EfficientDreamer: High-Fidelity and Stable 3D Creation via Orthogonal-view Diffusion Prior (Supplementary Materials)

Zhipeng Hu[1†], Minda Zhao[1†], Chaoyi Zhao[1], Xinyue Liang[1], Lincheng Li[1*],
Zeng Zhao[1], Changjie Fan[1], Xiaowei Zhou[2], Xin Yu[3]
[1]NetEase Fuxi AI Lab, [2]State Key Lab of CAD&CG, Zhejiang University [3]University of Queensland

1. More Implementation Details

1.1. Orthogonal-view Diffusion Model Training

We build our newly introduced orthogonal-view diffusion model on the *diffusers* [1] library. To fine-tune our model, we adopt the Stable Diffusion v2.1 as the base model and use the AdamW optimizer with ϵ-prediction. The training process involves 200K steps, with a warmup of 10K steps.

1.2. Text-to-3D Generation

In the coarse stage, we utilize NeuS [4] with a rendering resolution of 64×64 to optimize the 3D representation. The random camera distance is set within the range of [1.5, 2.0], while the fov range is set within the range of [40°, 70°]. We choose elevation between [-10°, 45°]. We adopt progressive levels of details from Neuralangelo [2] and the hash encoding resolution [3] spans from 2^4 to 2^{11} with 16 levels. The learning rate is set from 0.001 to 0.0002 with LinearLR scheduler. Similar settings are also utilized in the fine stage.

2. More Discussion

Generalizability of the Orthogonal-view Diffusion Model: We train the orthogonal-view diffusion model using the Objaverse dataset [1], which consists of hundreds of thousands of 3D models. However, it is important to note that the scale of the 3D dataset is relatively small compared to the text-image training pairs that are necessary to train the 2D diffusion model. Hence, it is crucial to clarify the generalizability of the orthogonal-view diffusion model, especially for unseen and counterfactual scenes whose details can be specified through text prompts. First, for objects mentioned in the text prompts that exist in the Objaverse dataset, we retrieve the most similar 3D models to the ones generated by EfficientDreamer. The results, as shown in Fig. 1(a), demonstrate that EfficientDreamer is capable of generating 3D models for specific text prompts, without overfitting to the 3D models in the training 3D dataset. Secondly, as depicted in Fig. 1(b), we can also achieve high-fidelity 3D creation for scenes that are not included in the 3D dataset. This further confirms the excellent generalizability of our orthogonal-view diffusion model.

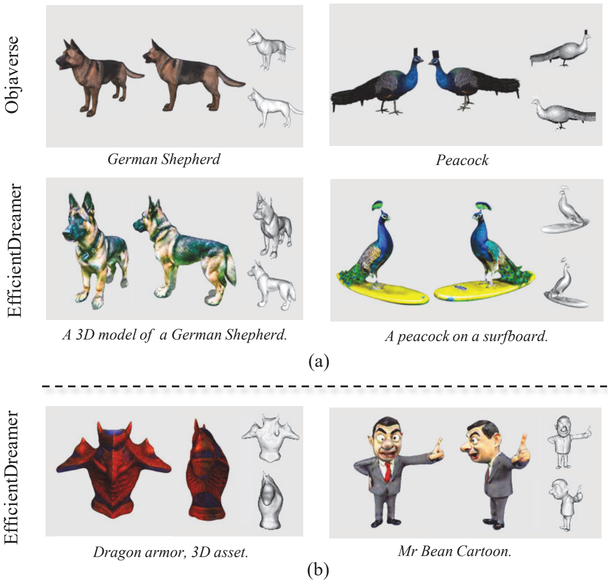
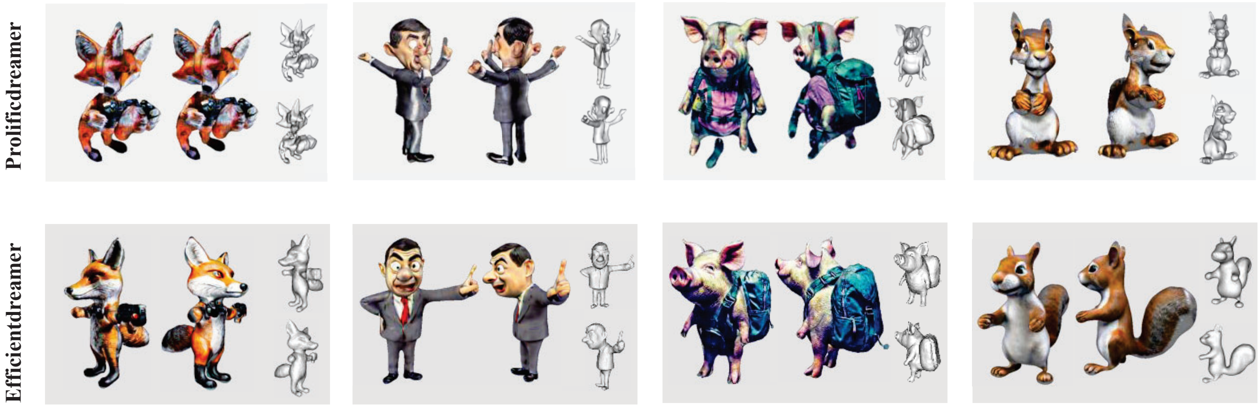


