PODIA-3D: Domain Adaptation of 3D Generative Model Across Large Domain Gap Using Pose-Preserved Text-to-Image Diffusion (Supplementary Material)

Gwanghyun Kim[1] Ji Ha Jang[1] Se Young Chun[1,2,†]
[1]Dept. of Electrical and Computer Engineering, [2]INMC & IPAI
Seoul National University, Republic of Korea
{gwang.kim, jeeit17, sychun}@snu.ac.kr

A. Videos

We provide supplementary videos that presents a more comprehensive visualization and demonstration of the effectiveness of our PODIA-3D method in adapting 3D generators across significant domain gaps at gwang-kim.github.io/podia_3d. The videos showcases the high level of text-image correspondence achieved as well as the high quality of 3D shapes, achieved by our approach.

B. Details on Methods

B.1. SD vs DGD vs PPD

Stable diffusion (SD) [28] is a latent-based text-to-image diffusion model. This model is composed of frozen VQGAN [4] encoder & decoder and a noise prediction model $\boldsymbol{\epsilon}_\phi^{\text{SD}}$, which is conditioned on time and text. The VQGAN encoder [4] E^V encodes an image $\boldsymbol{x}$ into a latent vector $\boldsymbol{q}_0 = E^V(\boldsymbol{x})$ and the decoder D^V converts the latent to the reconstructed image $\hat{\boldsymbol{x}} = D^V(\boldsymbol{q}_0)$. To train the noise prediction model $\boldsymbol{\epsilon}_\phi$, the latent $\boldsymbol{q}_0$ is first perturbed into $\boldsymbol{q}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{q}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$ through the forward diffusion [8], where $\boldsymbol{\epsilon} \sim \mathcal{N}(0,1)$, $t \sim \mathcal{U}([1,T])$, $\bar{\alpha}_t$ is noise schedule, and T is the total diffusion steps. In SD, T is set to $1,000$. Then, the noise prediction model $\boldsymbol{\epsilon}_\phi^{\text{SD}}$ is trained to predict the noise $\boldsymbol{\epsilon}$ included in $\boldsymbol{q}_t$, given $\boldsymbol{q}_t, t$ and the text prompt y representing $\boldsymbol{x}$, using following objective:

$$\mathbb{E}_{\boldsymbol{x},y,\boldsymbol{\epsilon},t}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\phi^{\text{SD}}(\boldsymbol{q}_t, y, t)\|_2^2],$$

where $(\boldsymbol{x}, y)$ are image-text pairs. In the noise prediction model $\boldsymbol{\epsilon}_\phi^{\text{SD}}$, the text prompt y is encoded to the text embedding through the CLIP [24] text encoder and the time t is converted to the time feature using the Fourier feature [34]. To enable classifier-free guidance [9], a single diffusion model is trained on conditional and unconditional objectives by randomly dropping y to $\varnothing$. The noisy latent $\boldsymbol{q}_t$ is given to the input of the first convolution layer of the UNet-based [29] autoencoder while the time feature and the text embedding are conditioned on the normalization layers through cross-attention [35] mechanism.

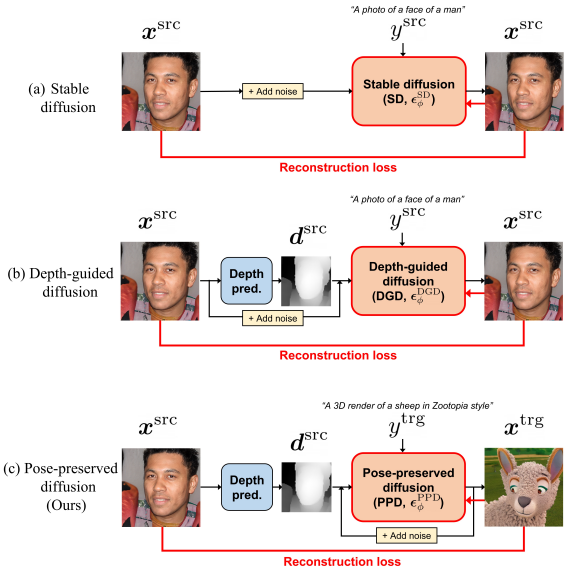

Figure S1. Comparison of training process for 3 different diffusion models: (a) Stable diffusion [28], (b) depth-guided diffusion [28] and (c) our pose-preserved diffusion.

Depth-guided diffusion model (DGD) [28] is a variant of SD [28], where the depth map $\boldsymbol{d} = h(\boldsymbol{x})$ is predicted from $\boldsymbol{x}$ using the pretrained depth estimation model h and concatenated with $\boldsymbol{q}_t$, and used as input to the UNet [29] autoencoder as an extra conditioning. The DGD noise prediction model $\boldsymbol{\epsilon}_\phi^{\text{DGD}}$ is fine-tuned from the pretrained $\boldsymbol{\epsilon}_\phi^{\text{SD}}$ using following objective:

$$\mathbb{E}_{\boldsymbol{x},y,\boldsymbol{\epsilon},t}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\phi^{\text{DGD}}(\boldsymbol{q}_t, y, \boldsymbol{d}, t)\|_2^2],$$

†Corresponding author.

