Another modification is the input data. The original input of ModelScopeT2V accepts a video with 32 frames, while we modify the frame number to 24 for multi-view image generation.

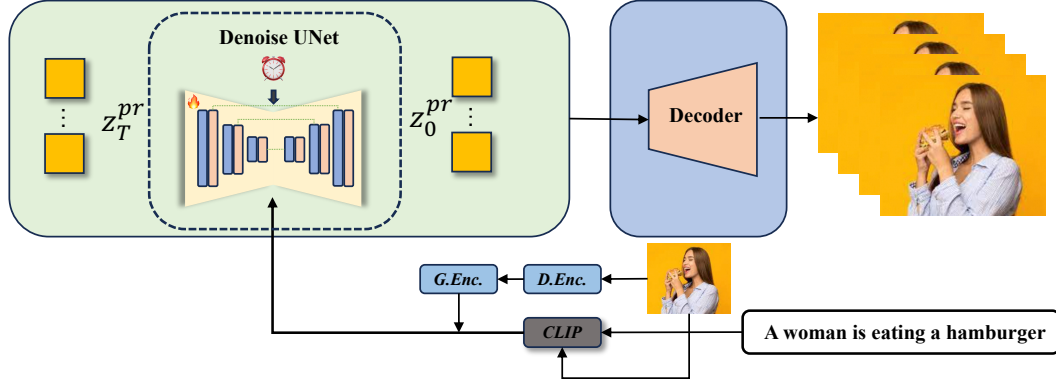


Figure 10: Overview of the architecture of I2VGen-XL.

A.2 I2VGEN-XL (IMAGE-BASED VIDEO GENERATION)

The primary paradigm of I2VGen-XL is illustrated in Fig. 10. It is important to note that the framework follows a one-staged approach, distinguishing it from the two-staged architecture proposed by Zhang et al. (2023). Upon careful examination, we discovered that the available open-source implementation actually corresponds to a one-staged model; hence, we decided to adopt this version as depicted in Fig. 10. This one-staged model allows for the incorporation of both images and text as conditions, providing additional global (*G.Enc.*) and detailed (*D.Enc.*) information extracted from the image.

I2VGen-XL Zhang et al. (2023) shares the same DenoiseUNet architecture and VQGAN architecture with ModelScopeT2V Wang et al. (2023a). However, they differ in terms of dataset utilization and condition injection methodology. To convert I2VGen-XL Zhang et al. (2023) into a multi-view image generator, we designated the input prompt as an empty string. For instance, we replaced 'A woman is eating a hamburger' with an empty prompt. Furthermore, we adjusted the frame number to 24 and set the training video resolution to 256×256 when fine-tuning a multi-view image generator based on I2VGen-XL Zhang et al. (2023). We observed that augmenting the training dataset size yields improvements in terms of generalizability. Specifically, our implemented image-based multi-view generation model was trained on a dataset comprising approximately 170K samples, carefully curated by excluding textureless instances from G-Objaverse. It is important to note that this dataset differs from the high-quality 28K text-video pairs utilized for fine-tuning our text-to-video model.

A.3 FEED-FORWARD RECONSTRUCTION

We have adopted the identical architecture proposed in LGM Tang et al. (2024). As illustrated in Fig. 11, this asymmetric UNet architecture offers advantages in terms of memory efficiency by mitigating the increase in points within the GaussianSplatting representation caused by high-resolution output. It incorporates dense self-attention, similar to MVDream Shi et al. (2023b). Considering computing resources, we did not extend it to accommodate 24 views. Yet, we have future plans for developing a dense view reconstruction model.

Sequentially, we fine-tune the asymmetric UNet using the "predicted x_0 ". Inspired by LGM Tang et al. (2024), we randomly select four views from the output of VideoMV as inputs for FFR, resulting in a Gaussian field. One notable distinction is that we incorporate supervision by rendering all 24

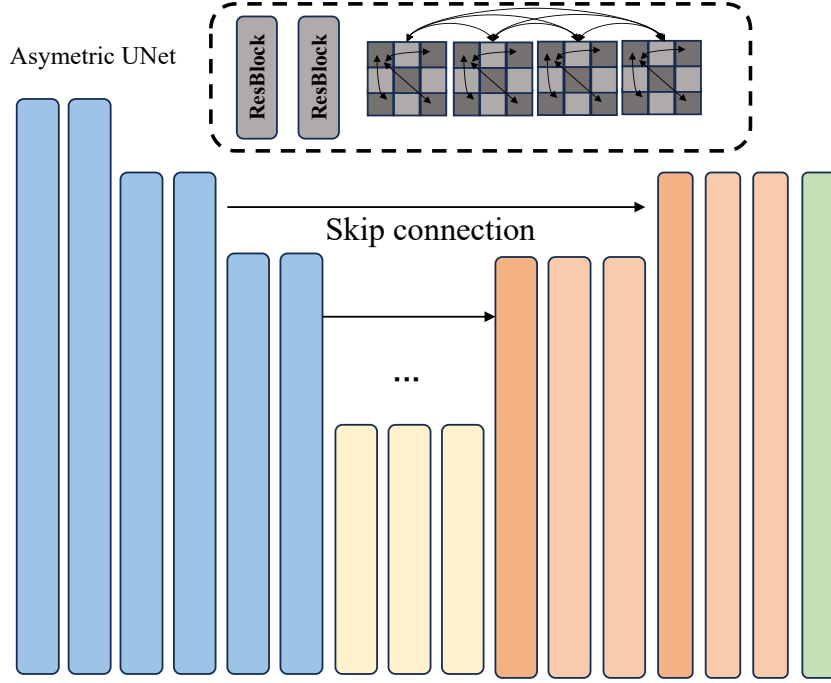


Figure 11: Architecture of the Asymmetric UNet.

views, deviating from the original implementation which utilized only 8 views. Additionally, we use a background color of $[128, 128, 128]$ instead of $[255, 255, 255]$ for Text-to-Multi-view image generation. On the other hand, for Image-to-Multi-view image generation, we still model the background as pure white. During the inference stage, we use fixed 4 orthogonal views for stable reconstruction performance, similar to LGM Tang et al. (2024). For further details on the configuration of this sparse view reconstruction pipeline, we suggest referring to the original implementation Tang et al. (2024).

B APPLICATIONS

B.1 DENSE VIEW RECONSTRUCTION

Our proposal can generate 24 dense views with specified camera poses using DDIM Song et al. (2020) sampling in 5 seconds. The dense view generated is enough for a reconstruction pipeline such as NeRF Mildenhall et al. (2020) or Neus Wang et al. (2021). We show reconstruction results on both image-based multi-view generation and text-based multi-view generation tasks.

For the image-based multiview generation task, we visualize results from NVS-based methods (Zero123(XL) Liu et al. (2023b), SyncDreameer Liu et al. (2023c)) and reconstruction-focused methods (Wonder3D Long et al. (2023), OpenLRM Hong et al. (2023); He & Wang (2023), and Shap-E Jun & Nichol (2023)) accompanied with our approach. As shown in Fig. 6, VideoMV achieves better reconstruction results among NVS-based methods due to the highly consistent dense view produced by our pipeline. Wonder3D Long et al. (2023) produces shapes with more details for the use of predicted normal maps, but sometimes produce floating artifacts. Inference-based methods such as Shap-E and OpenLRM may produce shapes that are not well aligned with the input images. The multi-view images generated by VideoMV can provide competitive prior for 3D generation compared with image-based NVS methods. Although the Neus Wang et al. (2021) reconstruction is smooth and does not provide geometry details, we can adopt the cross-domain attention mechanism proposed in Wonder3D Long et al. (2023) to produce aligned normal maps and enhance the performance of our dense view reconstruction.

