

---

# Wukong’s 72 Transformations: High-fidelity 3D Morphing via Flow Models – *Appendix* –

---

## Contents

<b>A</b>	<b>Model details</b>	<b>2</b>
<b>B</b>	<b>Ablation study</b>	<b>2</b>
<b>C</b>	<b>Rectified flow models vs. diffusion models</b>	<b>3</b>
<b>D</b>	<b>Comparison with other methods</b>	<b>4</b>
	D.1 Comparison with other textured 3D method . . . . .	4
	D.2 Comparison with shape morphing methods . . . . .	4
<b>E</b>	<b>More results</b>	<b>6</b>
<b>F</b>	<b>Broader impact</b>	<b>6</b>
<b>G</b>	<b>Limitation</b>	<b>6</b>

## A Model details

To enable high-fidelity textured 3D morphing, we build upon two pretrained flow-based transformer models introduced in Trellis (Xiang et al., 2024): structure flow model and SLat flow model, originally designed for unconditional 3D generation. The structure flow model operates on structured latent representation and follows a transformer-based architecture with 24 modulated transformer blocks with cross attentions. Each block contains self-attention, cross-attention, and feed-forward components, modulated via AdaLN (Guo et al., 2022) conditioning from a learned timestep embedding. Root Mean Square Normalization (RMSNorm) (Zhang and Sennrich, 2019) is applied to both the query and key representations prior to their use in the attention mechanism. The SLat flow model incorporates a hierarchical design with sparse 3D convolutional blocks and positional embeddings to encode spatial context. The transformer core comprises 24 modulated sparse transformer blocks with cross attentions, analogous in structure to the geometry model but enhanced with sparse attention and feed-forward operations. Additionally, the model includes dedicated output convolutional blocks for upsampling and decoding, ensuring fine-grained preservation and modulation of high-frequency texture details. We use the free-support Wasserstein barycenter solver from the POT library (`ot.lp.free_support_barycenter` (Lindheim, 2023)), which is based on linear programming (LP). The cost matrix is computed using the squared Euclidean distance in the CLIP (Radford et al., 2021) (for text) and DINOv2 (Oquab et al., 2023) (for image) embedding space.

## B Ablation study

To ensure a fair, apples-to-apples comparison with 3DRM, we implemented our full morphing method on top of the GaussianAnything (Yushi et al., 2025) framework—the same 3D generator used in 3DRM (Yang et al., 2025), results are shown in Table A1. We denote this variant as “Ours\*” in the table below. Across all evaluation metrics, including FID, PPL, V-CLIP, and GPT-based perceptual scores (STP-GPT, SEP-GPT), our method consistently outperforms 3DRM, even when both share the exact same backbone. This clearly demonstrates that the improvement is not solely due to the use of a stronger generator like Trellis, but rather stems from our core morphing algorithm. Note that the GPT-based results may differ from those presented in main paper, as they are computed using only the four methods reported here.

Table A1: **Quantitative comparison with GaussianAnything as backbone.**

Model	FID ↓	STP-GPT ↑	SEP-GPT ↑	PPL ↓	V-CLIP ↑
MorphFlow	147.70	0.38	0.41	3.10	0.78
3DRM	6.36	0.85	0.80	3.02	0.84
Ours*	5.15	0.93	0.91	2.94	0.87
<b>Ours</b>	<b>4.01</b>	<b>1.00</b>	<b>1.00</b>	<b>2.91</b>	<b>0.90</b>

Besides, We conduct quantitative evaluations with different threshold  $\tau$  values and present the results below as shown in Table A2. We observe that the performance is robust across a reasonable range of thresholds (0.2-0.8). We set a default threshold 0.3 in our evaluation.