Algorithm 1: Text-guided image-to-image translation (T-I2I)

Input: $\boldsymbol{\epsilon}_\phi \in \{\boldsymbol{\epsilon}_\phi^{\text{SD}}, \boldsymbol{\epsilon}_\phi^{\text{DGD}}, \boldsymbol{\epsilon}_\phi^{\text{PPD}}\}$, E^V, D^V, y, t_r, s, $*$

Function `T_I2I` ($\boldsymbol{x}^{\text{src}}$, y, $\boldsymbol{\epsilon}_\phi$, $*$) **:**

- $\boldsymbol{q}_0 = E^V(\boldsymbol{x}^{\text{src}})$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
- $\boldsymbol{q}_{t_r}^{\text{trg}} = \sqrt{\bar{\alpha}_{t_r}}\boldsymbol{q}_0 + \sqrt{1-\bar{\alpha}_{t_r}}\boldsymbol{\epsilon}$
- **if** $\boldsymbol{\epsilon}_\phi \in \{\boldsymbol{\epsilon}_\phi^{\text{DGD}}, \boldsymbol{\epsilon}_\phi^{\text{PPD}}\}$ **then**
  - $\boldsymbol{d} = h(\boldsymbol{x}^{\text{src}})$
- **for** $t = t_r, t_r - 1, \ldots, 1$ **do**
  - **if** $\boldsymbol{\epsilon}_\phi \in \{\boldsymbol{\epsilon}_\phi^{\text{DGD}}, \boldsymbol{\epsilon}_\phi^{\text{PPD}}\}$ **then**
    - $\tilde{\boldsymbol{\epsilon}} = s\boldsymbol{\epsilon}_\phi(\boldsymbol{q}_t^{\text{trg}}, y, \boldsymbol{d}, t) \setminus + (1-s)\boldsymbol{\epsilon}_\phi(\boldsymbol{q}_t^{\text{trg}}, \varnothing, \boldsymbol{d}, t)$
  - **else**
    - $\tilde{\boldsymbol{\epsilon}} = s\boldsymbol{\epsilon}_\phi(\boldsymbol{q}_t^{\text{trg}}, y, t) + (1-s)\boldsymbol{\epsilon}_\phi(\boldsymbol{q}_t^{\text{trg}}, \varnothing, t)$
  - $\boldsymbol{q}_{t-1}^{\text{trg}} = \text{Sampling}(\boldsymbol{q}_t^{\text{trg}}, \tilde{\boldsymbol{\epsilon}}, t)$
- $\boldsymbol{x}^{\text{trg}} = D^V(\boldsymbol{q}_0^{\text{trg}})$
- **return** $\boldsymbol{x}^{\text{trg}}$

where $(\boldsymbol{x}, y)$ are image-text pairs. DGD is fine-tuned from the pretrained SD.

Our pose-preserving diffusion model (PPD) has the same architecture as DGD, but it is trained differently. It uses the depth map $\boldsymbol{d}^{\text{src}} = h(\boldsymbol{x}^{\text{src}})$ from the source image $\boldsymbol{x}^{\text{src}}$, but the latent $\boldsymbol{q}_0^{\text{trg}} = E^V(\boldsymbol{x}^{\text{trg}})$ from the target image $\boldsymbol{x}^{\text{trg}}$, giving the noisy latent $\boldsymbol{q}_t^{\text{trg}} = \sqrt{\bar{\alpha}_t}\boldsymbol{q}_0^{\text{trg}} + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$. The PPD noise prediction model $\boldsymbol{\epsilon}_\phi^{\text{PPD}}$ is fine-tuned from the pretrained $\boldsymbol{\epsilon}_\phi^{\text{DGD}}$ using following objective:

$$\mathbb{E}_{\boldsymbol{x}^{\text{src}}, \boldsymbol{x}^{\text{trg}}, y^{\text{trg}}, \boldsymbol{\epsilon}, t}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\phi^{\text{PPD}}(\boldsymbol{q}_t^{\text{trg}}, y^{\text{trg}}, \boldsymbol{d}^{\text{src}}, t)\|_2^2],$$

where $(\boldsymbol{x}^{\text{src}}, \boldsymbol{x}^{\text{trg}}, y_{\text{trg}})$ are a set of the source image, the target image and the target text.

Text-guided image-to-image translation (T-I2I) is a technique that combines diffusion-based image-to-image translation [21] with text-to-image diffusion models [28, 25, 30]. The generalized T-I2I algorithm for $\boldsymbol{\epsilon}_\phi^{\text{SD}}, \boldsymbol{\epsilon}_\phi^{\text{DGD}}$ or $\boldsymbol{\epsilon}_\phi^{\text{PPD}}$ is described in Algorithm 1. Initially, we convert $\boldsymbol{x}^{\text{src}}$ to $\boldsymbol{q}_0 = E^V(\boldsymbol{x}^{\text{src}})$ using the VQGAN encoder [4] E^V and perturb it to generate $\boldsymbol{q}_{t_r}^{\text{trg}}$ where $t_r \in [1, T]$ is the return step. Next, $\boldsymbol{q}_0^{trg}$ is obtained from the noisy latent $\boldsymbol{q}_{t_r}^{\text{trg}}$ by proceeding the sampling process using $\boldsymbol{\epsilon}_\phi \in \{\boldsymbol{\epsilon}_\phi^{\text{SD}}, \boldsymbol{\epsilon}_\phi^{\text{DGD}}, \boldsymbol{\epsilon}_\phi^{\text{PPD}}\}$. The guidance scale s is used to adjust the scale of gradients from the target prompt y and the empty prompt $\varnothing$. Finally, we can obtain the target image $\boldsymbol{x}^{\text{trg}} = D^V(\boldsymbol{q}_0^{trg})$ using the VQGAN decoder [4] D^V. Any sampling method such as DDPM [8], DDIM [32], PLMS [20] can be used.

