

# AVATARGO: ZERO-SHOT 4D HUMAN-OBJECT INTERACTION GENERATION AND ANIMATION (—*Supplementary*—)

Yukang Cao<sup>1\*</sup> Liang Pan<sup>2†‡</sup> Kai Han<sup>3</sup> Kwan-Yee K. Wong<sup>3</sup> Ziwei Liu<sup>1†</sup>

<sup>1</sup>S-Lab, Nanyang Technological University, <sup>2</sup>Shanghai AI Laboratory, <sup>3</sup>The University of Hong Kong  
<https://yukangcao.github.io/AvatarGO/>

## CONTENTS

<b>A</b>	<b>Video results</b>	<b>2</b>
<b>B</b>	<b>Implementation details</b>	<b>2</b>
<b>C</b>	<b>More explanation on designing “Ours (Var-A)” and “Ours (Var-B)”</b>	<b>3</b>
<b>D</b>	<b>Training complexity</b>	<b>4</b>
<b>E</b>	<b>2D human-object interaction image generation</b>	<b>4</b>
<b>F</b>	<b>Direct rigging of 3D object and human models</b>	<b>5</b>
<b>G</b>	<b>Analysis by determining the animation of object by only the contact part</b>	<b>5</b>
<b>H</b>	<b>Comparisons with AvatarCraft, DreamWaltz and DreamAvatar</b>	<b>6</b>
<b>I</b>	<b>Societal impact.</b>	<b>6</b>
<b>J</b>	<b>Limitations.</b>	<b>6</b>
<b>K</b>	<b>More comparisons on 3D generation</b>	<b>7</b>
<b>L</b>	<b>More comparisons on 4D animation</b>	<b>8</b>

## LIST OF FIGURES

1	<b>Example generation of human-object interaction images.</b> Images generated by pose-conditioned ControlNet <a href="#">Zhang &amp; Agrawala (2023)</a> . . . . .	4
2	<b>Evaluation by directly rigging humans and objects</b> . . . . .	5
3	<b>Evaluation by using SMPL-X pose from contact human part</b> . . . . .	5
4	<b>Qualitative comparisons with DreamWaltz, AvatarCraft, and DreamAvatar</b> . .	6
5	<b>Evaluation on DreamWaltz’s animated results</b> . . . . .	6
6	<b>Comparisons on 3D compositional generations.</b> . . . . .	7
7	<b>Comparisons on 4D animation.</b> . . . . .	8

\*Part of the work has been done when interning at Shanghai AI Laboratory.

<sup>†</sup> Corresponding authors

<sup>‡</sup> Project lead

## A VIDEO RESULTS

To better visualize the generated results, we offer an improved demonstration of our method through rotated videos in the supplementary materials. To access this demonstration, please open the file named “**index.html**” provided in the supplementary.

## B IMPLEMENTATION DETAILS

Our network is built upon the official implementation of DreamGaussian4D (Ren et al., 2023) and Threestudio (Guo et al., 2023) (an open-source 3D generative project).

To ensure easy reproducibility, we first include all the hyperparameters for our 3D composition stage in Tab. 1.

Table 1: **Hyper-parameters of AvatarGO - 3D composition stage.**

<b>Camera setting</b>	Camera distance range	2.
	Radius	2.0
	Elevation range	(-30, 30)
	FoV range	49.1
<b>Render setting</b>	Resolution for 0-120 epochs	(128, 128)
	Resolution for 120-240 iters	(256, 256)
	Resolution for 240-400 iters	(512, 512)
<b>Diffusion setting</b>	Guidance scale	7.5
	$t$ range	(0.01, 0.97)
	Minimal step percent	0.01
	Maximal step percent	0.97
	$\omega(t)$	$\sqrt{\alpha_t}(1 - \alpha_t)$
<b>Initialization</b>	Rotation $\mathcal{R}$	$\text{torch.normal}(\text{mean}=[0.5, 0.5, 0.5, 0.5], \text{std}=0.1)$
	Translation $\mathcal{T}$	0.0
	Scale $\mathcal{S}$	$\text{torch.normal}(\text{mean}=1.0, \text{std}=0.3)$
<b>Learning rate</b>	Rotation $\mathcal{R}$	0.005
	Translation $\mathcal{T}$	0.005
	Scale $\mathcal{S}$	0.005
<b>LLM-guided contact retargeting</b>	threshold $a$	$1e-7$
<b>Training objectives</b>	$\lambda_{DS}^*$	1.0
<b>Hardware</b>	GPU	$1 \times \text{NVIDIA A100 (80GB)}$

Table 2: Hyper-parameters of AvatarGO - 4D animataion stage.

