
Supplementary Material

3DOT: Texture Transfer for 3DGS Objects from a Single Reference Image

Xiao Cao¹ Beibei Lin¹ Bo Wang² Zhiyong Huang¹ Robby T. Tan^{1,3}

¹ National University of Singapore ² University of Mississippi ³ ASUS Intelligent Cloud Services

{xiaocao, beibei.lin}@u.nus.edu

hawk.rsrch@gmail.com {dcshuang, robby.tan}@nus.edu.sg

1 Implementation Details

We employ a depth-based ControlNet [9] architecture as our diffusion backbone, following the approach used in Gaussctrl [8], which also utilizes Gaussian splatting training logic from NeRFStudio [7]. For each editing task, we select unedited views at every fourth frame from the full training set, and perform image editing, which maintains performance while also ensuring efficiency in terms of time cost. The inference step for the view editing process should be set within the range [300, 600]. We choose 500 steps for generation process for complicated texture transfer and 400 steps for color and material transfer. One advantage of our method is that our 3D-OTT is not sensitive to denoising steps thanks to the reference set and prompt-tuning guidance. The VRAM for Gaussian splatting training is around 12GB, and the VRAM for the progressive process is within 24GB. All experiments are conducted on a single Nvidia L40s.

2 Detailed quantitative results

We show detailed quantitative results in Table 2, 3, 4, 5, 6, 7, 8, 9, 10. For Vision-GPT score, we prompt GPT4-o with *"Now I am going to provide you with 6 pictures A B C D E F, you need to evaluate A B C D E who is more similar to the texture of F and score out of 100. Please focus on the texture of the objects in the pictures. The images will be provided in order. You should also give me reason of each scores."*

One sample of generated answer (*ice frozen horse* case) is:

- **Image A:** This image has a simpler texture and a predominantly bluish tint to it, and is closer in color to Image F, but lacks the glowing effect. **Score: 65/100**
- **Image B:** This image is closer to A in texture and color, and again lacks the dynamic effects of the F image's light and shadow. **Rating: 60/100**
- **Image C:** While there is more pronounced detail, the colors are on the natural tone side and differ from the glowing blue texture of the F image. **Score: 50/100**
- **Image D:** The texture is complex and varied, but the color differs greatly from F and is not a blue shade. **Score: 40/100**
- **Image E:** Sharp color, contains blue and noticeable glowing effects, closest to F in texture. **Rating: 85/100**

For user studies, we inform users to rate images from 1 to 5. The images generated by different baselines are disordered and anonymised. They can rate with the same score for different images in the same editing scenario.



Figure 1: Ablation study on Equation 8.

3 Discussion on 2D Method

In this section, we explore more 2D diffusion finetuning methods of texture transferring while preserving objects' identities as shown in Fig.2. DreamBooth [6] preserves the reference color and retains some aspects of material quality, whereas Textual Inversion [3] only restores an imprecise approximation of the reference color. Furthermore, neither method can control the 3D observation position or viewing angle of the object. We further explore the effect of directly encoding the style or object concept into a special text token without gradient guidance, as shown in Fig. 3. We further explore the effect of finetuning a single text token following both Textual Inversion's style encoding method and object encoding method. As shown in The prompt of the two cases is "cartoon bear", and the reference image converted into latent space is the reference image in the main paper *cartoon bear* edits. The failure of those 2D finetuning methods indicates that a single image alone is insufficient for encoding texture characteristics into a token. Our approach, which employs prompt-tuning-based gradient guidance (i.e., encoding differences in texture characteristics with gradient guidance), offers a more effective solution.

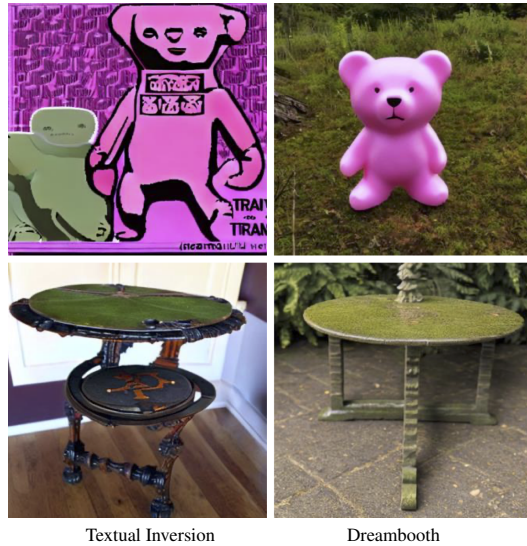


Figure 2: Experiments on using DreamBooth and Textual Inversion to finetune diffusion model in *pink plastic bear* and *moss table* case. Generated results show that single reference image finetuning is not sufficient to preserve object identity.