The comparison of the training process for each model is shown in Fig. S1.

B.2. Specialized-to-general sampling

In the initial ηT phase, with $\eta \in [0, 1]$ representing the PPD ratio, we leverage the PPD model to produce major structural components and pose information. For the remaining $(1-\eta)T$ duration, we employ Stable diffusion models to generate smaller image details or structures. The specialized-to-general sampling process is outlined in Algorithm 2.

Algorithm 2: Specialized-to-general sampling

Input: $\boldsymbol{\epsilon}_\phi^{\text{SD}}, \boldsymbol{\epsilon}_{\phi'}^{\text{PPD}}$, E^V, D^V, y, t_r, η, s,h $*$

Function `S_to_G` ($\boldsymbol{x}^{\text{src}}$, y, $\boldsymbol{\epsilon}_\phi^{\text{SD}}, \boldsymbol{\epsilon}_{\phi'}^{\text{PPD}}$, $*$) **:**

- $\boldsymbol{q}_0 = E^V(\boldsymbol{x}^{\text{src}})$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
- $\boldsymbol{q}_{t_r}^{\text{trg}} = \sqrt{\bar{\alpha}_{t_r}}\boldsymbol{q}_0 + \sqrt{1-\bar{\alpha}_{t_r}}\boldsymbol{\epsilon}$
- $\boldsymbol{d} = h(\boldsymbol{x}^{\text{src}})$
- **for** $t = t_r, t_r - 1, \ldots, 1$ **do**
  - **if** $t_r > (1-\eta)T$ **then**
    - $\tilde{\boldsymbol{\epsilon}} = s\boldsymbol{\epsilon}_{\phi'}^{\text{PPD}}(\boldsymbol{q}_t^{\text{trg}}, y, \boldsymbol{d}, t) \setminus + (1-s)\boldsymbol{\epsilon}_{\phi'}^{\text{PPD}}(\boldsymbol{q}_t^{\text{trg}}, \varnothing, \boldsymbol{d}, t)$
  - **else**
    - $\tilde{\boldsymbol{\epsilon}} = s\boldsymbol{\epsilon}_\phi^{\text{SD}}(\boldsymbol{q}_t^{\text{trg}}, y, t) + (1-s)\boldsymbol{\epsilon}_\phi^{\text{SD}}(\boldsymbol{q}_t^{\text{trg}}, \varnothing, t)$
  - $\boldsymbol{q}_{t-1}^{\text{trg}} = \text{Sampling}(\boldsymbol{q}_t^{\text{trg}}, \tilde{\boldsymbol{\epsilon}}, t)$
- $\boldsymbol{x}^{\text{trg}} = D^V(\boldsymbol{q}_0^{\text{trg}})$
- **return** $\boldsymbol{x}^{\text{trg}}$

B.3. Text-guided adaptation of 3D generative models

We utilize the total loss $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ADA}} + \lambda\mathcal{L}_{\text{den}}$, which includes both the ADA loss and the density regularization loss. Here are the details on the losses used for adversarially fine-tuning the 3D generator following the fine-tuning process in [16, 2].

ADA loss. The ADA loss, $\mathcal{L}_{\text{ADA}}$ [12], is an adversarial loss that incorporates adaptive dual augmentation and R1 regularization. Specifically, the loss is defined as follows:

$$\mathcal{L}_{\text{ADA}} = \mathbb{E}_{\boldsymbol{z}\sim\mathcal{Z}, \boldsymbol{c}\sim\mathcal{C}}[f(D_\omega(A(G_\theta(\boldsymbol{z}, \boldsymbol{c})), \boldsymbol{c})] \tag{S1}$$
$$+ \mathbb{E}_{(\boldsymbol{c}, \boldsymbol{x}^{\text{tda}})\in\mathcal{D}^{\text{tda}}}[f(-D_\omega(A(\boldsymbol{x}^{\text{tda}}), \boldsymbol{c}) + \lambda\|\nabla D_\omega(A(\boldsymbol{x}^{\text{tda}}), \boldsymbol{c})\|^2)]$$

where A is a stochastic non-leaking augmentation operator with probability p and $f(u) = -\log(1 + \exp(-u))$.