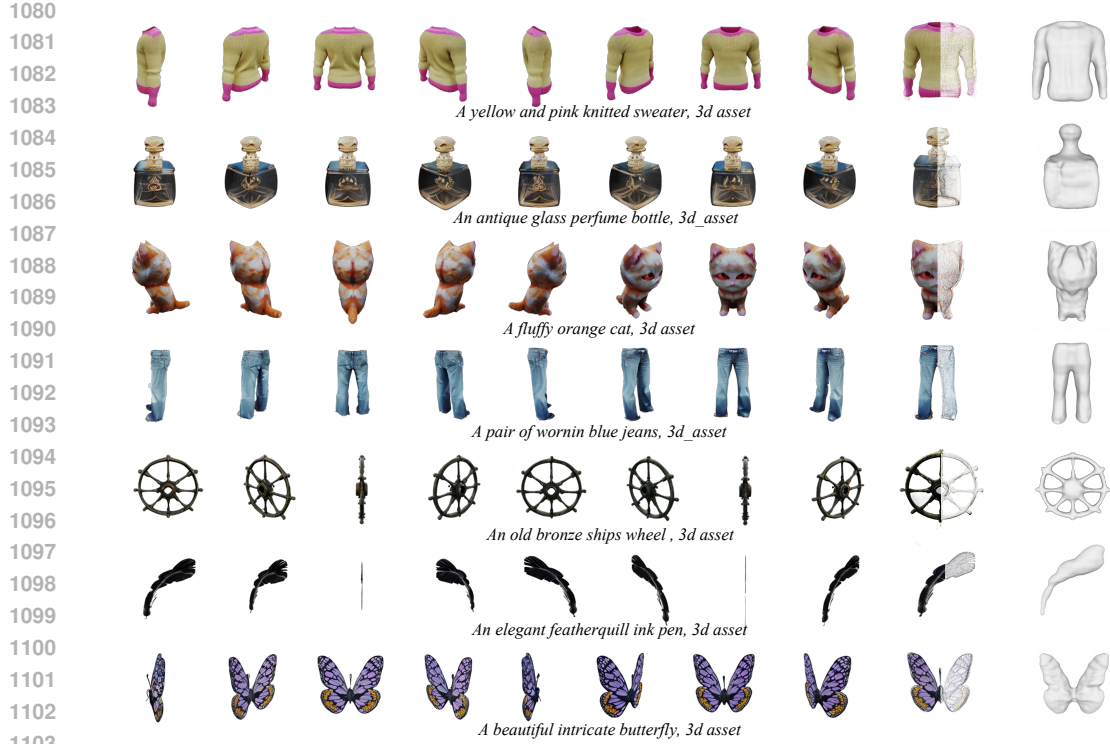


Figure 12: Text-based dense view reconstruction results of VideoMV.



Figure 13: Visualization of text-based multi-view score distillation.

For the text-based multi-view generation task, we visualize the results of VideoMV only. As depicted in Fig. 12, VideoMV can also recover geometry from multi-view images generated from text prompts by Neus Wang et al. (2021). As a by-product, we can also produce a Gaussian splatting field Kerbl et al. (2023a) from multi-view images in seconds.

B.2 DISTILLATION-BASED GENERATION

Our proposal can also be applied as the prior of score distillation sampling Poole et al. (2022). As shown in Fig. 13, we can distillate faithful shapes and texture from a multi-view score distillation loss and avoid the Janus problem most of the time. Note that we are focusing on consistent multi-view image generation, so we do not fully optimize the distillation pipeline. Distillation from dense views is also an interesting task for future work.

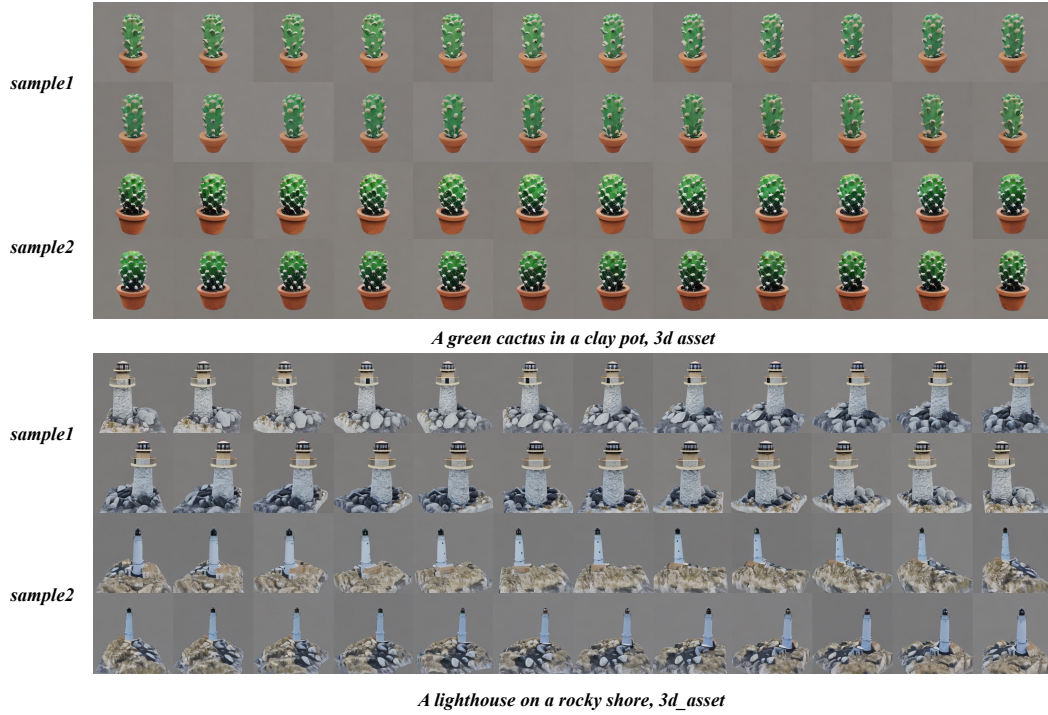


Figure 14: Visual various results of Text-to-Multi-View image generation on T3Bench (Part I).

C EXPERIMENT DETAILS

In our implementation of VideoMV, we maintain a learning rate of $3e^{-5}$ and a batch size of 32. We utilize the AdamW optimizer and employ FP16 for efficient gradient descent without weight decay. The training dataset comprises 28K samples for text-conditioned VideoMV and 170K samples for image-conditioned VideoMV, all sourced from G-Objaverse Qiu et al. (2023). For text-conditioned VideoMV, the training process converges within half an hour using 8 NVIDIA A100 GPUs. With further training, the performance slightly improves. Image-based VideoMV requires a total training time of 24 hours utilizing 8 NVIDIA A100 GPUs.

D TEXT-TO-MULTI-VIEW IMAGE GENERATION

The VideoMV demonstrates the ability to generate diverse outcomes by employing different random noises while maintaining the same prompt. As depicted from Fig. 14 to Fig. 18, VideoMV produces a range of astonishing results across various prompts selected from the multi-object list in T3Bench He et al. (2023).

D.1 MORE QUALITATIVE RESULTS

Despite the limited data used for fine-tuning large-scale video generative models, the alignment between prompts and visual information in both video datasets and image datasets (consisting of 1-frame videos) remarkably enhances generalizability to open-vocabulary scenarios, enabling VideoMV to generate multi-view images beyond the limitations of training datasets. To further enhance qualitative visualization, we adopt highly abstract prompts previously employed in Dream-Fusion Poole et al. (2022). As illustrated in Fig. 19 to Fig. 27, VideoMV consistently generates dense multi-view images based on abstract prompts, showing its ability to understand out-of-distribution data.