Table A2: **Ablation study on threshold  $\tau$ .**

Threshold $\tau$	0.2	0.3	0.4	0.6	0.8
FID ↓	4.54	4.20	4.17	4.25	4.49
PPL ↓	2.94	2.91	2.91	2.92	2.93
V-CLIP ↑	0.88	0.90	0.91	0.90	0.87

To evaluate the method’s generalization to real 3D data, we conducted experiments using the Headspace dataset (Dai et al., 2020), which contains high-quality 3D face scans along with corresponding rendered RGB images. In our pipeline, we used these rendered images as inputs and passed them through the DINOv2 (Oquab et al., 2023) encoder to extract texture and semantic features for morphing. The outputs were generated by our standard pipeline without any architectural changes or fine-tuning. Despite relying on pretrained components, our method shows strong generalization to

real-world 3D scans. Our method outperformed both MorphFlow (Tsai et al., 2022) and 3DRM (Our own implementation) (Yang et al., 2025) on the same evaluation protocol. Quantitative results are shown below in Table A3:

Table A3: **Quantitative results on Headspace dataset.**

Model	FID ↓	STP-GPT ↑	SEP-GPT ↑	PPL ↓	V-CLIP ↑
MorphFlow	95.24	0.53	0.47	3.22	0.84
3DRM	6.61	0.83	0.77	3.04	0.88
<b>Ours</b>	<b>3.97</b>	<b>1.00</b>	<b>1.00</b>	<b>2.88</b>	<b>0.96</b>

## C Rectified flow models vs. diffusion models

**Continuity** The first reason we choose the flow model over the diffusion model for 3D morphing is its mathematically grounded continuity with respect to the interpolation parameter  $\alpha$ . In flow-based generative models, the mapping from  $\alpha$  to the output  $F(\alpha)$  is deterministic and constructed via an invertible transformation  $T(z; \alpha)$ , typically defined by an ordinary differential equation (ODE). Under standard regularity conditions (e.g., Lipschitz continuity of the velocity field), the solution  $T(z; \alpha)$  is guaranteed to be continuously differentiable with respect to  $\alpha$  (Loud, 1987), ensuring that the morphing trajectory forms a smooth path in the output space. This deterministic nature makes it possible to precisely control intermediate shapes and textures, yielding consistent and artifact-free transitions.

In contrast, diffusion models are typically governed by stochastic differential equations (SDEs), which introduce randomness throughout the generative process. While deterministic sampling methods like DDIM (Song et al., 2021) exist and are widely used, the underlying denoising process in these models often follows a stochastic trajectory. Consequently, even with interpolated conditioning, the same value of  $\alpha$  can yield different outputs across runs. This inherent variability makes it difficult to ensure continuity or precise control in the morphing sequence, particularly at intermediate points where uncertainties compound. In contrast, rectified flow models generate deterministic and unique interpolation paths, enabling forward integration with a guaranteed likelihood formulation. This property makes them more suitable for achieving smooth, stable, and controllable interpolation, which aligns with our need for consistency in the latent space.

**Convexity and optimality** There are theoretical guarantees for flow models like the rectified flow model in maintaining the convexity of data during the generation process. This linearity ensures that any intermediate sample lies within the convex hull of the endpoints, thereby preserving the convexity of the data. In practice, the trajectory is hard to remain straight. There is analysis (Liu et al., 2023) on the straightness error on the trajectory, which states that even an imperfect trajectory is close enough to straight lines and ensure convexity of data to some extent. Theorems in this analysis further emphasize the uniqueness and optimality of the solution of rectified flow in matching distributions under convex cost functions. For diffusion models, the backward process generates data from the prior but does not theoretically guarantee convexity preservation. These models focus on matching data distributions, not preserving geometric properties like convexity. There is no theatrical guarantee on the data convexity in the backward process. The noise term in the reverse-time SDE can easily violate the convexity of original data. Also, under the same condition as the rectified flow model, the path of diffusion models is not assured to be optimal. There exists certain crossing flows in the matching of two distributions, leading to features that are out-of-distribution in practice.

**Inference speed** The theoretical basis for the faster inference of flow models (such as rectified flow models) primarily stems from the geometric properties of their trajectories and the efficiency of numerical simulation. For rectified flow model used in this paper, it aims to make generation trajectories as straight as possible. For ideal straight-line flows, the trajectory between any two points  $Z_0 \sim \pi_0$  and  $Z_1 \sim \pi_1$  is given by the linear interpolation  $Z_t = tZ_1 + (1 - t)Z_0$ . In this case, the drift field of the ODE is a constant  $v(Z_t, t) = Z_1 - Z_0$ , which can be solved exactly with a single Euler step:  $Z_1 = Z_0 + v(Z_0, 0) \cdot 1$ . This eliminates the need for time discretization errors. Even in non-ideal cases, the optimized trajectories are close to straight lines, significantly reducing the number of required steps (in this paper, we take 20 steps). Diffusion models use nonlinear, stochastic

Table A4: **Quantitative comparison with shape morphing methods.**

Metrics	MapTree	BIM	SmoothShells	NeuroMorph	SRIF	Ours
Dirichlet ↓	17.7309	12.4723	14.0198	22.0461	6.4702	<b>4.5163</b>
Cov. ↑	0.3967	0.4665	0.6275	0.1099	0.6418	<b>0.8510</b>

trajectories requiring many steps—typically 2,000 without sampling techniques or around 200 with them—to achieve good results. Also, the reverse SDE process requires noise sampling, leading to additional computation cost.