Camera setting	Camera distance range	2.
	Radius	2.0
	Elevation range	(-30, 30)
	FoV range	49.1
Render setting	Resolution for 0-120 epochs	(128, 128)
	Resolution for 120-240 iters	(256, 256)
	Resolution for 240-400 iters	(512, 512)
Diffusion setting to calculate $\mathcal{L}_{SDS}^*$	Guidance scale	7.5
	$t$ range	(0.01, 0.97)
	Minimal step percent	0.01
	Maximal step percent	0.97
	$\omega(t)$	$\sqrt{\alpha_t}(1 - \alpha_t)$
Diffusion setting to calculate $\mathcal{L}_{SDS}$	Guidance scale	7.5
	Guidance rescale	0.75
	$t$ range	(0.02, 0.98)
	Minimal step percent	0.02
	Maximal step percent	0.98
	gradient clip	[0, 1.5, 2.0, 1000]
	gradient clip pixel	True
	gradient clip threshold	1.0
	$\omega(t)$	$\sqrt{\alpha_t}(1 - \alpha_t)$
Initialization	Rotation $\mathcal{R}$	[-0.16, -0.16, -0.16, 0.5]
	Translation $\mathcal{T}$	0.0
Learning rate	Rotation $\mathcal{R}$	0.001
	Translation $\mathcal{T}$	0.001
Training objectives	$\lambda_{CA}$	1e+3
	$\lambda_{SDS}^*$	1.0
	$\lambda_{SDS}$	1.0
Hardware	GPU	1 $\times$ NVIDIA A100 (80GB)

In the 4D animation stage, we apply HexPlane (Cao & Johnson, 2023) to produce features from point position  $\mathbf{x}_c$  and timestamp  $\mathbf{t}$ , followed by an MLP to predict the offset for Gaussian attributes, i.e., point location  $\mathbf{x}$ , scaling matrix  $\mathbf{s}$ , rotation matrix  $\mathbf{R}$ . Specifically, the HexPlane encoder lifts the inputs to a higher frequency dimension  $F((\mathbf{x}_c, \mathbf{t})) \in \mathbb{R}^{128}$ , while the MLP is set to the default in DreamGaussian4D with ResNet (He et al., 2016).

To further ensure easy reproducibility, we first include all the hyperparameters for our 4D animation stage in Tab. 2. The other hyper-parameters are set to be the default of DreamGaussian4D (Guo et al., 2023).

## C MORE EXPLANATION ON DESIGNING “OURS (VAR-A)” AND “OURS (VAR-B)”

“Ours (Var-A)”: This is a version where we have disabled the Lang-SAM initialization in our 3D static compositional generation. Comparing this with our final method shows that without assistance from Lang-SAM, the diffusion model struggles to accurately interpret human-object images.

“Ours (Var-B)”: While Comp4D (Xu et al., 2024) separates 3D scenes into two components and applies trajectories to one component for compositional 4D generation, it leaves the other component static. This method is not suitable for our scenarios where both humans and objects are dynamic. Therefore, we design "Ours (Var-B)" by adopting the Comp4D strategy: allowing the object to follow a trajectory while the human moves independently. Specifically, we replace our correspondence-aware motion supervision, as defined in Eq. ??, with SDS supervision strategy via the video diffusion model used in Comp4D. Comparing this approach with our final method demonstrates that our correspondence-aware motion supervision more effectively preserves the relationship between humans and objects throughout the animation process.