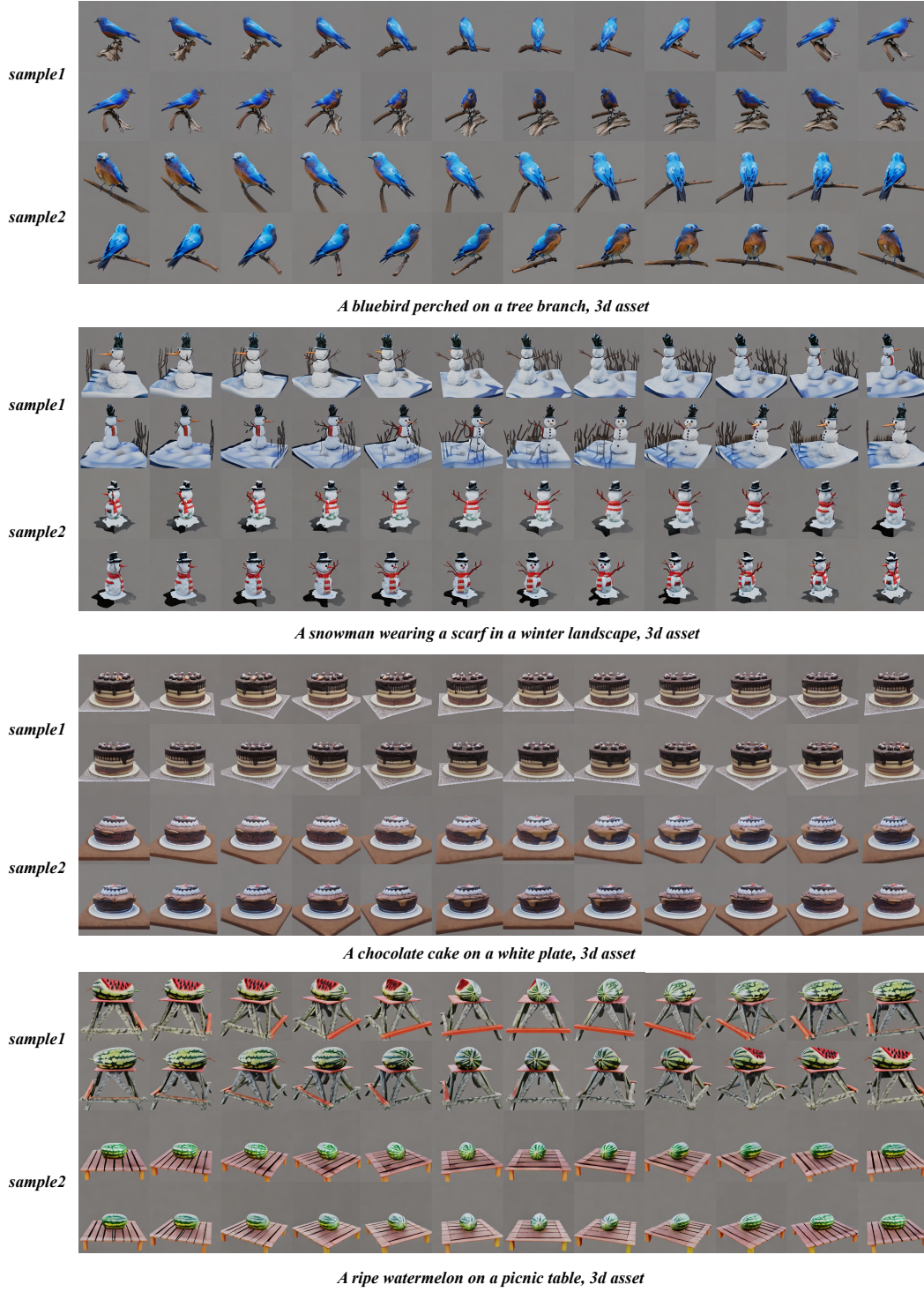
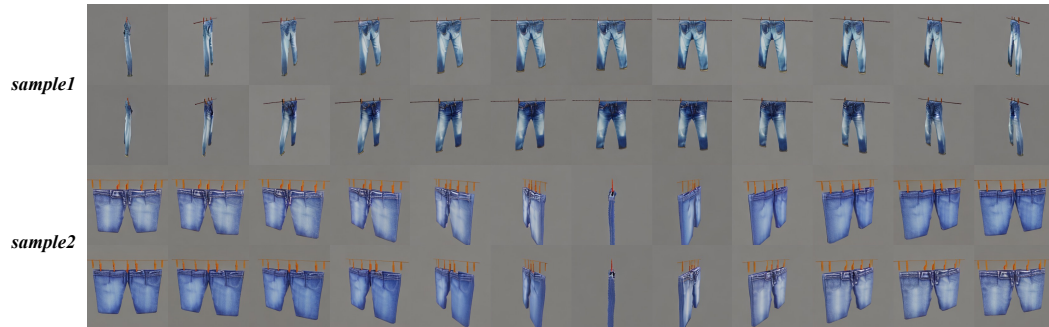
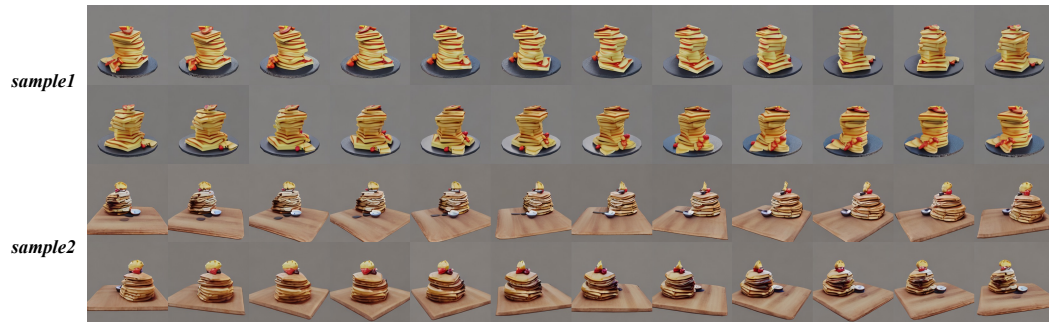


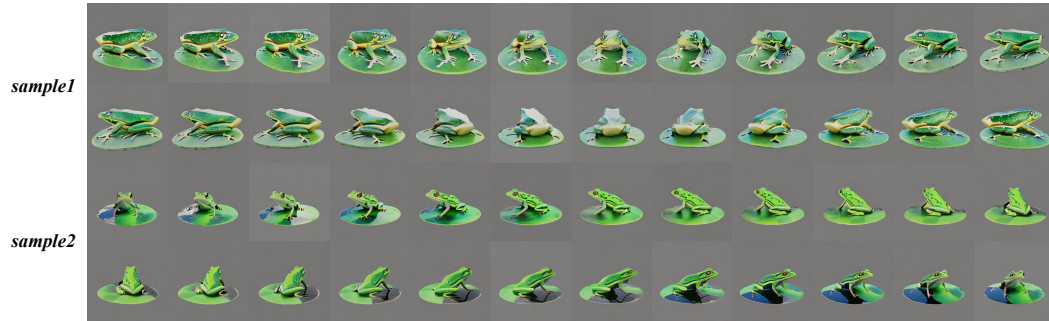
Figure 15: Visual various results of Text-to-Multi-View image generation on T3Bench (Part II).



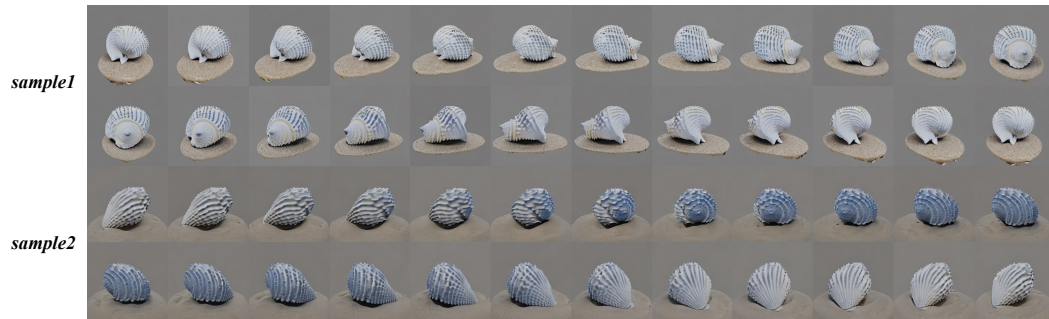
A pair of blue jeans hanging on a clothesline, 3d asset



A stack of pancakes on a breakfast table, 3d asset



A green frog on a lily pad, 3d asset



A white seashell on a sandy beach, 3d asset

Figure 16: Visual various results of Text-to-Multi-View image generation on T3Bench (Part III).