Figure 1. Visualization results for more discussion. (a) Objects from the Objaverse dataset or EfficientDreamer for the given text prompts. (b) 3D models generated by EfficientDreamer, which are not included in the Objavserse dataset.

Impact of Variational Score Distillation (VSD) in ProlificDreamer: Once we obtain precise 3D representations for the given text prompts using our orthogonal-view diffusion model, we utilize the SDS and VSD for geometry and texture optimization in the fine stage, following the method proposed by ProlificDreamer [5]. However, we demonstrate that such operations in the fine stage cannot solve the Janus problem, as the VSD guidance still relies on the pre-

[1] https://github.com/huggingface/diffusers



A 3D model of a fox holding a videogame controller.

Mr Bean Cartoon.

A pig wearing a backpack, high quality.

A squirrel, animated movie character, high detail 3D model.

Figure 2. Comparison with ProlificDreamer with Variational Score Distillation (VSD).

trained diffusion model. To verify this, we generate the 3D models following the procedure in ProlificDreamer. In the first stage, we optimize the 3D representation using 64×64 NeRF rendering and then optimize the geometry and texture using SDS and VSD guidance. The results are shown in Fig. 2. It is evident that these models still exhibit noticeable Janus problems, while our EfficientDreamer can effectively resolve these issues.

3. More Results

In Fig. 3, Fig. 4 and Fig 5, we present additional comparison results of 3D generation on various text prompts, utilizing both our method and other text-to-3D methods. In Fig. 6, Fig. 7 and Fig. 8, we compare our method with other methods by exhibiting the generated 3D models from eight different viewpoints. We also show the corresponding normal maps of our method. The results demonstrate that our methods can effectively resolve the Janus problem while enhancing the quality of 3D content creation. For more comprehensive results, please refer to the supplementary video and project page: `https://efficientdreamer.github.io`.