Density regularization loss. We also employ density regularization, which has been proven to be useful in mitigating the occurrence of undesired shape distortions by encouraging smoothness in the density field [2]. For each rendered scene, we randomly select points v from the volume $\mathcal{V}$, as well as additional perturbed points that have been slightly distorted by Gaussian noise δv. We then compute the L1 loss between the predicted densities, as shown below:

$$\mathcal{L}_{\text{den}} = \mathbb{E}_{v\in\mathcal{V}}[\|\sigma_\theta(v) - \sigma_\theta(v + \delta v)\|]. \tag{S2}$$

Algorithm for our text-guided adaptation of 3D generative models is provided in Algorithm 3. We translate the source

Algorithm 3: Text-guided adaptation of 3D generative models

Input: G_θ, $\mathcal{D}^{\text{src}}$, $\boldsymbol{\epsilon}_\phi^{\text{SD}}$, $\boldsymbol{\epsilon}_{\phi'}^{\text{PPD}}$,$K$, A, f, $*$
Output: $G_{\theta'}$

// Pose-aware target dataset generation
$\mathcal{D}^{\text{tda}} = \{\}$
for $i = 1, 2, \ldots, N$ **do**
  $(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{src}}) \in \mathcal{D}^{\text{src}}$
  $\boldsymbol{x}_i^{\text{tda}}$=S_to_G($\boldsymbol{x}_i^{\text{src}}$, y, $\boldsymbol{\epsilon}_\phi^{\text{SD}}$, $\boldsymbol{\epsilon}_{\phi'}^{\text{PPD}}$, $*$)
  Append $(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{tda}})$ to $\mathcal{D}^{\text{tda}}$

// Adversarial fine-tuning of 3D generators
$G_{\theta'} \leftarrow \text{clone}(G_\theta)$, $D_\omega \leftarrow$ Initialize D
for $k = 1, 2, \ldots, K$ **do**
  for $i = 1, 2, \ldots, N$ **do**
    $\boldsymbol{z}_i \in \mathcal{Z}$, $\boldsymbol{c}_i \in \mathcal{C}$, $v_i \in \mathcal{V}$
    // Update $G_{\theta'}$
    $\mathcal{L}_{\text{ADA}}^{G} = -f(D_\omega(A(G_{\theta'}(\boldsymbol{z}_i, \boldsymbol{c}_i)), \boldsymbol{c}_i)$
    $\mathcal{L}_{\text{den}} = \|\sigma_{\theta'}(v_i) - \sigma_{\theta'}(v_i + \delta v_i)\|$
    $\theta' \leftarrow \text{Update\_}G(\theta', \mathcal{L}_{\text{ADA}}^{G} + \lambda_{\text{den}}\mathcal{L}_{\text{den}})$
    // Update D_ω
    $\mathcal{L}_{\text{ADA}}^{D,\text{fake}} = f(D_\omega(A(G_{\theta'}(\boldsymbol{z}_i, \boldsymbol{c}_i)), \boldsymbol{c}_i)$
    $(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{tda}}) \in \mathcal{D}^{\text{tda}}$
    $\mathcal{L}_{\text{ADA}}^{D,\text{real}} = f(-D_\omega(A(\boldsymbol{x}^{\text{tda}}), \boldsymbol{c}_i)$
      $+\lambda\|\nabla D_\omega(A(\boldsymbol{x}^{\text{tda}}), \boldsymbol{c}_i)\|^2)$
    $\omega \leftarrow \text{Update\_}D(\omega, \mathcal{L}_{\text{ADA}}^{D,\text{fake}} + \mathcal{L}_{\text{ADA}}^{D,\text{real}})$

image $\boldsymbol{x}^{\text{src}}$ to yield the target image $\boldsymbol{x}^{\text{tda}}$ guided by a text prompt y using PPD and specialized-to-general sampling, constructing a set of $(\boldsymbol{c}, \boldsymbol{x}^{\text{tda}})$. Initially, we duplicate the pre-trained 3D generator, G_θ, to create $G_{\theta'}$. We also initialize a pose-conditioned discriminator, D_ω. For $i = 1, 2, ..., N$, we start by sampling a random latent vector $\boldsymbol{z}$ and camera parameter $\boldsymbol{c}$. We then compute the ADA loss for the generator, denoted as $\mathcal{L}^G$ADA, using the generator $G_{\theta'}(\boldsymbol{z}_i, \boldsymbol{c}_i)$, the discriminator D_ω, and the stochastic non-leaking augmentation A. Additionally, for each rendered scene, we randomly select points v from the volume $\mathcal{V}$ to calculate the density regularization loss $\mathcal{L}_{\text{den}}$, along with another loss. These losses are then used to update the generator. Subsequently, we calculate the ADA losses for the discriminator using both generated images, denoted as $\mathcal{L}_{\text{ADA}}^{D,\text{fake}}$, and real target images, denoted as $\mathcal{L}_{\text{ADA}}^{D,\text{real}}$. These two losses are then combined to update the discriminator. We repeat this process for K epochs.

C. Implementation Details

C.1. 3D generative model

We select EG3D [2], which is a state-of-the-art 3D-aware generative model, as our source generator. Especially, we adopt EG3D [2] pretrained on 512^2 images in FFHQ [14]. The EG3D generator is comprised of four main components: a backbone, a decoder, a volume rendering module, and a super-resolution component. The backbone features the StyleGAN2 [15] generator and a mapping network. The decoder, on the other hand, is a multilayer perceptron that has a single hidden layer. The super-resolution module, which utilizes two StyleGAN2 blocks, is also integrated into the generator. Lastly, the EG3D discriminator is based on a StyleGAN2 discriminator that features dual discrimination and camera pose-conditioning.