## D Comparison with other methods

### D.1 Comparison with other textured 3D method

We compare our method with existing textured 3D morphing approaches, including MorphFlow, 3DRM, and our own. While the main paper presents qualitative comparisons with 3DRM, here we additionally provide a side-by-side qualitative comparison with MorphFlow. As shown in the Fig. A1, rows 1 and 3 display results from MorphFlow, and rows 2 and 4 show our corresponding outputs. Our method produces noticeably clearer and more accurate shapes and textures, with smoother and more coherent morphing transitions. These visual improvements are consistent with the quantitative results reported in the main paper, further corroborating the effectiveness of our approach.

Figure A1: **Qualitative comparison with MorphFlow.**

### D.2 Comparison with shape morphing methods

Although previous 3D shape morphing methods do not consider texture transformation, we provide a comparison focused solely on shape deformation. As shown in Table A4, we compare our method with several state-of-the-art shape morphing approaches, including MapTree (Ren et al., 2020), BIM (Kim et al., 2011), SmoothShells (Eisenberger et al., 2020), NeuroMorph (Eisenberger et al., 2021), and SRIF (Sun et al., 2024). Following the evaluation protocol from (Sun et al., 2024), we use the SHREC07 (Temeriac et al., 2007) dataset and report performance using Dirichlet energy (Ezuz et al., 2019) and Coverage (Huang and Ovsjanikov, 2017) metrics. Our method achieves superior results across both metrics, demonstrating more efficient and accurate shape interpolation.

For qualitative comparison, we show results against SRIF in Fig. A2, where rows 1, 3 and 5 show SRIF’s outputs and rows 2, 4 and 6 show ours. Our morphing process is smoother and preserves finer details in intermediate shapes—for example, the gecko’s toes in row 2 and the head structure in row 4. Additional comparison with NeuralMorph is presented in Fig. A3, where our results are again significantly more detailed and coherent.

To ensure that intermediate shapes remain faithful to both source and target, we introduce a shape-aware initialization. Specifically, we render both front and back views of the source and target objects and use these images as inputs to the flow model to extract an initial condition feature. This feature is then refined by minimizing the geometric difference between generated shapes and the original meshes, leading to accurate and consistent 3D representations throughout the morphing sequence.