## D TRAINING COMPLEXITY

In our study, our results, detailed in both the main paper and the Appendix, involve training the 3D stage for 400 epochs on a single NVIDIA A100 GPU, taking approximately 10 minutes. Similarly, the 4D stage requires roughly 20 minutes of training on the same GPU. To compare with other methods: 1) In the experiments for 3D compositional generation, HumanGaussian (Liu et al., 2023) demands approximately 2 hours to complete 3600 epochs; GraphDreamer (Gao et al., 2023) adopts a two-stage training approach, with the coarse stage taking roughly 3 hours for 10000 epochs and the fine stage requiring around 6 hours for 20000 epochs. 2) Additionally, in our experiments with 4D animation, DreamGaussian4D (Ren et al., 2023) completes training of their 3-stage network in around 10 minutes; TC4D (Bahmani et al., 2024) demands approximately 1 hour for the first stage over 10000 epochs, 3 hours for the second stage over 20000 epochs, and roughly 30 hours for the third stage over 30000 epochs.

## E 2D HUMAN-OBJECT INTERACTION IMAGE GENERATION

Because of the limited availability of human-object interaction images within the 2D dataset utilized for training diffusion models, existing models encounter challenges in accurately capturing the spatial dynamics and contact between humans and objects. This limitation is evident in Figure 1, where we noticed that during the process of 2D image generation, the diffusion model would struggle to create such images. This inadequacy significantly hampers the ability of diffusion models to generate realistic 3D human-object interactions.



Figure 1: **Example generation of human-object interaction images.** Images generated by pose-conditioned ControlNet Zhang & Agrawala (2023)

## F DIRECT RIGGING OF 3D OBJECT AND HUMAN MODELS

We conducted experiments by directly positioning the 3D objects in a reasonable position relative to the humans. As shown in Fig. 2, without further adjustments such as rescaling or rotating, the relationships between humans and objects are not accurately depicted. Penetration issues will also exist in some examples. Even with manual adjustments, such as rescaling and rotating the 3D objects, significant human effort is required, and the interactions between humans and objects still lack accuracy. For instance, Fig. 2 illustrates that humans frequently appear with open hands, which fails to convincingly "hold" the objects and significantly undermines the user experience.

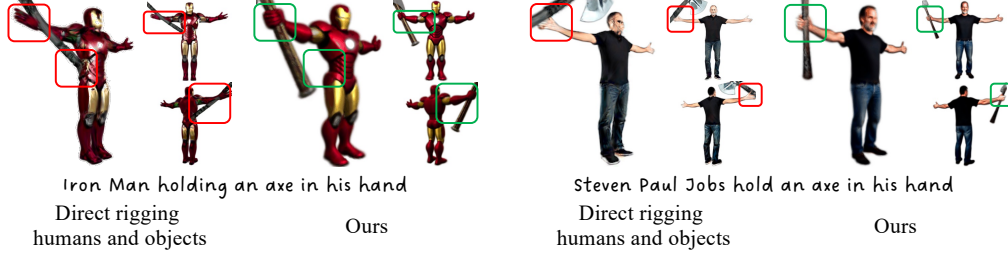


Figure 2: Evaluation by directly rigging humans and objects

## G ANALYSIS BY DETERMINING THE ANIMATION OF OBJECT BY ONLY THE CONTACT PART

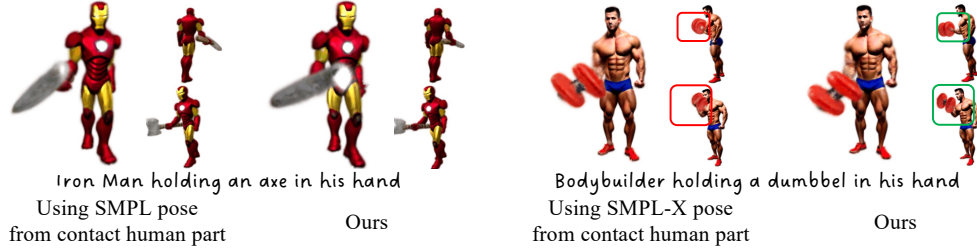


Figure 3: Evaluation by using SMPL-X pose from contact human part

We conducted experiments using the contact part of the human body to determine the object's motion. The results are shown in Fig. 3. We found that this approach works well when the object is positioned far from the body, but it can encounter penetration issues when the object is close to the body (see "Bodybuilder holding a dumbbell in his hand"). We will incorporate this discussion into the updated paper.