C.2. Text-to-image diffusion models

We utilized Stable diffusion v2 [28], which comprises of several components such as the VQGAN encoder and decoder [4], UNet-based autoencoder [29], and OpenCLIP-ViT/H text encoder [24, 31, 11]. To train this model, we used a filtered subset of LAION-5B [31]. Specifically, the model underwent training for 550k steps on 256^2 images, 1,000k steps on 512^2 images, and 140k steps on 768^2 images.

We utilize the depth-guided diffusion model [28] for our experiments. It is trained for 550k steps on 256^2 images and for 850k steps on 512^2 images in a filtered subset of LAION-5B [31], without the use of a depth map. To incorporate depth information, we modify the UNet autoencoder [29] by adding an additional input channel to receive the depth map generated by MiDaS [26, 27]. The depth-guided diffusion model is fine-tuned for 200k steps on 512^2 images in the same dataset with the updated architecture.

We utilize DEIS [42] as our diffusion sampling method, which is a state-of-the-art method that accelerates the diffusion process while maintaining high quality. We configure the number of inference steps to 30 and set the return step t_r and guidance scale s to 990 and 10, respectively.

C.3. Depth estimation network

We utilize MiDaS [26, 27] as our depth map estimation model, which computes relative inverse depth from a single image. This transformer-based model is trained on 12 distinct datasets, including WSVD [36], TartanAir [38], IRS [37], KITTI [6], MegaDepth [19], ApolloScape [10], HRWSI [40], ReDWeb [39], DIML [17], Movies, BlendedMVS [41], and NYU Depth V2 [22]. The model is trained using multi-objective optimization to ensure high quality on a wide range of inputs.

C.4. Details on training of PPD

We generated 3,000 source images and corresponding depth maps using the pre-trained EG3D [2] model on FFHQ [14] and MiDaS [26, 27]. For the target images, we first created 3,000 identity-mixed images and then generated an additional $3,000 \times 5$ images using text-guided

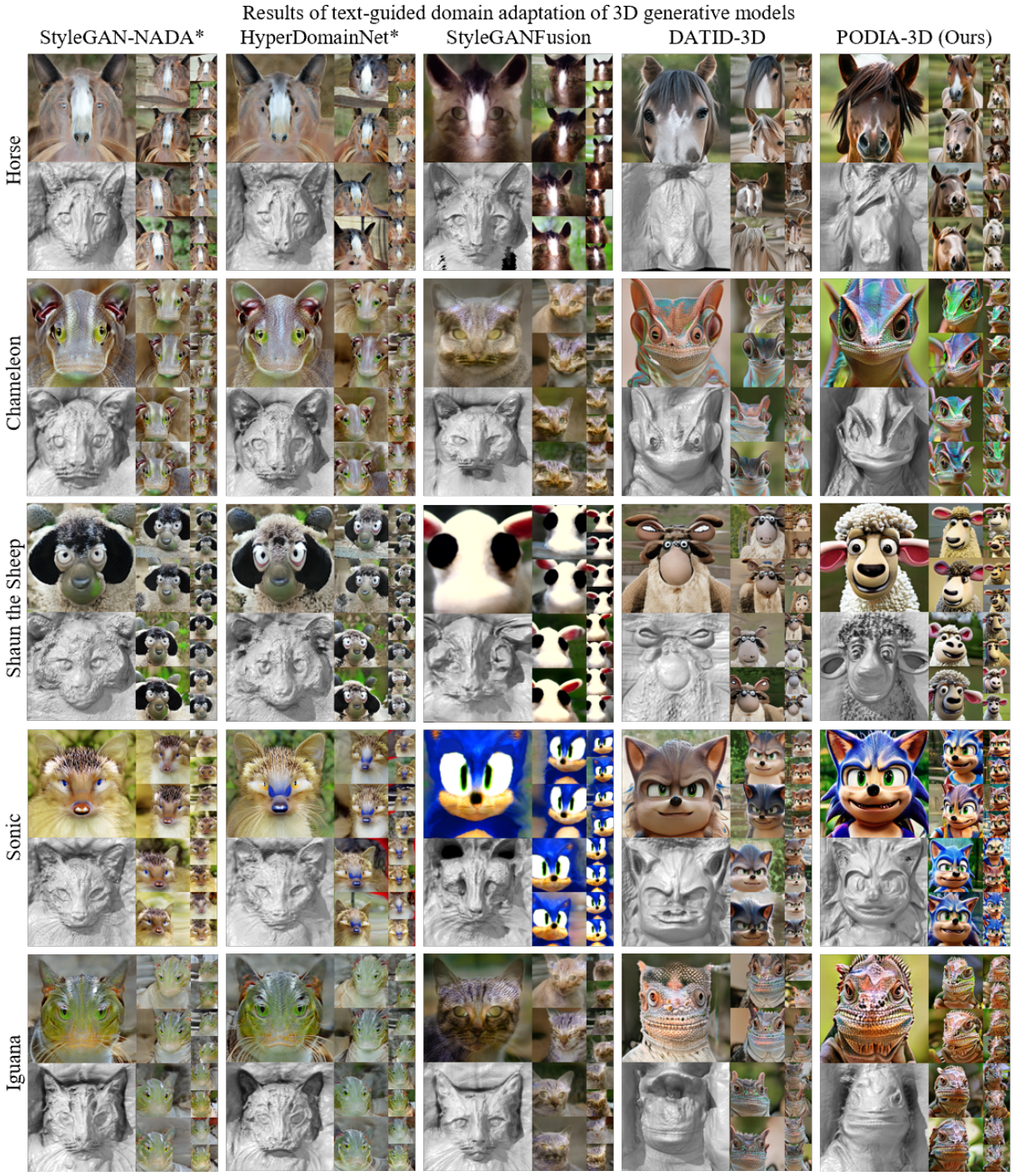

Figure S2. Qualitative comparison between our text-guided domain adaptation results and other baselines. We enable EG3D [2] to synthesize the multi-view consistent images in wide range of text-guided domains (animals & characters) with high text-image correspondence and high quality of 3D shape. For the results of pose-controlled synthesis, please see the supplementary videos.