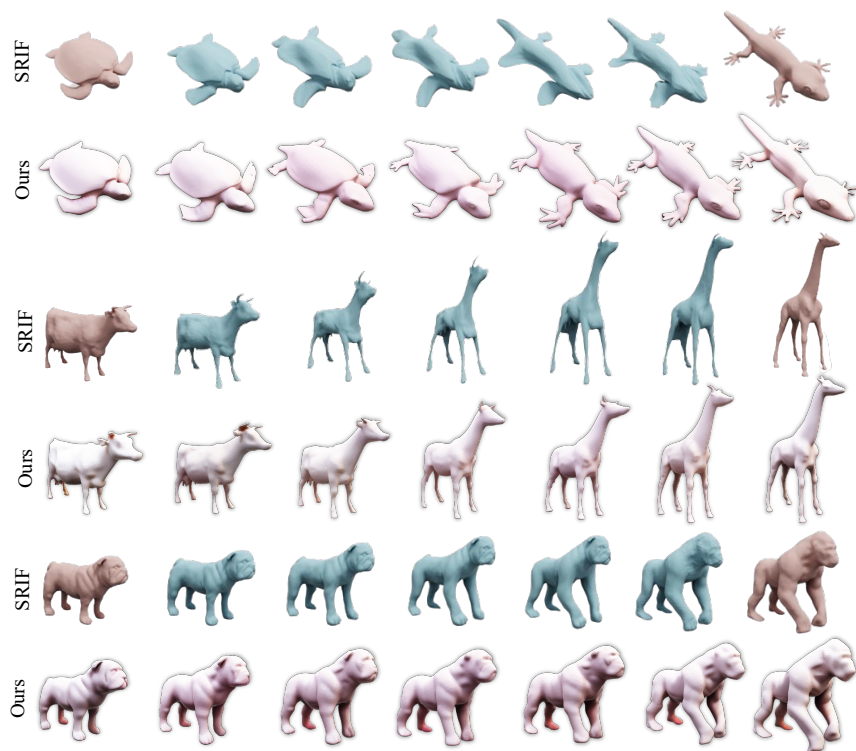


Figure A2: Qualitative comparison with SRIF.



Figure A3: Qualitative comparison with NeuralMorph.

## E More results

Fig. A4 and Fig. A5 illustrate the textured 3D morphing process generated from “Wukong” to a variety of objects. The texture can be flexibly inherited from either the source or the target image, depending on the user preference. Fig. A6 and Fig. A7 further demonstrate morphing between additional object pairs, showcasing the versatility of our method. Notably, our method is capable of performing textured 3D morphing not only between geometrically complex objects, but also across different semantic categories, highlighting its superior robustness and generalizability.

## F Broader impact

Our work introduces WUKONG, a training-free framework for high-quality textured 3D morphing, which significantly lowers the barrier to creating detailed and semantically consistent 3D transformations from simple prompts. This greatly reduces the efforts on 3D content creation for artists, designers, and educators, enabling broader access to advanced generative tools without requiring technical expertise in 3D modeling or animation. The ability to produce controllable and high-fidelity morphing sequences could benefit applications in virtual reality, digital storytelling, education, and creative industries. We hope our work inspires further research into controllable and efficient 3D generation techniques, and that it serves as a foundation for inclusive and creative applications of generative 3D content.

## G Limitation

While our method achieves state-of-the-art performance in textured 3D morphing, several limitations remain. First, like existing morphing methods, our approach still encounters difficulties on cases involving extreme topological changes, such as splitting or merging parts. These scenarios remain a general challenge in the field and are not yet fully addressed by existing methods. Second, since our method operates without explicit 3D supervision or correspondence annotations, its results may be sensitive to ambiguities in the input prompts or inconsistencies in multi-view generation, especially when the input lacks structural clarity.



Figure A4: Textured 3D morphing of Wukong (The Monkey King).