## H COMPARISONS WITH AVATARCRAFT, DREAMWALTZ AND DREAMAVATAR

In Fig. 4, we provide qualitative comparisons with AvatarCraft, DreamWaltz, and DreamAvatar. We observed that AvatarCraft and DreamAvatar are highly constrained by the SMPL prior model, making it difficult for them to create human models with effective object interactions. While DreamWaltz can generate some object interactions, these interactions are often inaccurate. Additionally, DreamWaltz has trouble maintaining proper interactions throughout the animation, as presented in Fig. 5.



Figure 4: **Qualitative comparisons with DreamWaltz, AvatarCraft, and DreamAvatar**



Figure 5: **Evaluation on DreamWaltz's animated results**

## I SOCIETAL IMPACT.

The progress in 4D avatar generation with object interactions holds promise for numerous AR/VR applications, yet also raises concerns regarding potential misuse, such as creating misleading or nonexistent human-object pairings. We advocate for responsible research and deployment, promoting openness and transparency in practices to mitigate any potential negative consequences.

## J LIMITATIONS.

While opening new doors for human-centric 4D content generation, we acknowledge AvatarGO has certain limitations: 1) Our pipeline operates under the assumption of rigid-body dynamics for 3D objects, making it unsuitable for animating non-rigid content such as flags; 2) our method presumes that continuous contact between objects and avatars, making it challenges for tasks like "Dribbling the basketball," where the human and object inevitably disconnect at certain points. Nevertheless, our current approach does not cover all possible scenarios, it effectively handles continuous contact and rigid connections, which are commonly encountered in real-world applications.

## K MORE COMPARISONS ON 3D GENERATION

We provide more qualitative comparisons with HumanGaussian (Liu et al., 2023), GraphDreamer (Gao et al., 2023), and “Ours (Var-A)” in Fig. 6. These results serve to reinforce the claims made in Sec. ?? of the main paper, providing further evidence of the superior performance of AvatarGO in compositing 3D human and object models.