image-to-image translation with 9 text prompts that guarantee pose preservation. To further increase the variety of target images, we collected 3,000 images using the EG3D model pre-trained on AFHQ-cat [13, 3] and translated them

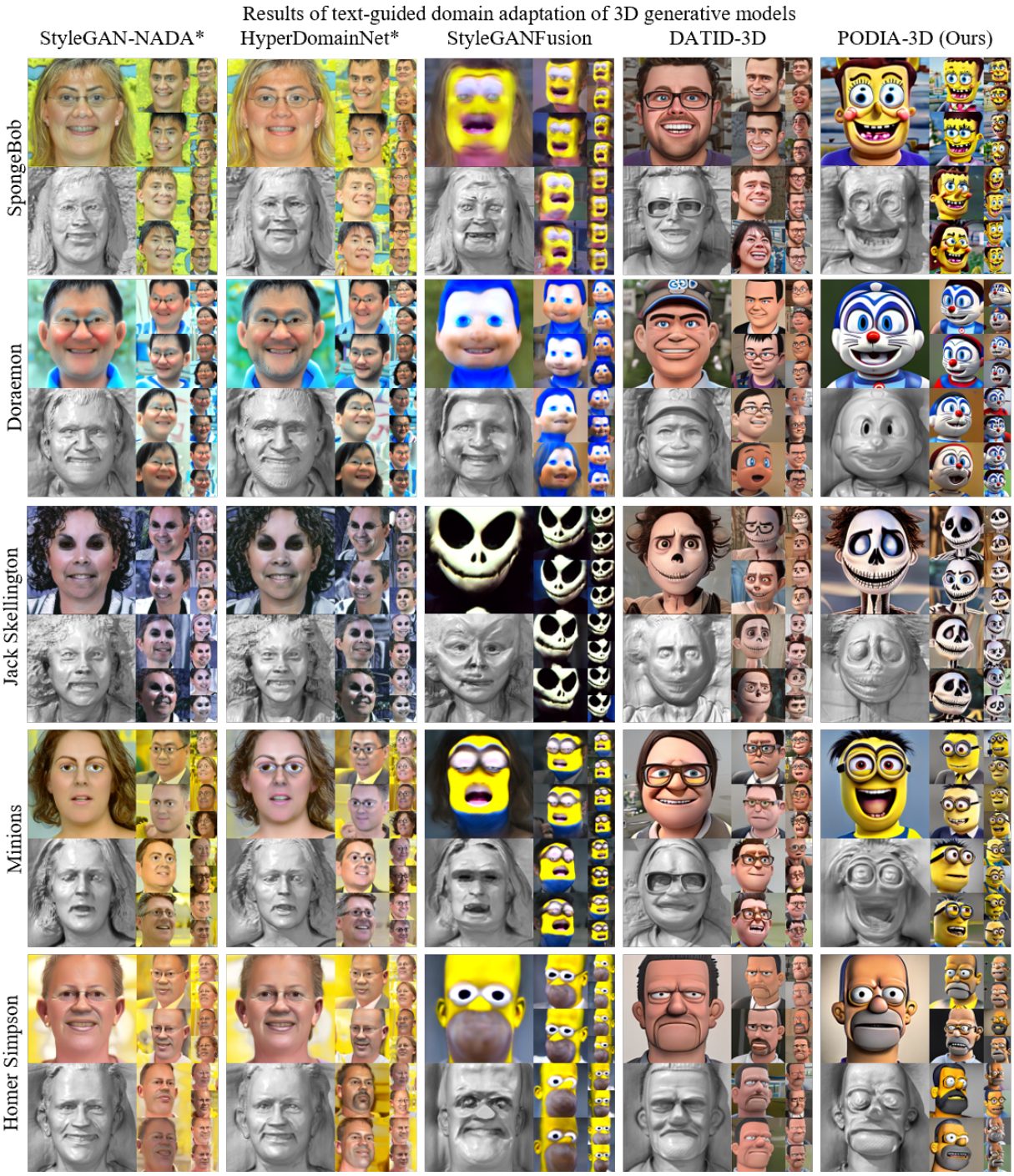

Figure S3. Qualitative comparison between our text-guided domain adaptation results and other baselines. We enable EG3D [2] to synthesize the multi-view consistent images in wide range of text-guided domains (characters) with high text-image correspondence and high quality of 3D shape. For the results of pose-controlled synthesis, please see the supplementary videos.

using 6 prompts, resulting in a total of $3,000 \times 3$ target images.