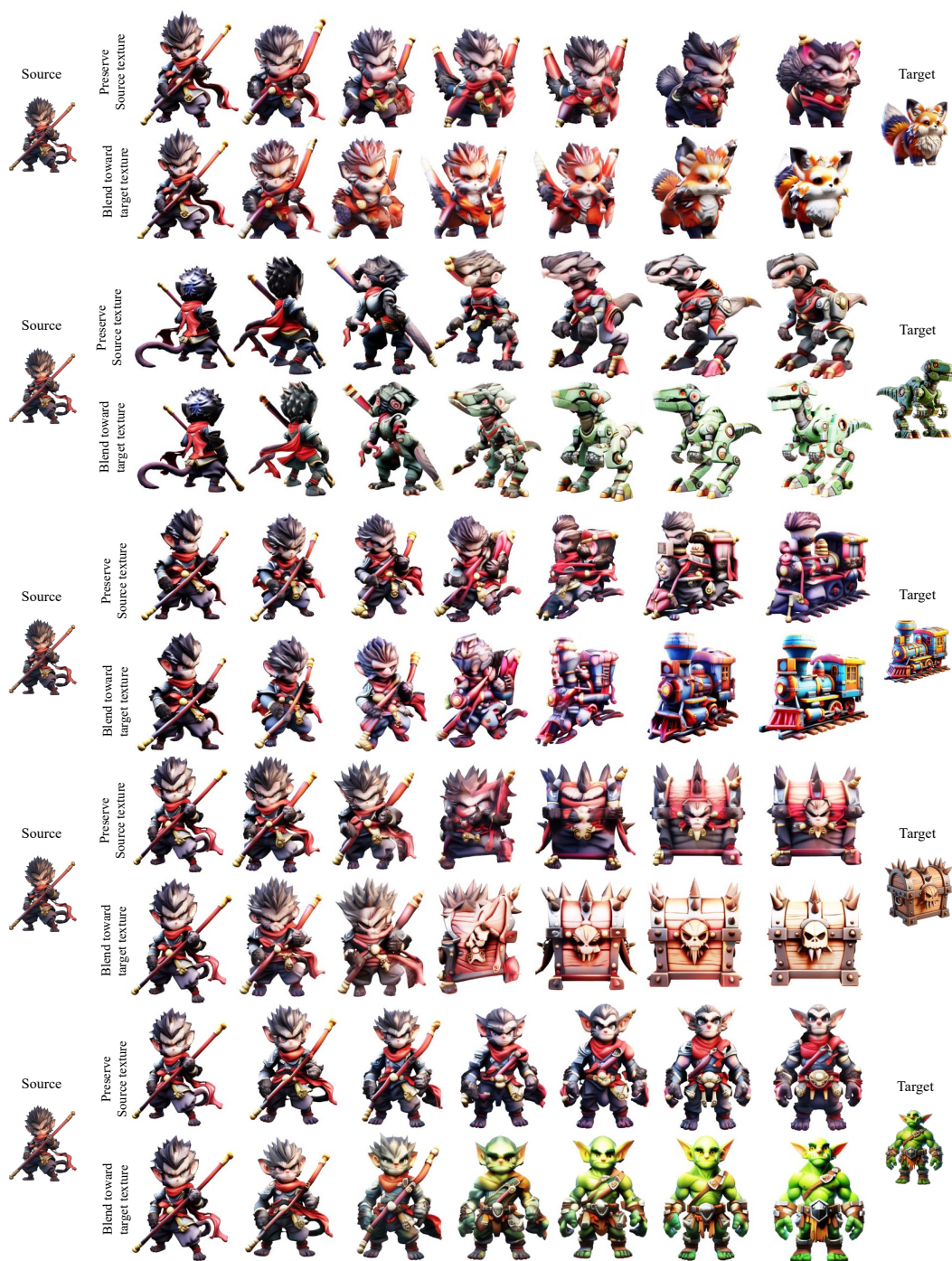


Figure A5: Textured 3D morphing of Wukong (The Monkey King).