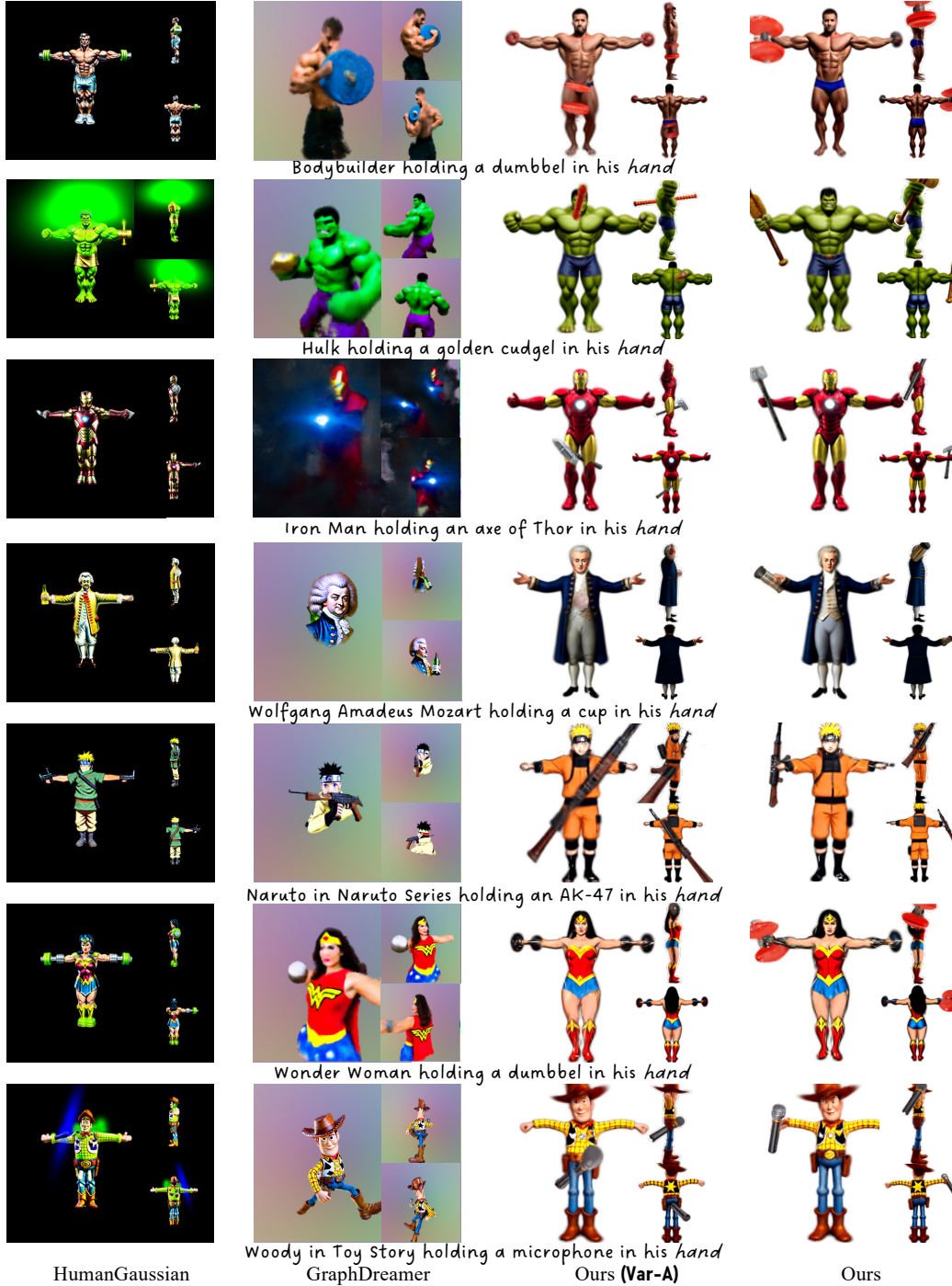


Figure 6: Comparisons on 3D compositional generations.



## L MORE COMPARISONS ON 4D ANIMATION

We further provide more qualitative comparisons of 4D animation with DreamGaussian4D (Ren et al., 2023), HumanGaussian (Liu et al., 2023), and ‘Ours (Var-B)’. The results can be found in Fig. 7. These comparisons further demonstrate the superiority of AvatarGO in maintaining the spatial correlation during animations and in addressing the penetration issues.



Figure 7: Comparisons on 4D animation.



## REFERENCES

- Sherwin Bahmani, Xian Liu, Yifan Wang, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. *arXiv preprint arXiv:2403.17920*, 2024. 4
- Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 130–141, 2023. 3
- Gege Gao, Weiyang Liu, Anpei Chen, Andreas Geiger, and Bernhard Schölkopf. Graphdreamer: Compositional 3d scene synthesis from scene graphs. *arXiv preprint arXiv:2312.00093*, 2023. 4, 7
- Yuan-Chen Guo, Ying-Tian Liu, Ruizhi Shao, Christian Laforte, Vikram Voleti, Guan Luo, Chia-Hao Chen, Zi-Xin Zou, Chen Wang, Yan-Pei Cao, and Song-Hai Zhang. threestudio: A unified framework for 3d content generation. <https://github.com/threestudio-project/threestudio>, 2023. 2, 3
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 3
- Xian Liu, Xiaohang Zhan, Jiaxiang Tang, Ying Shan, Gang Zeng, Dahua Lin, Xihui Liu, and Ziwei Liu. Humangaussian: Text-driven 3d human generation with gaussian splatting. *arXiv preprint arXiv:2311.17061*, 2023. 4, 7, 8
- Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dream-gaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023. 2, 4, 8
- Dejia Xu, Hanwen Liang, Neel P Bhatt, Hezhen Hu, Hanxue Liang, Konstantinos N Platanotis, and Zhangyang Wang. Comp4d: Llm-guided compositional 4d scene generation. *arXiv preprint arXiv:2403.16993*, 2024. 3
- Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 1, 4