References

[1] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 1

[2] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8456–8465, 2023. 1

[3] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 1

[4] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 1

[5] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023. 1

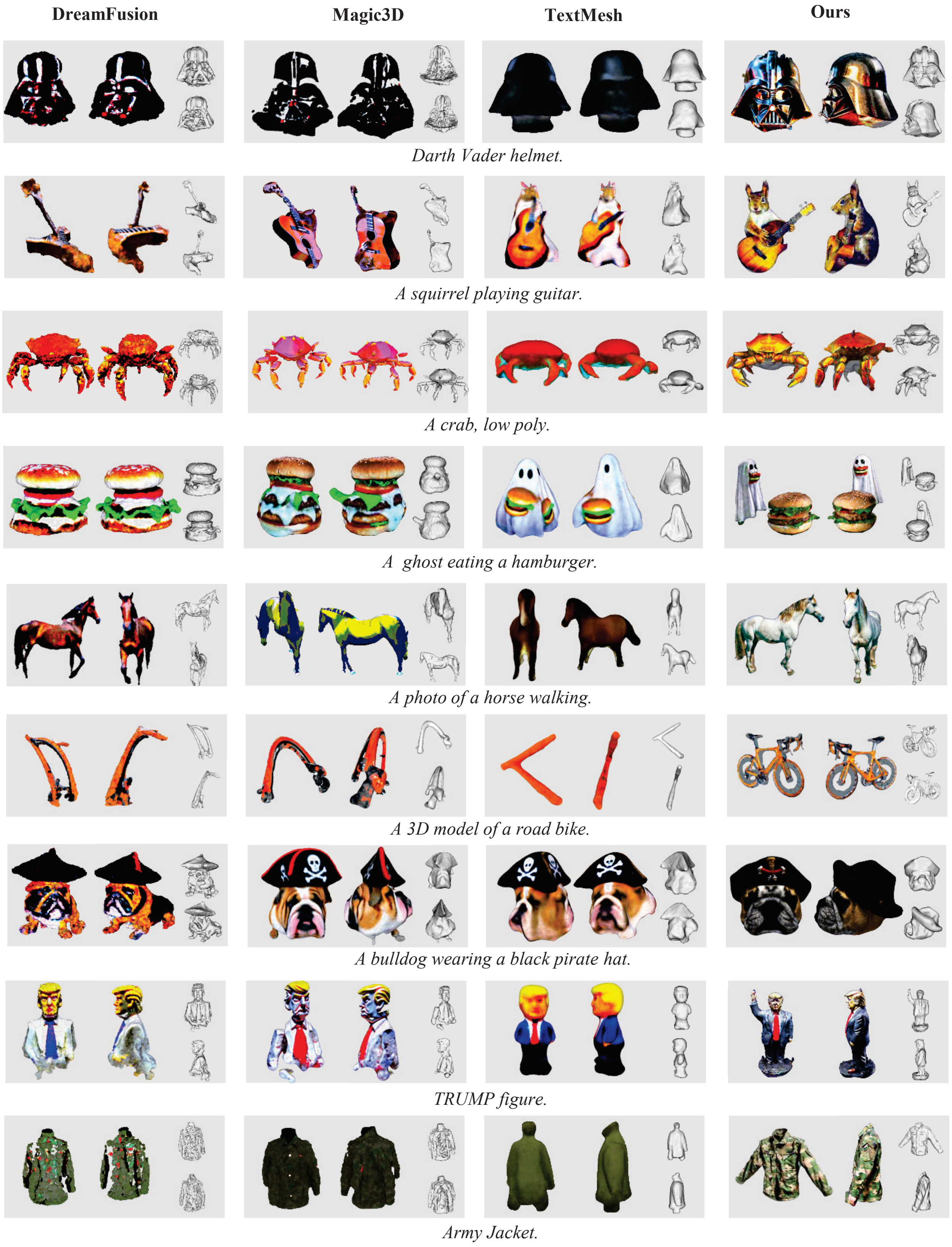

DreamFusion
Magic3D
TextMesh
Ours
Darth Vader helmet.
A squirrel playing guitar.
A crab, low poly.
A ghost eating a hamburger.
A photo of a horse walking.
A 3D model of a road bike.
A bulldog wearing a black pirate hat.
TRUMP figure.
Army Jacket.

Figure 3. Comparison with other text-to-3D methods. We render each 3D model from two views.