We conduct fine-tuning of the depth-guided diffusion models on this dataset for 2 epochs using Adam opti-

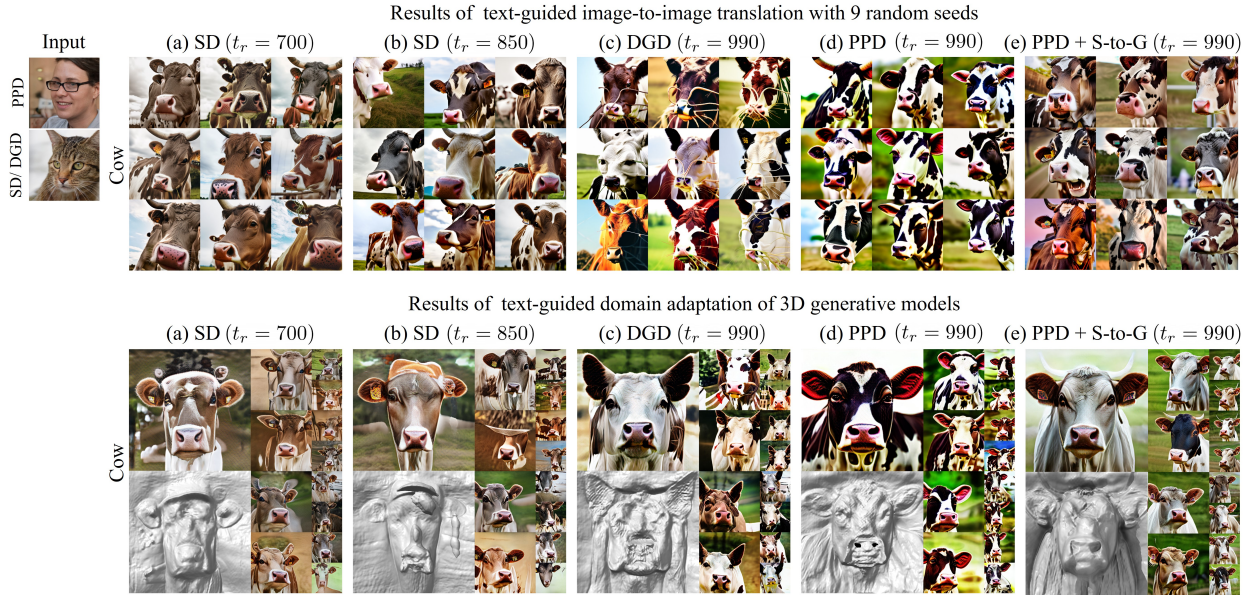

Figure S4. Results of text-guided image-to-image translation and text-guided domain adaptation of EG3D [2] using Stable diffusion (SD) [28], depth-guided diffusion (DGD) [28], our pose-preserved diffusion (PPD), and specialized-to-general (S-to-G). Pose-preserved diffusion enables image translation with pose-consistency and domain adaptation with high-quality of 3D shapes. S-to-G allows to resolve the bias issue in details.

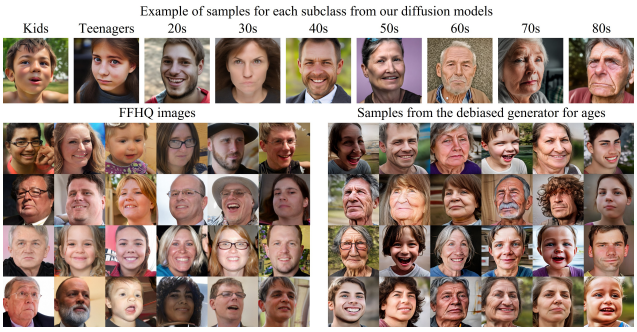

Figure S5. Results of debiasing human face images for ages.

mizer [18] and a batch size of 2 until the models have been trained on 50k∼100k images. The entire process takes approximately 15 hours using a single NVIDIA RTX 3090. Once trained, the models can be applied to any text prompt.

C.5. Details on text-guided adaptation of 3D generative model

We fine-tune the 3D generative models by using a batch size of 20 and training them until they have been exposed to 50k∼100k images. Both the generator and discriminator are optimized using Adam [18] with a learning rate of 0.002. During training, we apply image blurring with progressively diminishing degree as the input to the discriminator, as suggested by [13, 2], and we do not use style mixing. We employ ADA loss combined with R1 regularization ($\lambda = 5$) and add

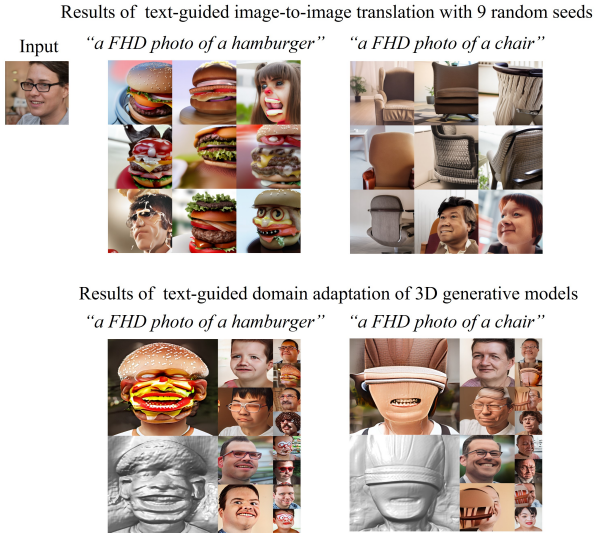

Figure S6. Domain adaptation to certain non-living objects such as chairs and hamburgers, which lack directional information, results in low text-image correspondence.

a density regularization term with strength $\lambda_{\text{den}} = 0.25$.