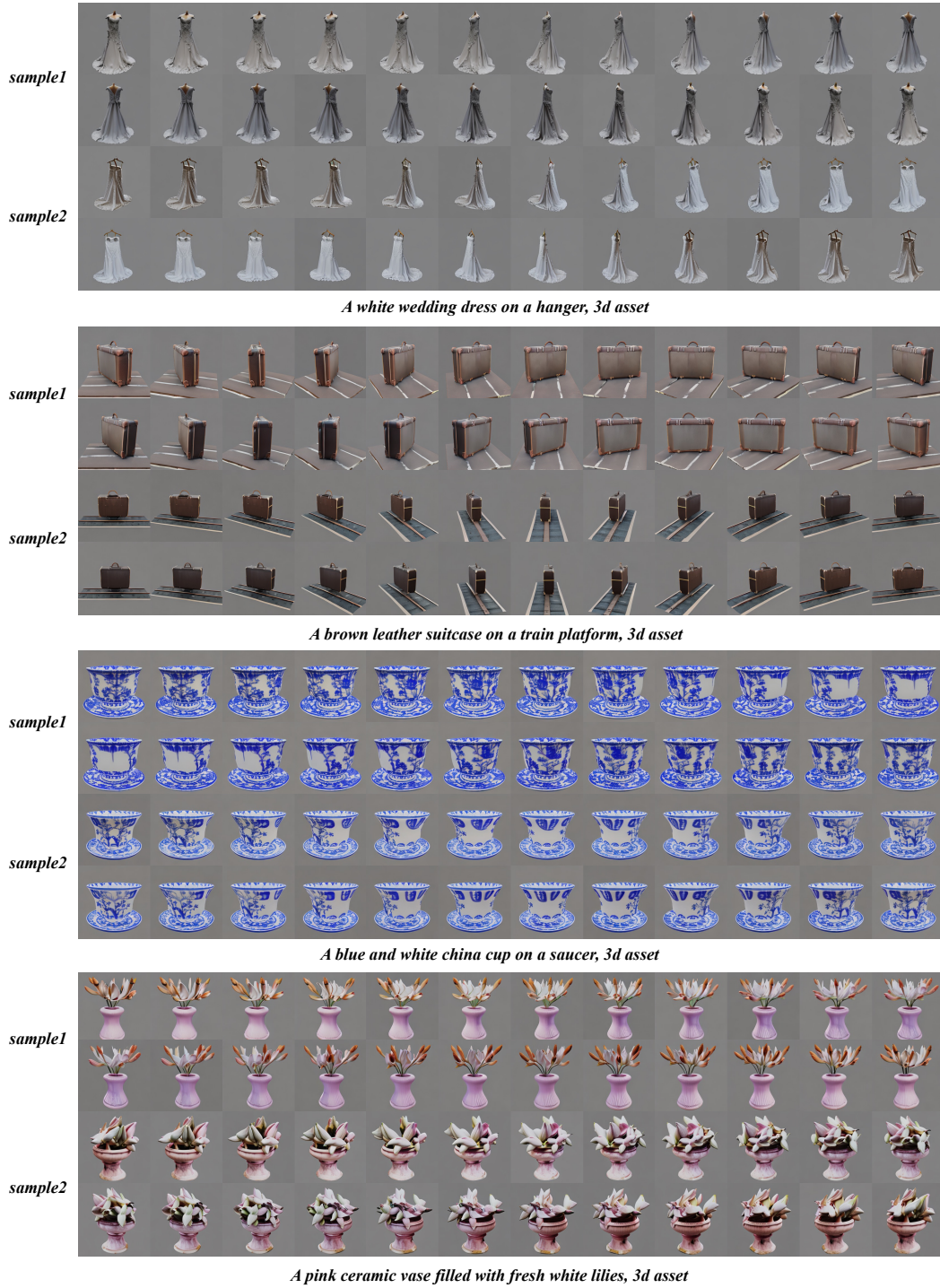


Figure 17: Visual various results of Text-to-Multi-View image generation on T3Bench (Part IV).

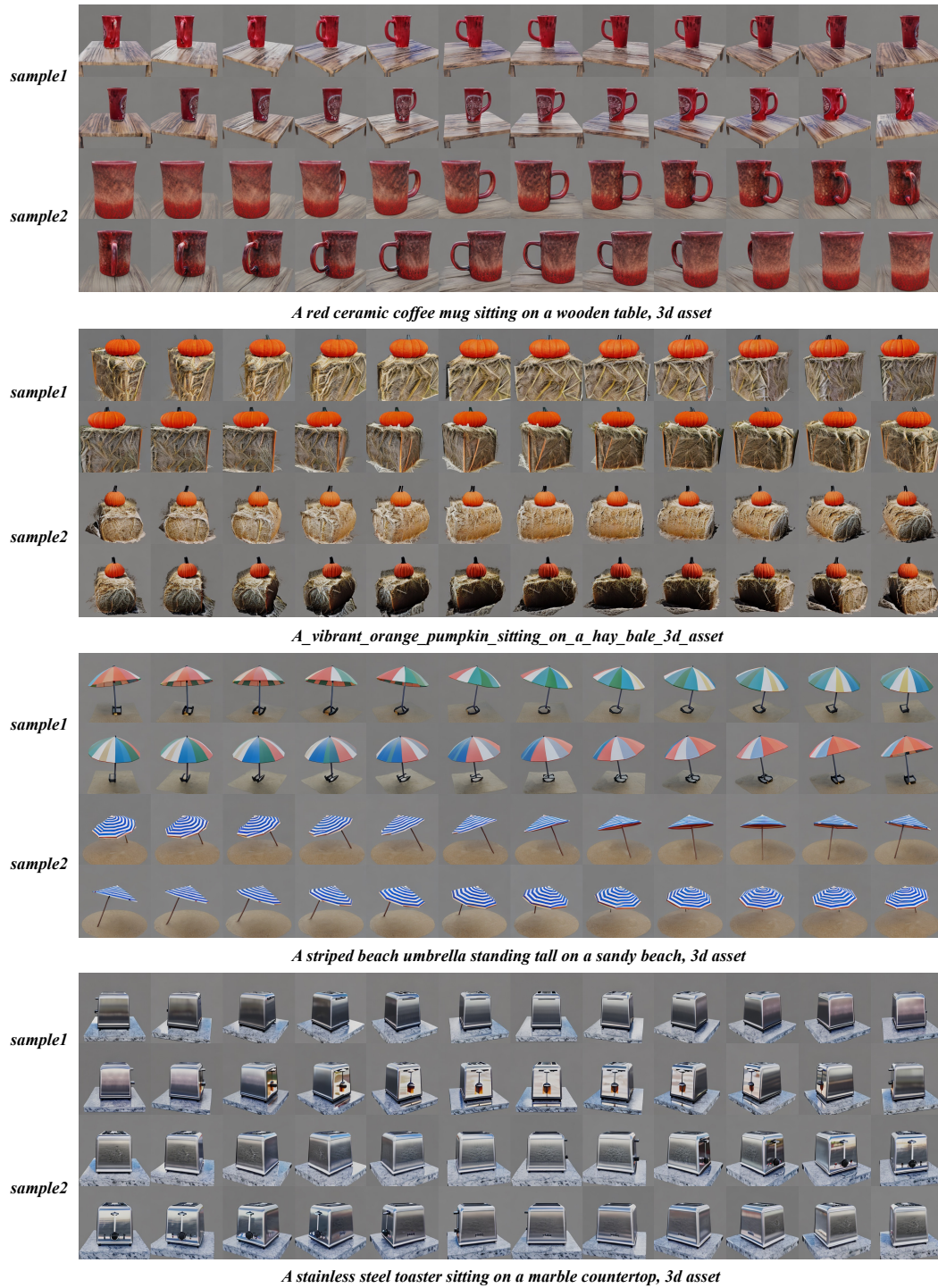


Figure 18: Visual various results of Text-to-Multi-View image generation on T3Bench. (Part V)



Figure 19: Visual results of Text-to-Multi-View image generation (Part I), prompts from DreamFusion.

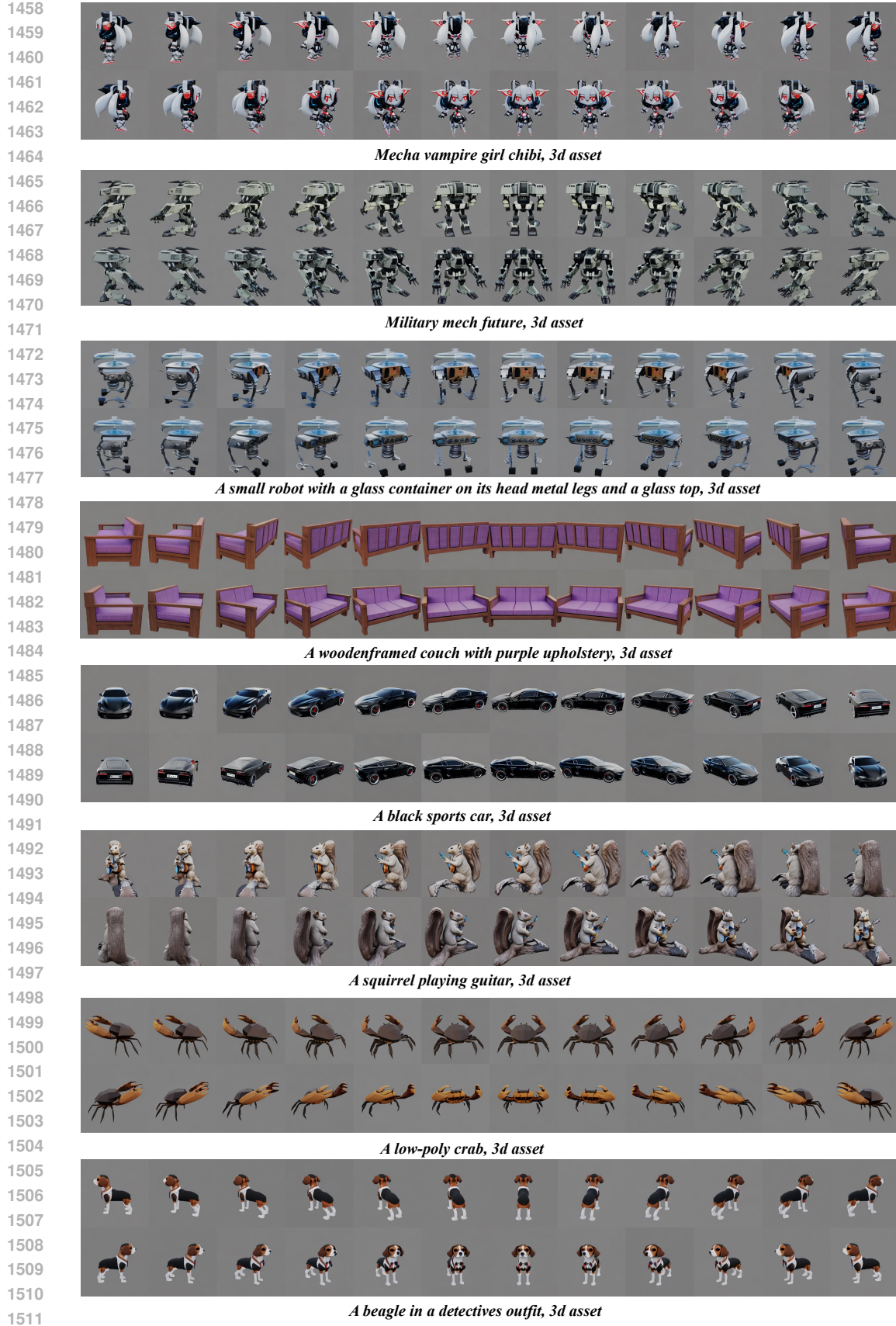


Figure 20: Visual results of Text-to-Multi-View image generation (Part II), prompts from DreamFusion.