Figure 3: Experiments on 2D finetuning method: Textural Inversion fails to encode the specific object or style into a token with only one training image (i.e., reference image).

Evaluation Metric / $w_T : w_{T'} : w_R$	0:0:5	2.5:2.5:2.5	2.5:2.5:5	2.5:5:2.5	0:5:0
Clip Score \uparrow	0.7407	0.8447	0.8648	0.8651	0.8795
Lpips (VGG) \downarrow	0.3093	0.3038	0.3157	0.3114	0.2933
Lpips (Alex) \downarrow	0.2999	0.2921	0.3020	0.2960	0.2894
Evaluation Metric / $w_T : w_{T'} : w_R$	2.5:5:2.5	5:0:0	5:2:1	5:2.5:2.5	5:5:5
Clip Score \uparrow	0.8651	0.8570	0.9125	0.8259	0.8528
Lpips (VGG) \downarrow	0.3114	0.2971	0.2841	0.3159	0.3377
Lpips (Alex) \downarrow	0.2960	0.2916	0.2890	0.2997	0.3191

Table 1: Ablation study on Equation 13.

Lpips (Alex)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	0.115	0.085	0.142	0.125	0.038
Golden bear	0.162	0.191	0.158	0.167	0.111
White table	0.046	0.059	0.075	0.046	0.047
Moss table	0.096	0.140	0.124	0.133	0.094
Ice horse	0.102	0.092	0.097	0.124	0.056
Fire horse	0.145	0.119	0.128	0.128	0.081
Lego dinosaur	0.226	0.236	0.206	0.204	0.123
Wooden dinosaur	0.113	0.134	0.123	0.160	0.095
Hawk	0.313	0.272	0.295	0.281	0.237
Hulk	0.390	0.355	0.344	0.345	0.284

Table 2: Detailed Alex-based LPIPS score

Lpips (VGG)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	0.113	0.095	0.137	0.117	0.073
Golden bear	0.150	0.172	0.148	0.147	0.116
White table	0.076	0.073	0.088	0.073	0.068
Moss table	0.104	0.119	0.110	0.109	0.096
Ice horse	0.082	0.068	0.072	0.088	0.056
Fire horse	0.127	0.102	0.113	0.112	0.085
Lego dinosaur	0.205	0.204	0.173	0.173	0.122
Wooden dinosaur	0.113	0.114	0.115	0.138	0.092
Hawk	0.307	0.277	0.292	0.283	0.250
Hulk	0.399	0.370	0.343	0.363	0.289

Table 3: Detailed VGG-based LPIPS score

4 Preliminaries

4.1 Gaussian Splatting

3D Gaussian Splatting [5] (3DGS) was recently proposed for novel view synthesis and is well-known for its efficiency and precision. In 3DGS, the object can be represented by a set of 3D Gaussian ellipses $\mathcal{G} = \{g_i(\mathbf{x} \mid (\sigma_i, \mu_i, \Sigma_i, c_i))\}$ where g_i is given as Equation 1, $\sigma \in \mathbb{R}_0^+$ refers to opacity, $\mu_k \in \mathbb{R}^{3 \times 1}$ refers to Gaussian center, $\Sigma \in \mathbb{R}^{3 \times 3}$ refers to the covariance matrix, and c refers to the color distribution of each Gaussian ellipse.

$$g_i(\mathbf{x} \mid (\sigma_i, \mu_i, \Sigma_i, c_i)) = e^{(-\frac{1}{2}(\mathbf{x}-\mu_i)^\top \Sigma_i^{-1}(\mathbf{x}-\mu_i))}. \quad (1)$$

The covariance matrices Σ_i , due to their positive semi-definite definition, can be formulated as shown in Equation 2:

$$\Sigma_i = R_i S_i S_i^T R_i^T, \quad (2)$$

where S_i refers to scaling matrix and R_i^T refers to rotation matrix.

Given to be rendered pixel x , the color can be obtained by alpha blending of explicitly stored Gaussian ellipses as Equation 3:

$$c(x) = \sum_{i \in N} c_i \alpha_i \mathbf{G}_i^{2D}(x) \prod_{j=1}^{i-1} (1 - \alpha_j \mathbf{G}_j^{2D}(x)), \quad (3)$$

where $\mathbf{G}_i^{2D}(x)$ refers to the sorted 2D Gaussians related with pixel x and $c_i \in \mathbb{R}$ are corresponding coefficients [1].