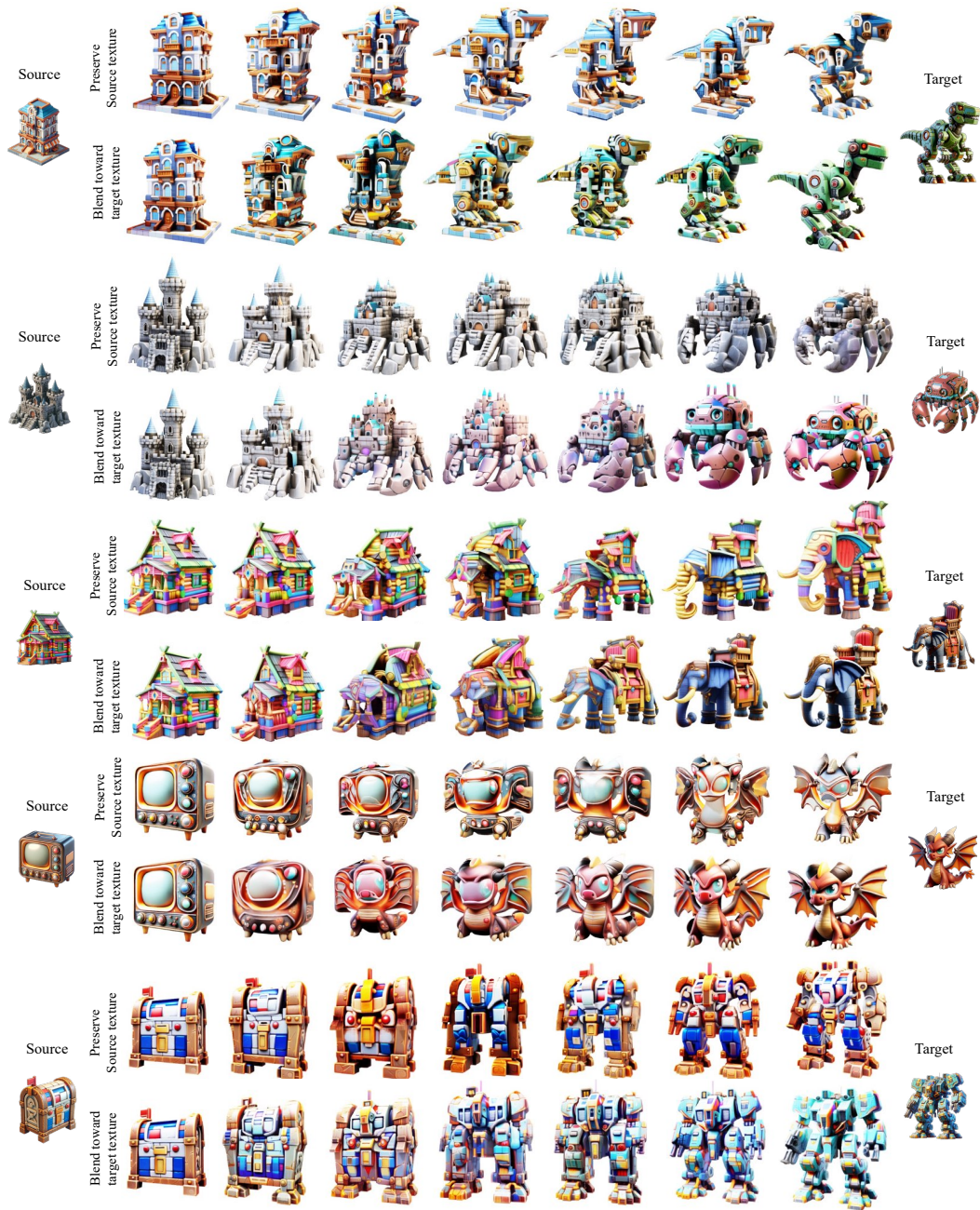


Figure A6: Textured 3D morphing of different objects.