The US Capitol building with a white exterior and goldenyellow dome, 3d asset



A blue poison dart frog sitting on a water lily, 3d asset



A frog wearing a sweater, 3d asset



A dachshund dressed up in a hotdog costume, 3d asset



The US Capitol building with a white exterior and goldenyellow dome, 3d asset



A beautiful rainbow fish, 3d asset



A blue motorcycle, 3d asset



A bear dressed as a lumberjack, 3d asset

Figure 21: Visual results of Text-to-Multi-View image generation (Part III), prompts from DreamFusion.

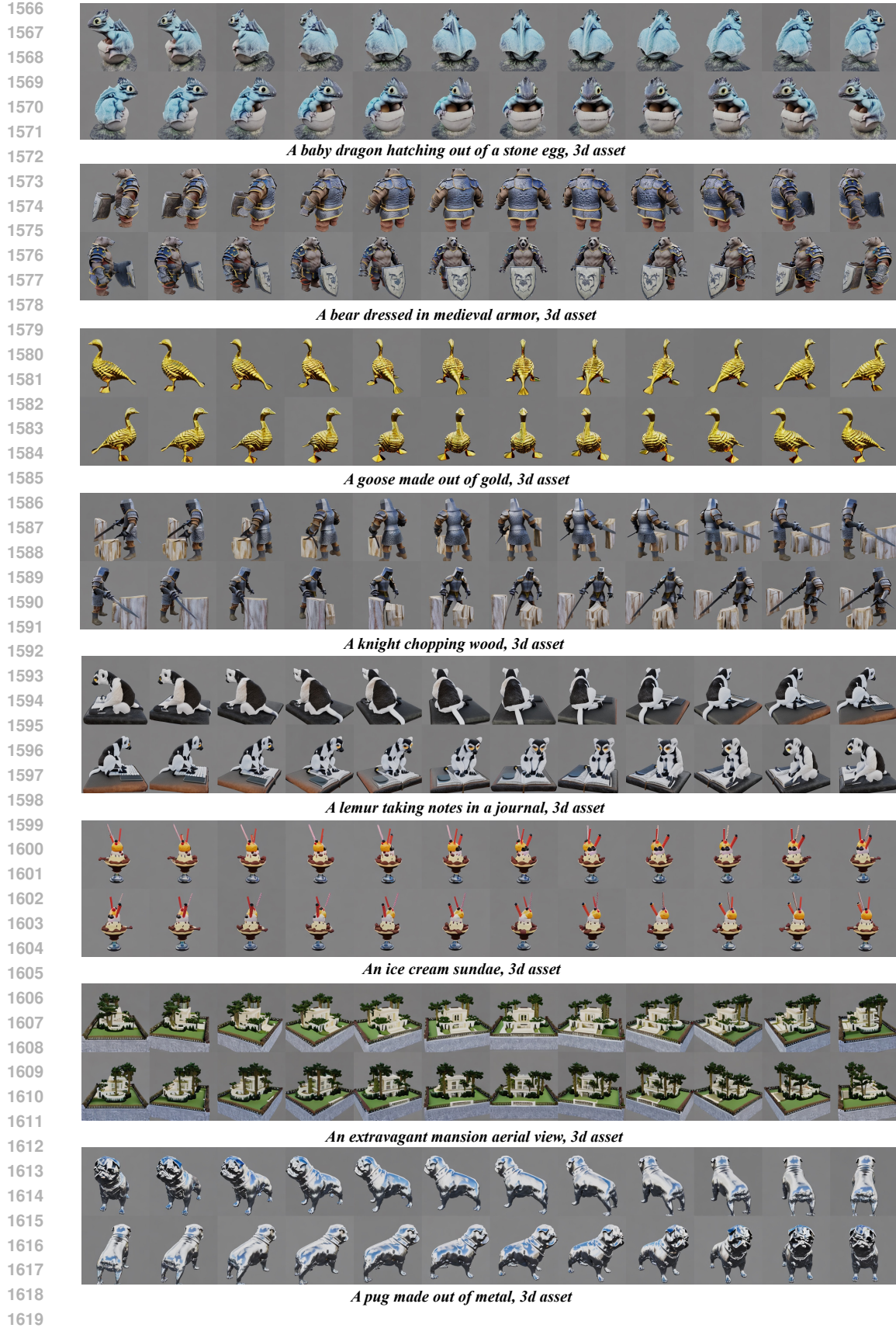
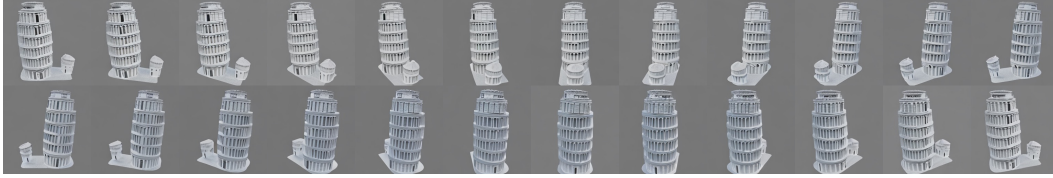


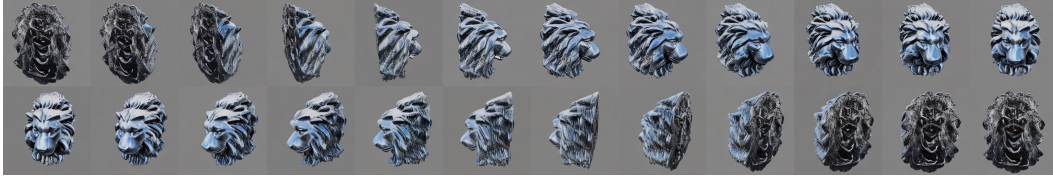
Figure 22: Visual results of Text-to-Multi-View image generation (Part IV), prompts from DreamFusion.



A terracotta bunny, 3d asset



The leaning tower of Pisa aerial view, 3d asset



A metal sculpture of a lions head highly detailed, 3d asset



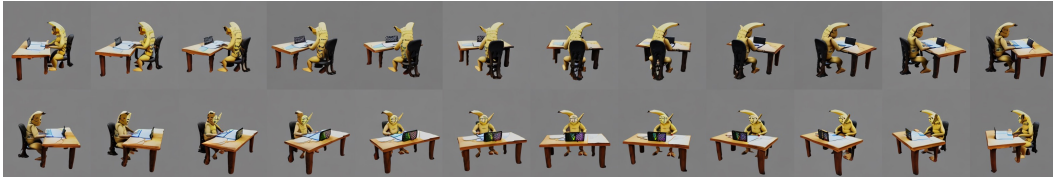
A shiny red stand mixer, 3d asset



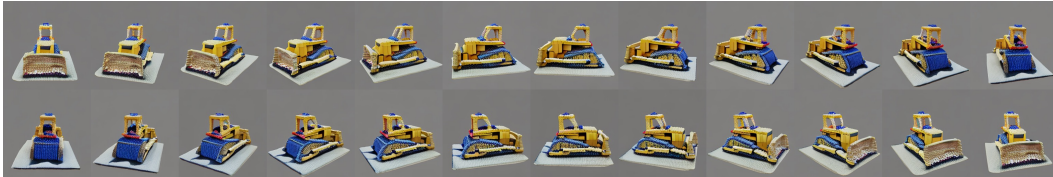
A baby bunny sitting on top of a stack of pancakes, 3d asset



A 3d model of an adorable cottage with a thatched roof, 3d asset



A wide angle DSLR photo of a humanoid banana sitting at a desk doing homework, 3d asset



A bulldozer made out of toy bricks, 3d asset

Figure 23: Visual results of Text-to-Multi-View image generation (Part V), prompts from DreamFusion.