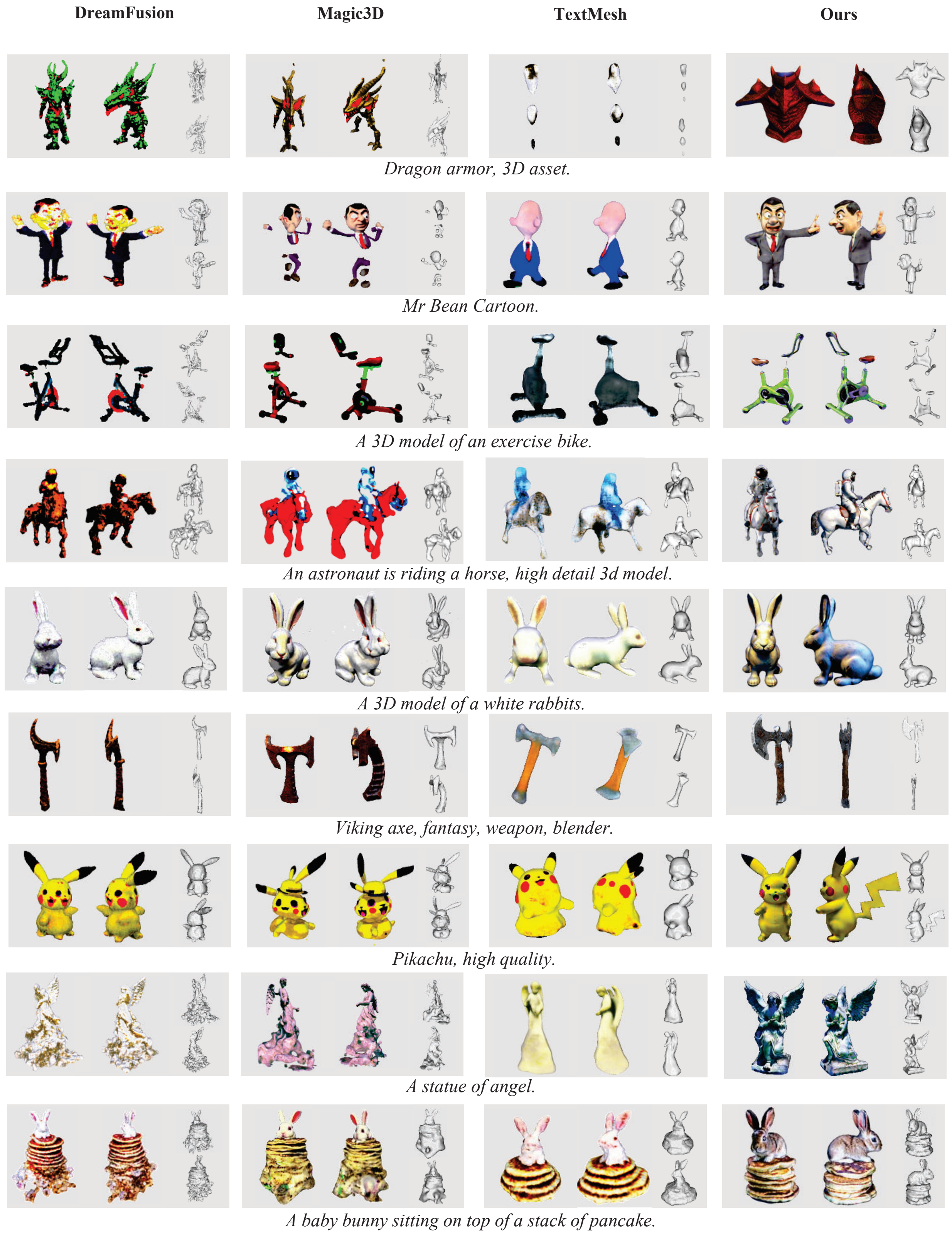