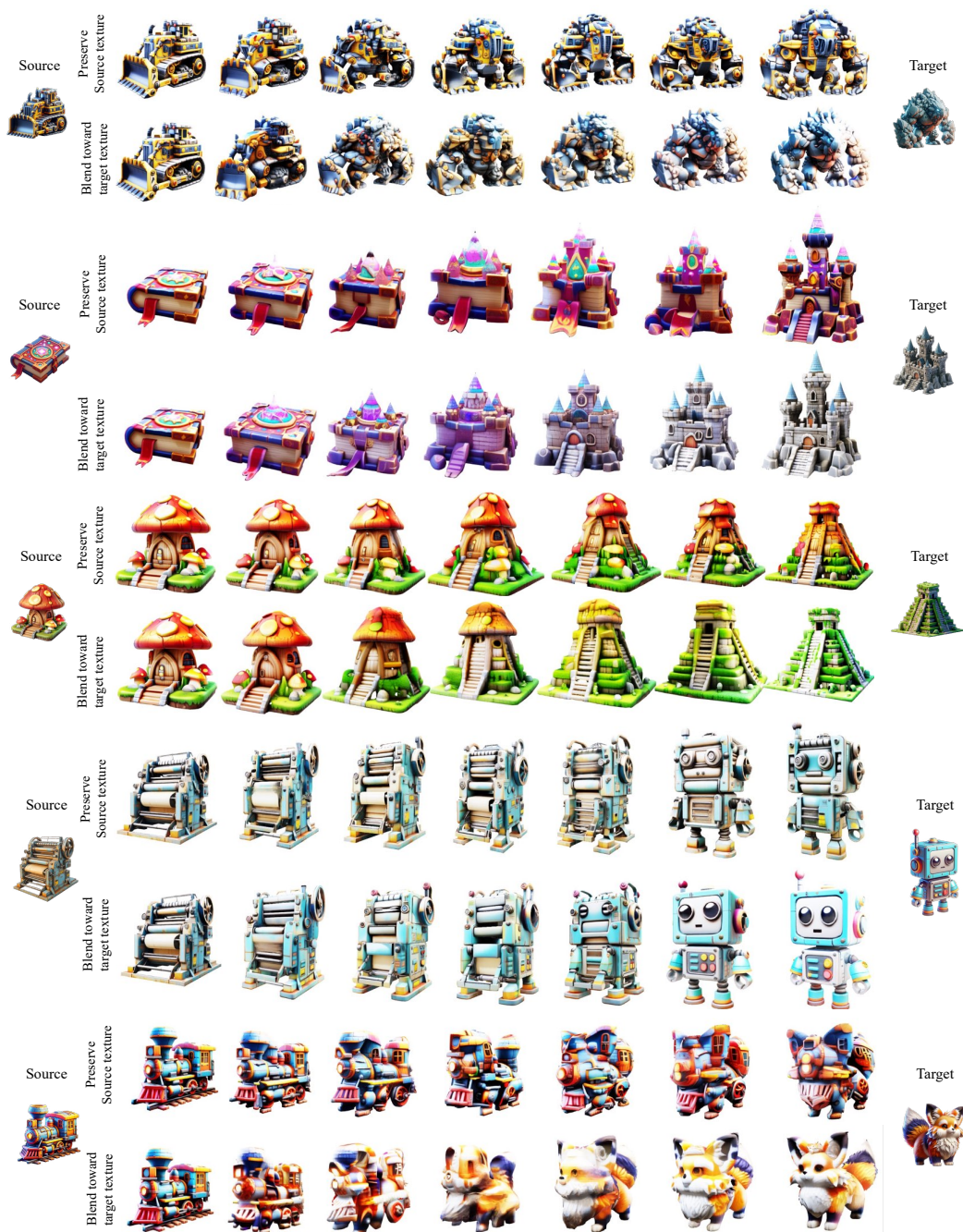


Figure A7: Textured 3D morphing of different objects.

## References

- Hang Dai, Nick Pears, William Smith, and Christian Duncan. Statistical modeling of craniofacial shape and texture. *IJCV*, 2020. 2
- Marvin Eisenberger, Zorah Lahner, and Daniel Cremers. Smooth shells: Multi-scale shape registration with functional maps. In *CVPR*, 2020. 4
- Marvin Eisenberger, David Novotny, Gael Kerchenbaum, Patrick Labatut, Natalia Neverova, Daniel Cremers, and Andrea Vedaldi. Neuromorph: Unsupervised shape interpolation and correspondence in one go. In *CVPR*, 2021. 4
- Danielle Ezuz, Justin Solomon, and Mirela Ben-Chen. Reversible harmonic maps between discrete surfaces. *TOG*, 2019. 4
- Yunhui Guo, Chaofeng Wang, Stella X Yu, Frank McKenna, and Kincho H Law. Adaln: a vision transformer for multidomain learning and predisaster building information extraction from images. *J Comput Civil Eng*, 2022. 2
- Ruqi Huang and Maks Ovsjanikov. Adjoint map representation for shape analysis and matching. In *CGF*, 2017. 4
- Vladimir G Kim, Yaron Lipman, and Thomas Funkhouser. Blended intrinsic maps. *TOG*, 2011. 4
- Johannes von Lindheim. Simple approximative algorithms for free-support wasserstein barycenters. *COA*, 2023. 2
- Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 3
- Warren S Loud. Differential equations. by an tikhonov, ab vasil’eva and ag sveshnikov. *Am Math Mon*, 1987. 3
- Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2
- Jing Ren, Simone Melzi, Maks Ovsjanikov, and Peter Wonka. Maptree: Recovering multiple solutions in the space of maps. *TOG*, 2020. 4
- Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 3
- Mingze Sun, Chen Guo, Puhua Jiang, Shiwei Mao, Yurun Chen, and Ruqi Huang. SRIF: Semantic shape registration empowered by diffusion-based image morphing and flow estimation. In *SIGGRAPH Asia*, 2024. 4
- Maja Temerinac, Marco Reisert, and Hans Burkhardt. Shrec’07-protein retrieval challenge. *SMI*, 2007. 4
- Chih-Jung Tsai, Cheng Sun, and Hwann-Tzong Chen. Multiview regenerative morphing with dual flows. In *ECCV*, 2022. 3
- Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024. 2
- Songlin Yang, Yushi Lan, Honghua Chen, and Xingang Pan. Textured 3d regenerative morphing with 3d diffusion prior. *arXiv preprint arXiv:2502.14316*, 2025. 2, 3
- LAN Yushi, Shangchen Zhou, Zhaoyang Lyu, Fangzhou Hong, Shuai Yang, Bo Dai, Xingang Pan, and Chen Change Loy. Gaussiananything: Interactive point cloud flow matching for 3d generation. In *ICLR*, 2025. 2
- Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *NeurIPS*, 2019. 2