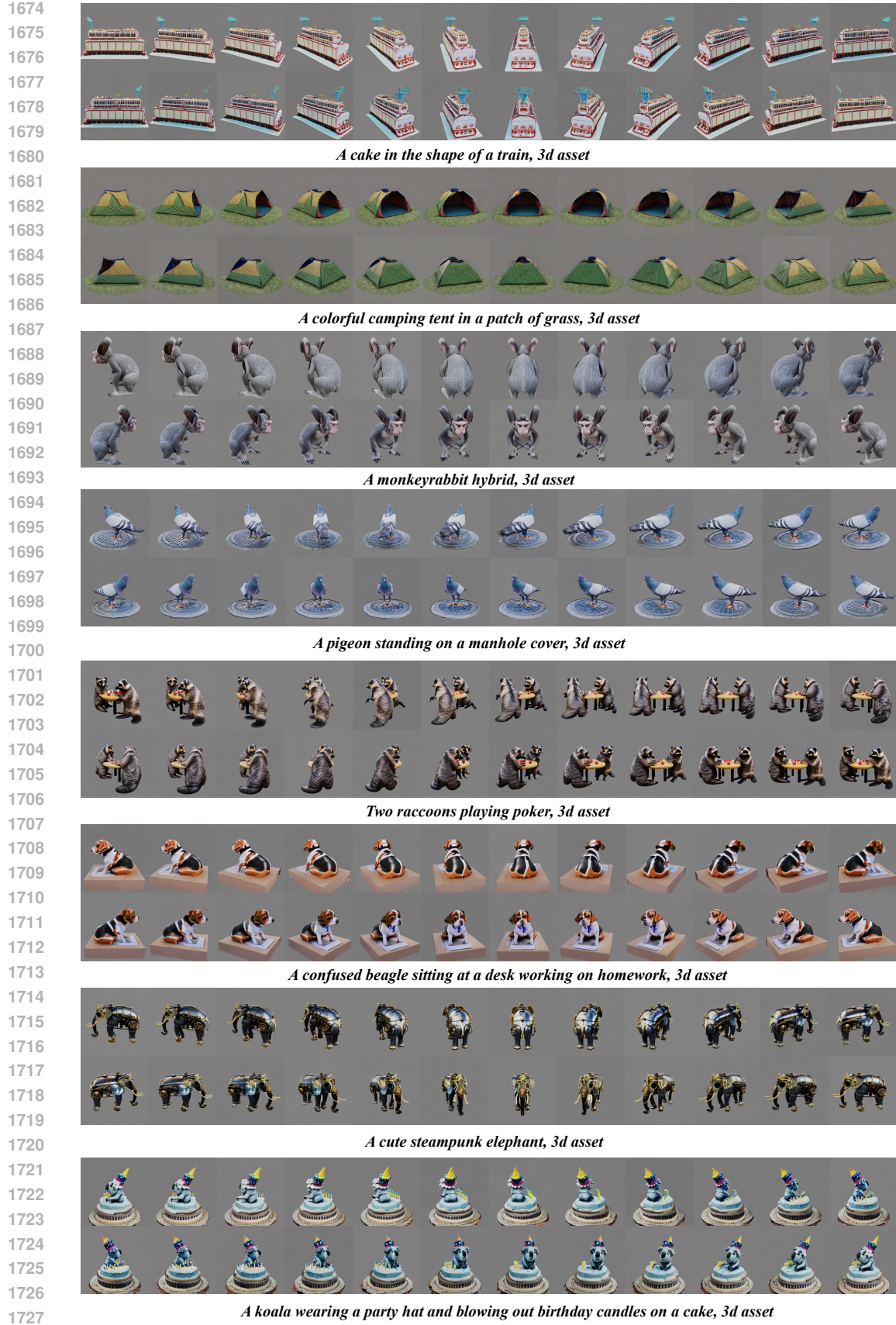


Figure 24: Visual results of Text-to-Multi-View image generation (Part VI), prompts from DreamFusion.



Figure 25: Visual results of Text-to-Multi-View image generation (Part VII), prompts from DreamFusion.

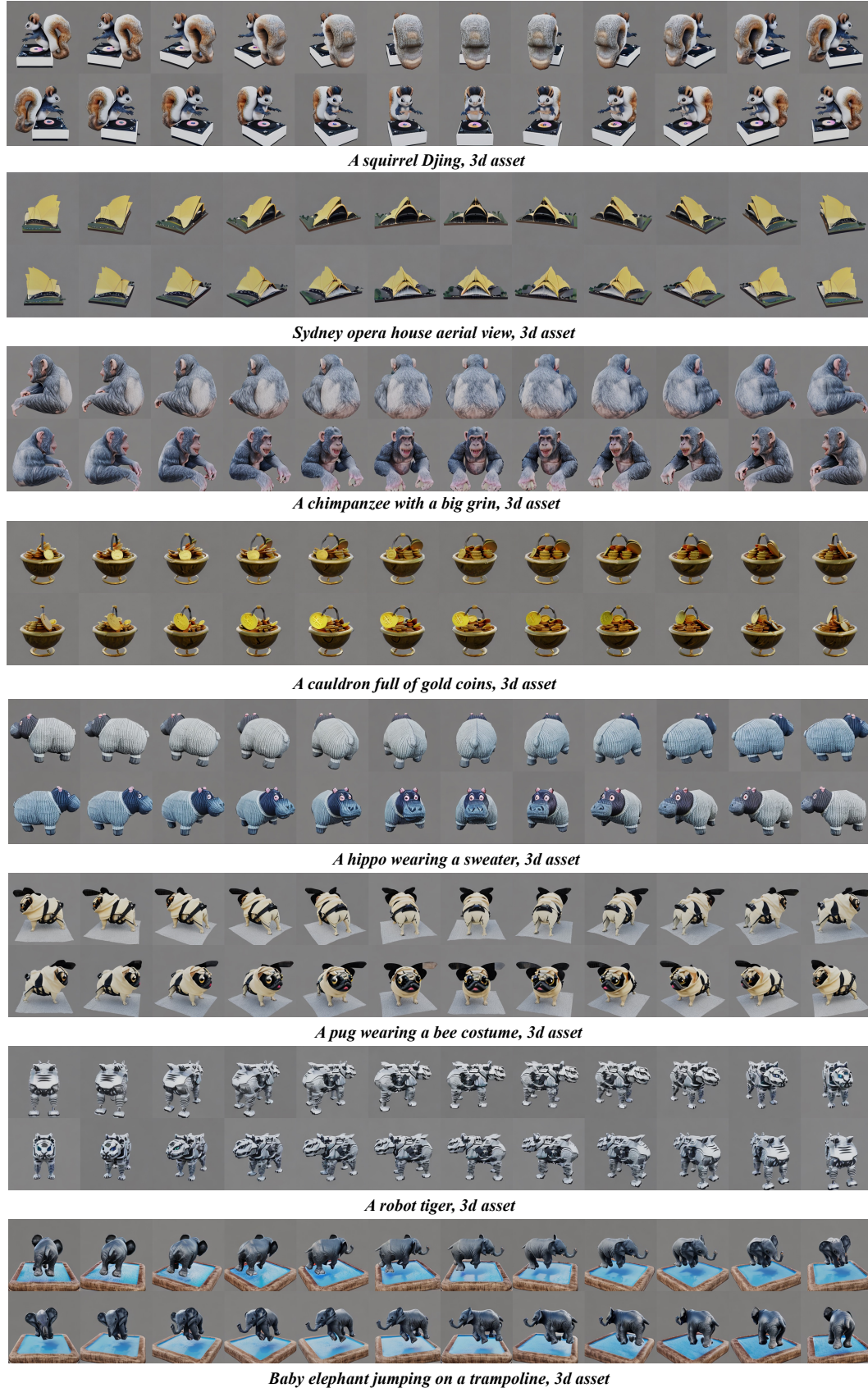


Figure 26: Visual results of Text-to-Multi-View image generation (Part VIII), prompts from DreamFusion.