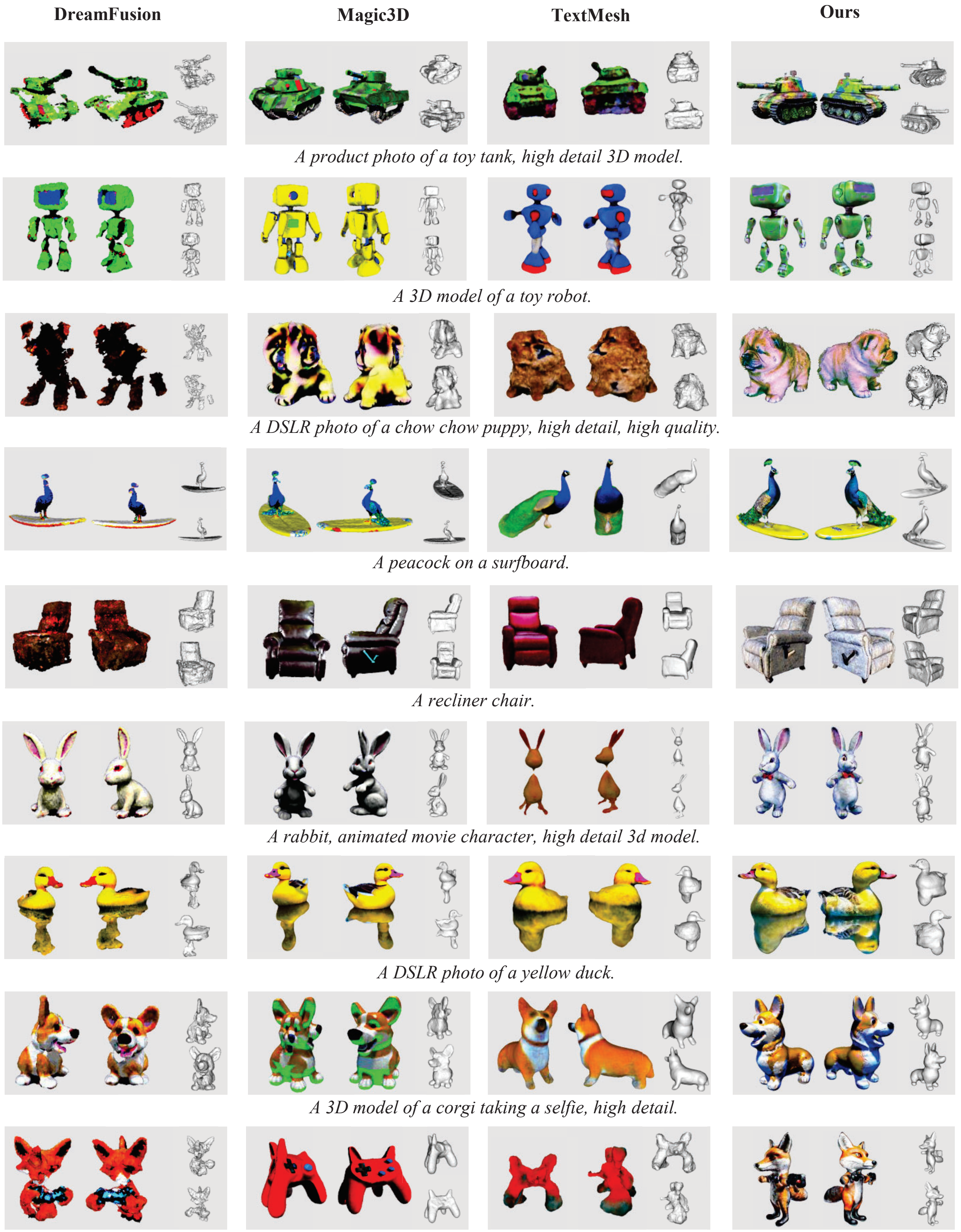