4.2 Diffusion with Classifier-free Guidance

In the diffusion model, the image classifier is introduced to better trade off mode coverage and sample fidelity by using estimated classifier gradient [2]. To make this process compact, classifier-free guidance technique is introduced [4]. They use the single neural network to parametrize conditional diffusion (i.e., with classifier) and unconditional diffusion (i.e., without classifier) and jointly train it by randomly setting the class instruction \mathbf{c} to be unconditional class identifier (i.e., null or negative prompt) $\mathbf{c} = \emptyset$.

The sampling is performed by linearly combining conditional and unconditional score estimates as Equation 4:

$$\epsilon_\theta(\mathbf{z}_\lambda, \mathbf{c}) = (1 + w)\tilde{\epsilon}_\theta(\mathbf{z}_\lambda, \mathbf{c}) - w\tilde{\epsilon}_\theta(\mathbf{z}_\lambda, \emptyset), \quad (4)$$

where ϵ_θ refers to the final noise, and $\tilde{\epsilon}_\theta$ refers to the noise predicted with the associated condition. This sampling process can be further rewritten Equation 5 in practice:

$$\epsilon_\theta(\mathbf{z}_\lambda, \mathbf{c}) = \tilde{\epsilon}_\theta(\mathbf{z}_\lambda, \emptyset) + w(\tilde{\epsilon}_\theta(\mathbf{z}_\lambda, \mathbf{c}) - \tilde{\epsilon}_\theta(\mathbf{z}_\lambda, \emptyset)). \quad (5)$$

The rewritten formulation can be interpreted as using the unconditional result as starting point enhanced with the text prompt feature guidance scaled by factor w and the prompt feature is extracted by the simple subtraction process.

5 Ablation Studies on Equation 8

We conduct ablation studies to better assess the effect of $w'_\mathbb{T}$, $w_\mathbb{T}$, and $w_\mathbb{R}$ in Equation 8, as shown in Table 1 and Figure 1. We notice that a large $w_\mathbb{R}$ value leads to a smooth object; large $w'_\mathbb{T}$ leads to over-saturation; $w_\mathbb{T}$ is responsible for maintaining object identity. Hence, we choose the values of $w_\mathbb{T} = 5$, $w'_\mathbb{T} = 2$, and $w_\mathbb{R} = 1$ for balanced performance.

Clip score	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	0.913	0.914	0.875	0.896	0.926
Golden bear	<u>0.860</u>	0.850	0.900	0.854	0.939
White table	0.957	0.944	<u>0.922</u>	0.919	0.968
Moss table	0.953	0.922	0.924	0.938	0.963
Ice horse	0.892	0.920	0.910	0.850	0.922
Fire horse	0.834	<u>0.913</u>	0.817	0.849	0.957
Lego dinosaur	0.910	0.876	0.921	0.894	0.949
Wooden dinosaur	0.935	0.910	<u>0.925</u>	0.850	0.9593
Hawk	0.790	0.760	0.752	0.660	0.837
Hulk	0.873	0.899	0.692	0.862	0.913

Table 4: Detailed CLIP scores under each scene

Vision-GPT	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	30	45	25	55	75
Golden bear	30	50	40	45	55
White table	30	<u>40</u>	50	55	55
Moss table	70	60	<u>40</u>	50	75
Ice horse	65	60	50	40	85
Fire horse	<u>30</u>	55	40	45	80
Lego dinosaur	30	<u>25</u>	40	45	85
Wooden dinosaur	70	60	80	55	85
Hawk	40	50	65	70	80
Hulk	60	75	<u>50</u>	<u>80</u>	85

Table 5: Vision-GPT scores under each scene

User study (#1)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	3	3	1	2	5
Golden bear	<u>2</u>	<u>2</u>	4	1	5
White table	2	2	<u>4</u>	3	4
Moss table	3	2	<u>1</u>	1	4
Ice horse	3	2	1	1	4
Fire horse	<u>1</u>	3	1	1	4
Lego dinosaur	2	<u>1</u>	1	1	4
Wooden dinosaur	1	1	1	1	4
Hawk	1	<u>2</u>	<u>2</u>	2	4
Hulk	2	4	<u>1</u>	<u>2</u>	3

Table 6: User study (user#1)

User study (#2)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	3	3	1	2	5
Golden bear	<u>2</u>	<u>2</u>	3	3	5
White table	1	1	<u>3</u>	<u>2</u>	5
Moss table	3	2	<u>1</u>	1	5
Ice horse	<u>3</u>	3	2	1	5
Fire horse	<u>1</u>	<u>3</u>	2	2	5
Lego dinosaur	1	<u>1</u>	2	2	4
Wooden dinosaur	2	2	<u>2</u>	<u>2</u>	5
Hawk	1	2	4	3	5
Hulk	2	4	<u>1</u>	3	5