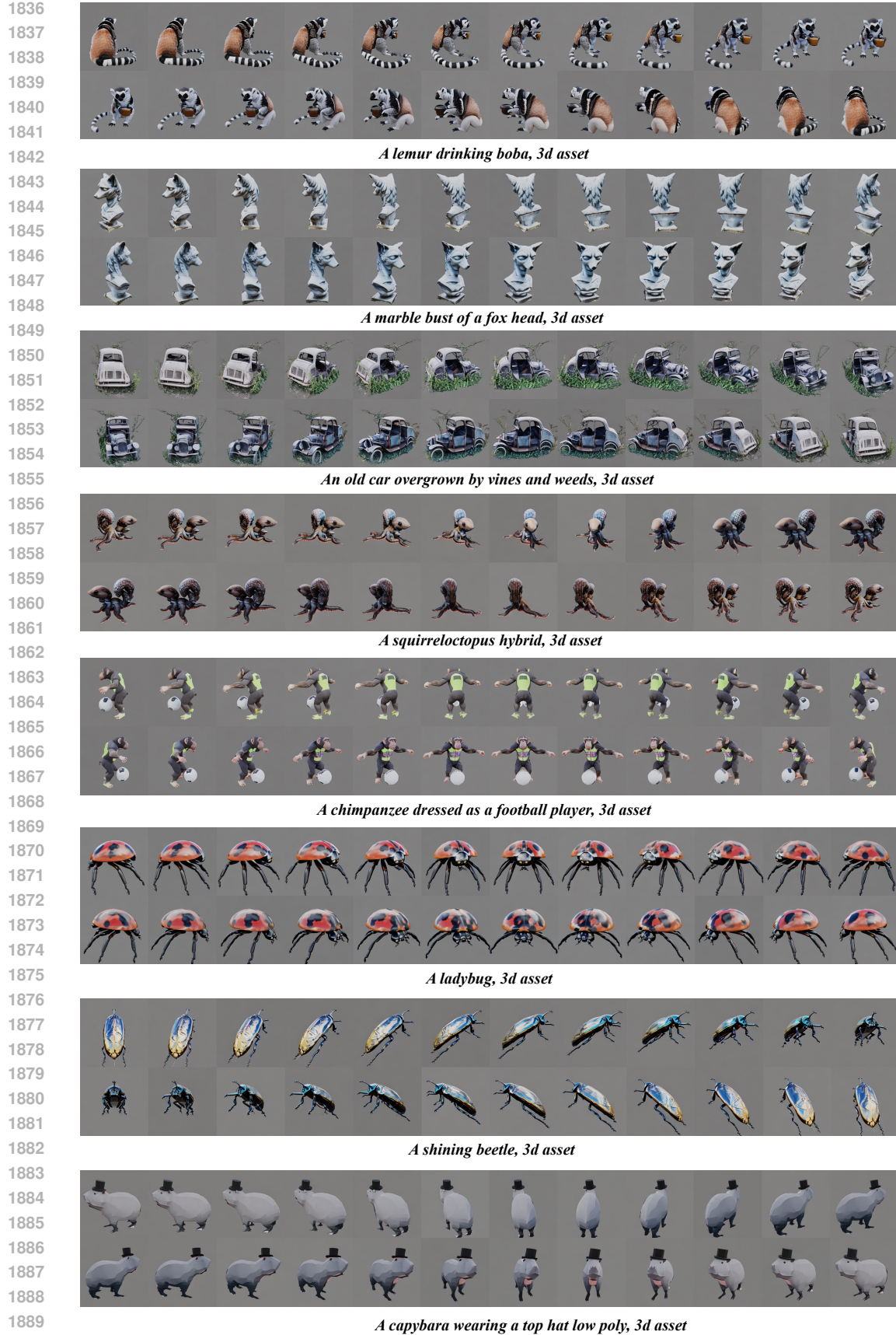


Figure 27: Visual results of Text-to-Multi-View image generation (Part IX), prompts from DreamFusion.

Table 4: Quantitative results of image-to-3D on GSO Downs et al. (2022) dataset

Method	Chamfer Dist.	Volume IOU
Zero123-XL	0.0354	0.4846
SyncDreameer	0.0278	0.5156
VideoMV	0.0257	0.5228

E IMAGE-TO-MULTI-VIEW IMAGE GENERATION

Although the input image provides some guidance for dense pixel generation in multi-view scenarios, VideoMV is capable of generating various plausible results even from invisible angles. We present two typical examples in Fig. 28 to illustrate that VideoMV can produce diverse yet faithful outputs based on the given image input.



Figure 28: Visual various results of Image-to-Multi-View image generation.

E.1 MORE QUALITATIVE RESULTS ON GOOGLE SCANNED OBJECT

Due to the page limitations in the main paper, we have included additional qualitative results of Google Scanned Objects in Fig. 29 for a comprehensive analysis. Note that the first image serves as the input for VideoMV.

E.2 MORE QUALITATIVE RESULTS ON WEB IMAGES

The VideoMV technique can also be applied to web images that lack underlying 3D models. In this study, we present a visualization of limited cases and encourage readers to experiment with our code on a wider range of web images. As illustrated in both Fig. 30 and Fig. 31, VideoMV demonstrates its capability to generate feasible results using either generated or in-the-wild images.

E.3 NUMERICAL RESULTS ON IMAGE-TO-3D

As a dense multi-view generative model, VideoMV aims to tackle the challenging task of synthesizing novel views with higher density and consistency based on a given prompt or single image. Unlike previous approaches Liu et al. (2023b;c), we do not employ any reconstruction optimization in VideoMV. Instead, inspired by prior work Wang et al. (2021), we present relevant Volume IOU and Chamfer Distance metrics on the GSO dataset using the off-the-shelf MVS method, such as NeuS Wang et al. (2021). As depicted in Tab. 4, VideoMV outperforms state-of-the-art methods in terms of Chamfer Distance and Volume IOU metrics, indicating that leveraging increased consistency in multi-view images for reconstruction can result in improved accuracy in 3D geometry.

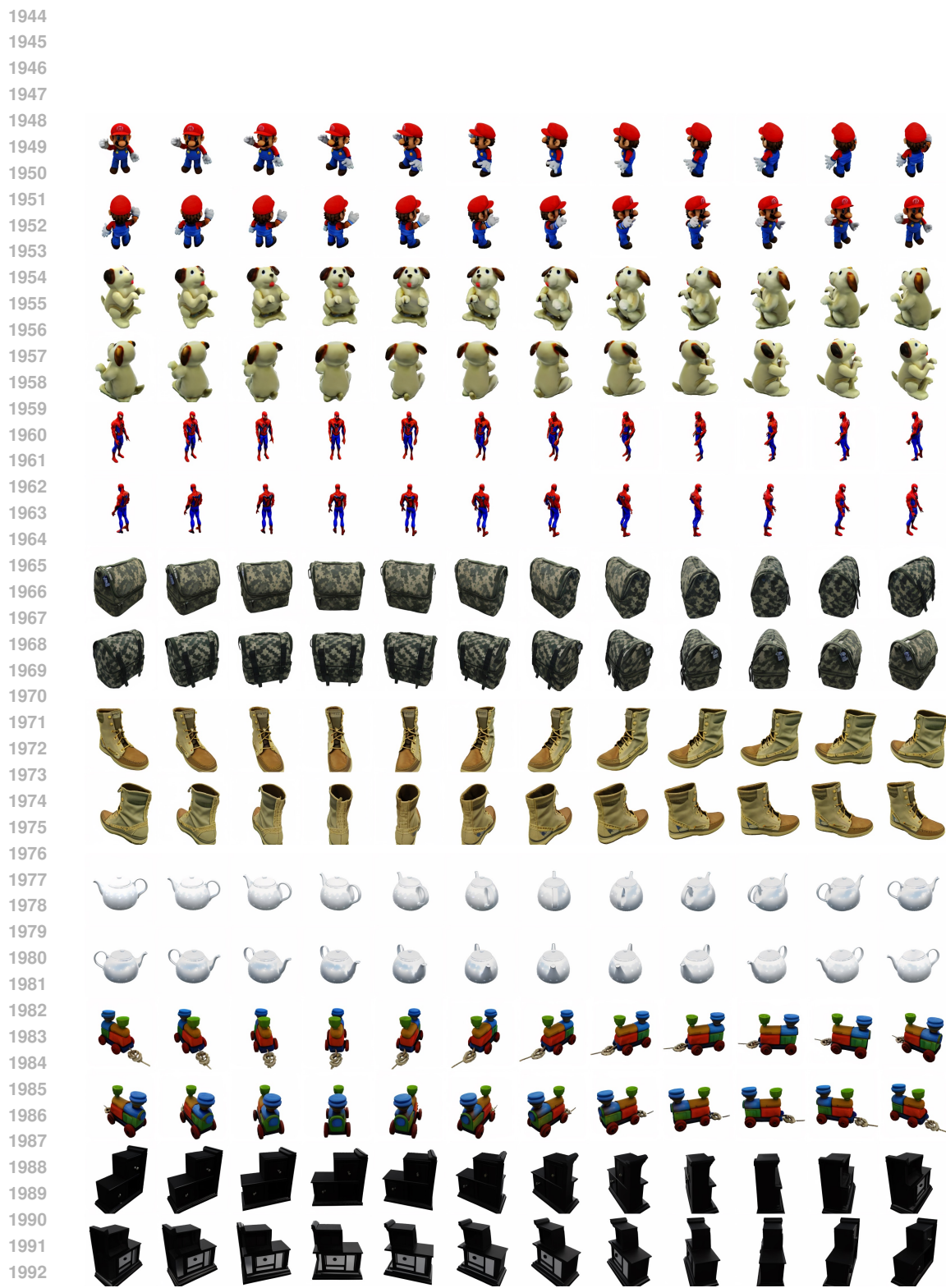


Figure 29: Visual results of Image-to-Multi-View image generation.