DreamFusion
Magic3D
TextMesh
Ours
Dragon armor, 3D asset.
Mr Bean Cartoon.
A 3D model of an exercise bike.
An astronaut is riding a horse, high detail 3d model.
A 3D model of a white rabbits.
Viking axe, fantasy, weapon, blender.
Pikachu, high quality.
A statue of angel.
A baby bunny sitting on top of a stack of pancake.

Figure 4. Comparison with other text-to-3D methods. We render each 3D model from two views.



A product photo of a toy tank, high detail 3D model.

A 3D model of a toy robot.

A DSLR photo of a chow chow puppy, high detail, high quality.

A peacock on a surfboard.

A recliner chair.

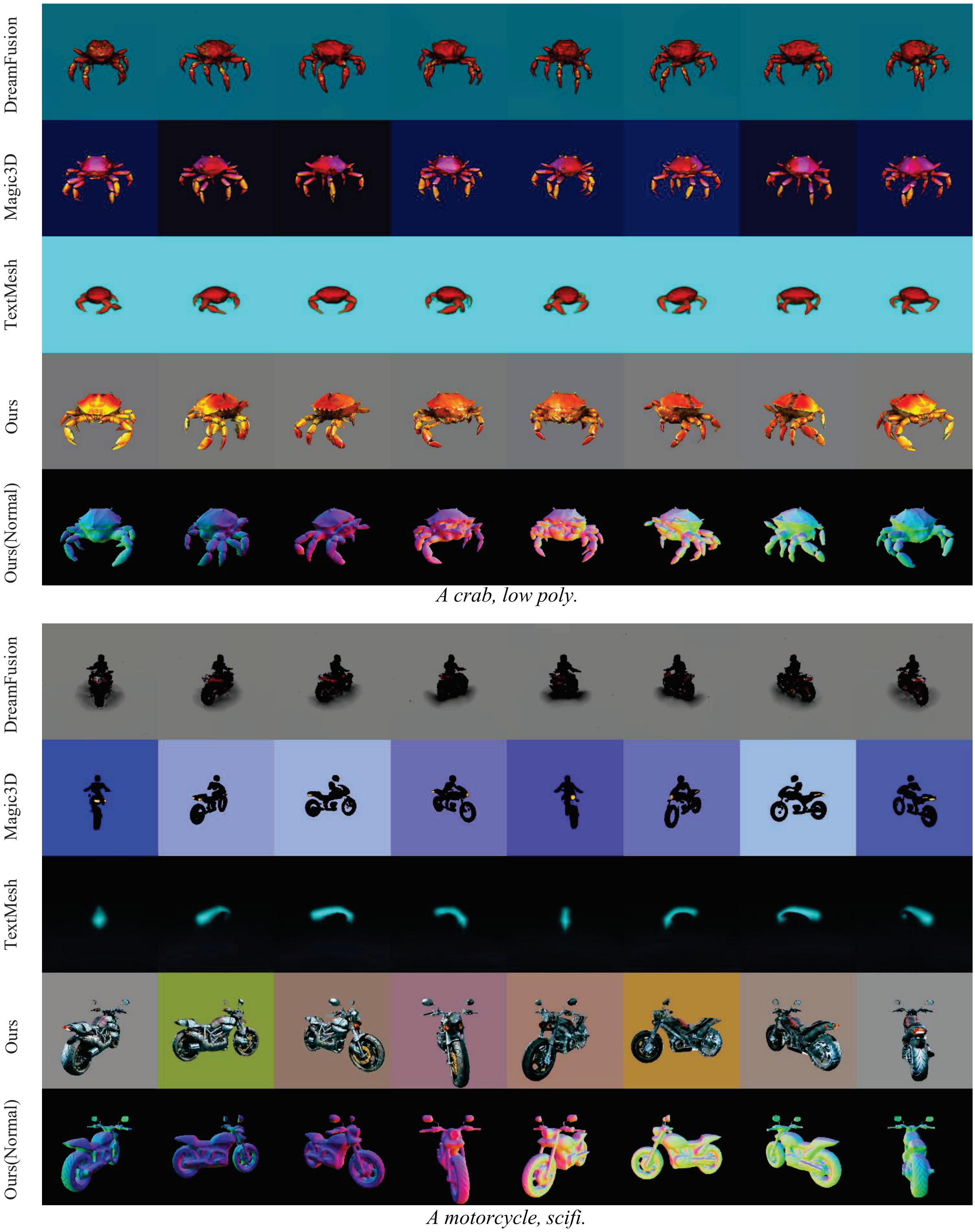
A rabbit, animated movie character, high detail 3d model.

A DSLR photo of a yellow duck.

A 3D model of a corgi taking a selfie, high detail.

A 3D model of a fox holding a videogame controller.