Table 7: User study (user#2)

User study (#3)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	3	4	2	3	5
Golden bear	2	$\bar{2}$	4	4	5
White table	4	3	$\bar{1}$	$\bar{1}$	5
Moss table	$\bar{1}$	4	3	1	5
Ice horse	1	3	4	1	5
Fire horse	1	4	$\bar{2}$	1	5
Lego dinosaur	1	$\bar{1}$	1	1	3
Wooden dinosaur	2	3	1	1	5
Hawk	1	2	4	3	5
Hulk	2	4	$\bar{1}$	3	5

Table 8: User study (user#3)

User study (#4)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	3	3	1	2	5
Golden bear	$\bar{2}$	$\bar{1}$	4	1	5
White table	4	4	$\bar{1}$	1	4
Moss table	4	3	1	1	$\bar{4}$
Ice horse	2	3	2	1	5
Fire horse	1	$\bar{3}$	1	1	5
Lego dinosaur	1	$\bar{1}$	2	1	5
Wooden dinosaur	1	2	1	1	5
Hawk	1	2	4	3	5
Hulk	2	4	$\bar{1}$	3	5

Table 9: User study (user#4)

User study (#5)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	2	2	3	4	5
Golden bear	2	1	4	$\bar{4}$	5
White table	2	1	$\bar{4}$	$\bar{3}$	4
Moss table	3	2	$\bar{3}$	3	4
Ice horse	$\bar{3}$	2	2	4	4
Fire horse	$\bar{3}$	2	2	4	4
Lego dinosaur	$\bar{3}$	2	2	4	5
Wooden dinosaur	$\bar{2}$	3	3	$\bar{4}$	4
Hawk	1	2	4	3	5
Hulk	2	4	$\bar{1}$	3	5

Table 10: User study (user#5)

User study (#6)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	3	3	1	2	5
Golden bear	$\bar{2}$	$\bar{2}$	3	2	4
White table	1	1	$\bar{3}$	2	5
Moss table	3	2	$\bar{1}$	1	5
Ice horse	$\bar{3}$	3	2	1	5
Fire horse	$\bar{1}$	$\bar{3}$	2	2	5
Lego dinosaur	1	$\bar{1}$	2	2	4
Wooden dinosaur	2	2	$\bar{2}$	$\bar{2}$	5
Hawk	1	2	4	2	5
Hulk	2	4	$\bar{1}$	2	5

Table 11: User study (user#6)

User study (#7)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	4	3	1	2	5
Golden bear	<u>2</u>	2	3	2	4
White table	1	2	<u>3</u>	2	3
Moss table	2	2	<u>1</u>	1	5
Ice horse	2	3	2	1	5
Fire horse	1	<u>3</u>	2	2	4
Lego dinosaur	1	<u>1</u>	2	2	4
Wooden dinosaur	2	2	<u>2</u>	—	
ul2	4	—	—		
Hawk	1	2	3	2	5
Hulk	2	3	<u>1</u>	2	4

Table 12: User study (user#7)

User study (#8)	IN2N	IGS2GS	Gaussctrl	DGE	Ours
Pink bear	3	3	1	2	4
Golden bear	<u>2</u>	<u>2</u>	3	2	4
White table	1	2	<u>3</u>	2	4
Moss table	3	2	<u>1</u>	1	5
Ice horse	3	3	1	1	5
Fire horse	<u>2</u>	<u>3</u>	2	2	5
Lego dinosaur	2	<u>2</u>	2	2	5
Wooden dinosaur	<u>2</u>	<u>2</u>	<u>2</u>	—	
ul2	5	—	—		
Hawk	3	2	3	2	5
Hulk	2	3	<u>2</u>	2	5

Table 13: User study (user#8)

References

- [1] Minghao Chen, Iro Laina, and Andrea Vedaldi. Dge: Direct gaussian 3d editing by consistent multi-view editing. *arXiv preprint arXiv:2404.18929*, 2024.
- [2] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.
- [3] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.
- [4] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.
- [5] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.
- [6] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023.
- [7] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, et al. Nerfstudio: A modular framework for neural radiance field development. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–12, 2023.
- [8] Jing Wu, Jia-Wang Bian, Xinghui Li, Guangrun Wang, Ian Reid, Philip Torr, and Victor Adrian Prisacariu. Gaussctrl: multi-view consistent text-driven 3d gaussian splatting editing. *arXiv preprint arXiv:2403.08733*, 2024.
- [9] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.