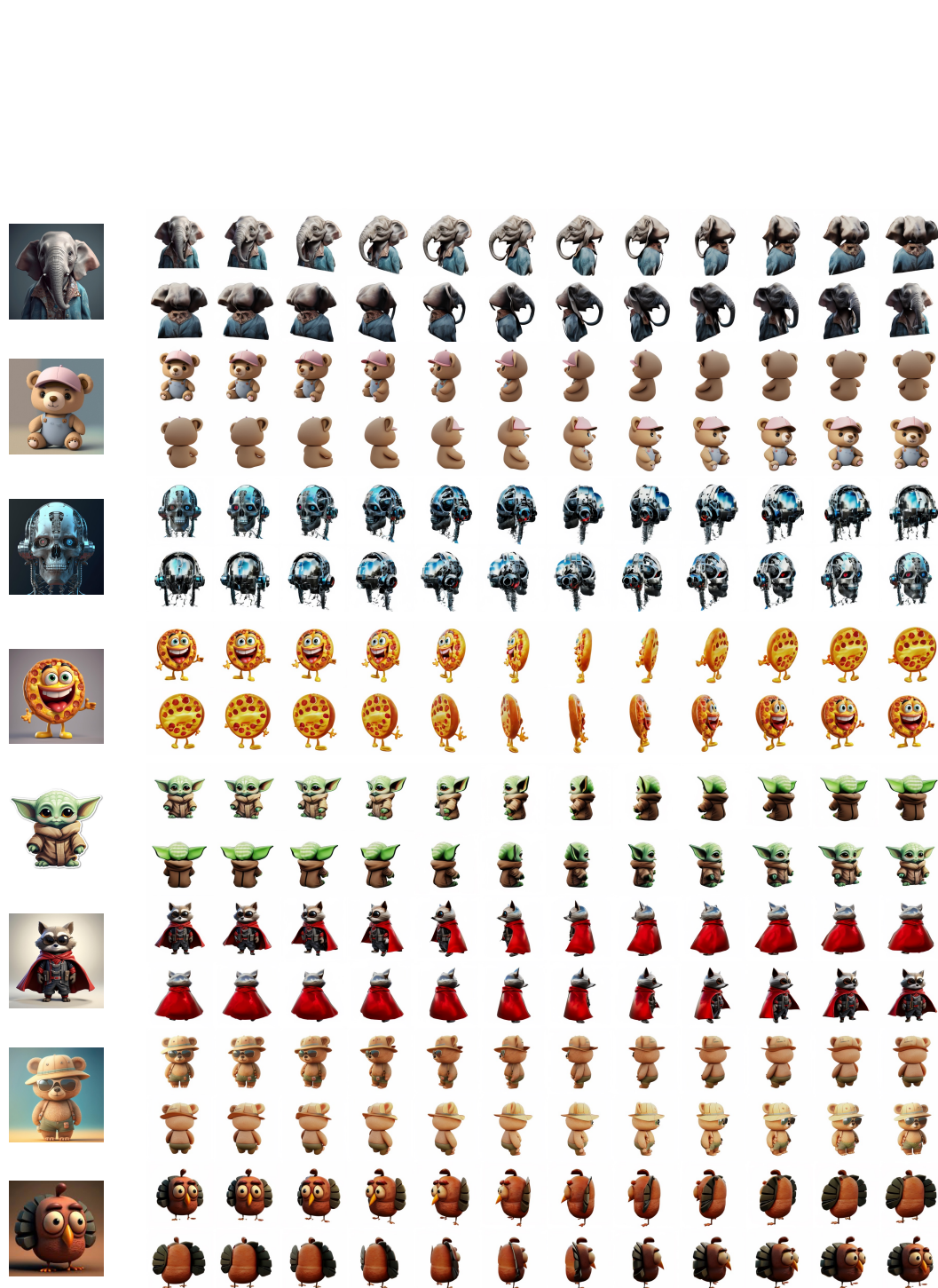


Figure 30: Visual results of Image-to-Multi-View image generation from web images. (Part I)

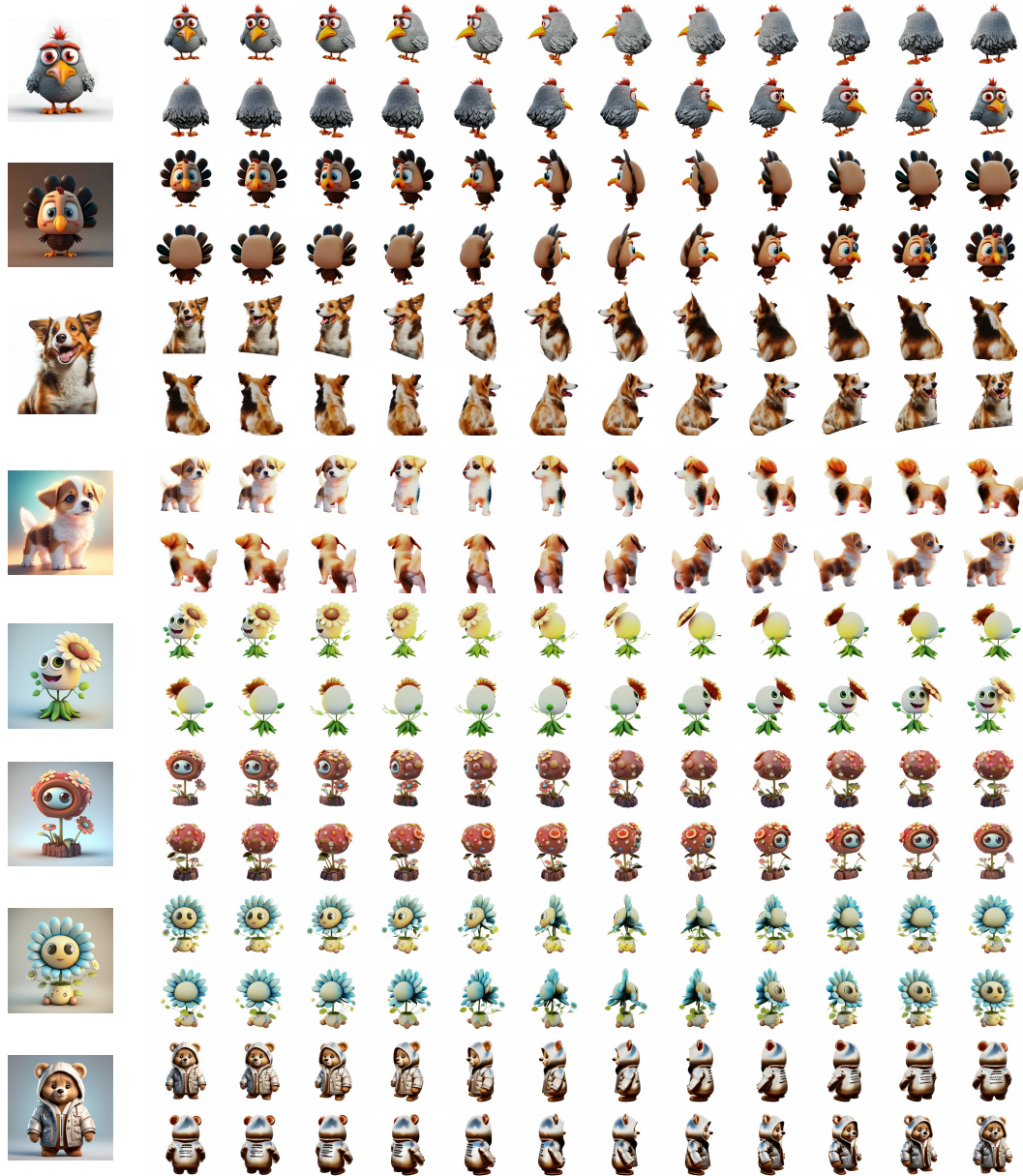


Figure 31: Visual results of Image-to-Multi-View image generation from web images. (Part II)