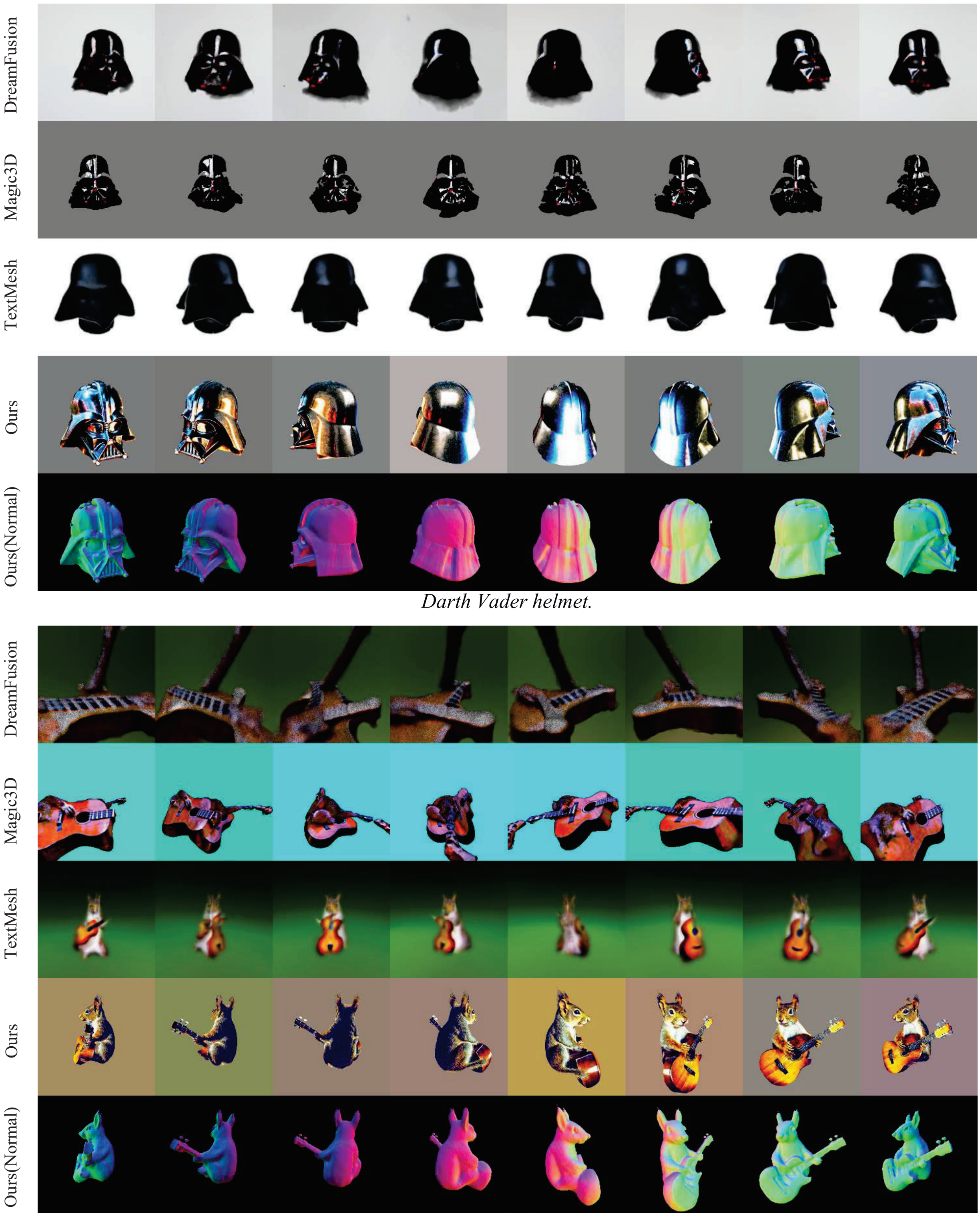
Figure 5. Comparison with other text-to-3D methods. We render each 3D model from two views.



A crab, low poly.

A motorcycle, scifi.

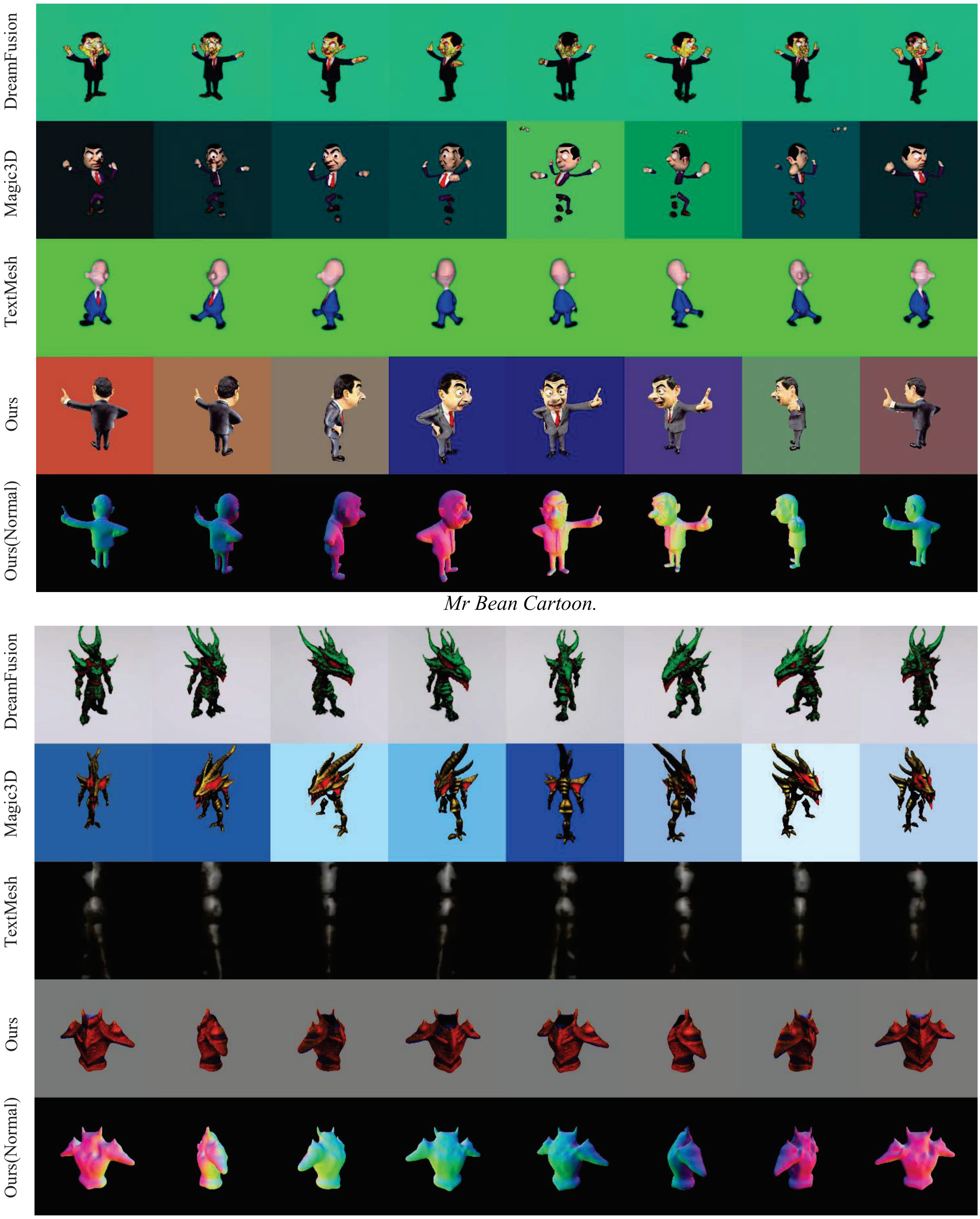
Figure 6. Comparison with other text-to-3D methods: In our evaluation, we render images from eight uniformly sampled views for each method. Additionally, we present the normal maps of each object generated by our approach, providing a comprehensive visual representation of the 3D models.



Darth Vader helmet.

A squirrel playing guitar.

Figure 7. Comparison with other text-to-3D methods: In our evaluation, we render images from eight uniformly sampled views for each method. Additionally, we present the normal maps of each object generated by our approach, providing a comprehensive visual representation of the 3D models.



Mr Bean Cartoon.

Dragon armor, 3D asset.

Figure 8. Comparison with other text-to-3D methods: In our evaluation, we render images from eight uniformly sampled views for each method. Additionally, we present the normal maps of each object generated by our approach, providing a comprehensive visual representation of the 3D models.