Using EG3D and PPD with a DEIS [42] scheduler and 30 inference steps, it takes 1.8 seconds to translate and save one image on a single NVIDIA A100. We generated 1,000 target images for each text prompt in most experiments, taking an average of 30 minutes for text-guided target

dataset generation. We update the 3D generator with a batch size of 20 using 50k∼100k training images depending on the target text prompt, which takes 1.5∼3 hours for each prompt. Unlike the DATID-3D [16] pipeline, our method does not require time-consuming CLIP-based filtering process and pose reconstruction-based filtering process that take 30 minutes∼1 hour additionally, making our pipeline more efficient and simplified.

C.6. Details on specialized-to-general sampling

Thus, we set the PPD ratio to 0.4 to preserve the pose as well as improve details. We will include details in the revision.

C.7. 3D shape visualization

We utilize marching cubes to obtain iso-surfaces from the density field using marching cubes, as described in [2]. After that, We use UCSF Chimerax [7] to visualize the 3D surfaces.

D. Experimental Details

D.1. Text prompts

We construct the prompts by combining domain-specifying words (e.g. Dog) with preset prompts (e.g. a FHD photo of) following previous works for text-guided domain adaptation [1, 5, 16]. Throughout the main paper and its supplementary materials, we employ a brief text prompt to denote each individual text prompt.

Text prompts for training PPD. For images from identity mixing and a different domain generator, we create target domain prompts y^{trg} using specifying words representing source domains (e.g. human). For images from T-I2I with pose-guaranteed prompts $y^{\text{p-guar}}$, we use $y^{\text{p-guar}}$ as y^{trg}. We chose the prompts that guarantee pose-consistency after the text-guided translation based on our observation. Specifically, we started with the text prompts used in the previous works [1, 5, 16] and two evaluators carefully assess the results of the translation in terms of pose consistency and text-image correspondence. We also empirically select new prompts following the same evaluation process. Table S1 lists complete text prompts used to collect the target images for training our pose-preserved diffusion model.

Text prompts for the results. The complete text prompts used for the results to each concise prompt are summarized in Tablee S2.

Text prompts for text-guided debiasing. We obtained the information on subclasses from ChatGPT. Then, we make the subclass text $y_i^{\text{trg},\mathcal{A}}$ by replacing superclass specifying word in superclass text y^{trg} with subclass specifying word. The complete text prompts that correspond to each subclass prompt used for text-guided debiasing can be found in Table S3.

D.2. Baselines

In StyleGAN-NADA* that is a extended version of StyleGAN-NADA [5] to 3D generative model, we update the EG3D model [2] using the directional CLIP loss as follows:

$$\mathcal{L}_{\text{dir-CLIP}} = 1 - \frac{\langle \Delta I, \Delta T \rangle}{\|\Delta I\| \|\Delta T\|}, \tag{S3}$$

where $\Delta I = E_I^C(\boldsymbol{x}^{\text{gen}}) - E_I^C(\boldsymbol{x}^{\text{src}}), \Delta T = E_T^C(y^{\text{tar}}) - E_T^C(y^{\text{src}})$. We follow the instructions provided in the paper and implement the loss and optimization part using the official StyleGAN-NADA codebase [5]. To obtain the best results, we performed early stopping during total 2,000 iterations of model training.

In HyperDomainNet*, a 3D extension of HyperDomainNet [1], an indomain angle consistency loss is included in addition to the directional CLIP loss to maintain CLIP similarities between images before and after domain adaptation.

$$\mathcal{L}_{\text{indomain}} = \sum_{i,j}^{n} (\langle E_I^C(\boldsymbol{x}_i^{\text{gen}}), E_I^C(\boldsymbol{x}_j^{\text{gen}}) \rangle - \langle E_I^C(\boldsymbol{x}_i^{\text{src}}), E_I^C(\boldsymbol{x}_j^{\text{src}}) \rangle)^2, \tag{S4}$$

We follow the instructions provided in the paper and implement the loss and optimization part using the official HyperDomainNet codebase [1]. To obtain the best results, we performed early stopping during total 2,000 iterations of model training.

StyleGANFusion [33] adopts the SDS loss [23] to guide the text-guided adaptation of 2D and 3D generators using text-image-diffusion models, as defined:

$$\mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\epsilon}[||\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t, y, t) - \boldsymbol{\epsilon}||_2^2] \tag{S5}$$

where $\boldsymbol{x}$ is a perturbed image or latent thorugh forward diffusion, $\boldsymbol{\epsilon} \sim \mathcal{N}(0,1)$, $t \sim \mathcal{U}([1,T])$, and T is the total diffusion steps. We employ LPIPs regularization [43] and fine-tune the EG3D model [2] for 2,000 iterations following the instructions in the paper.

In DATID-3D [16], pose-aware target dataset is generated using Stable diffusion [28]. Then, the target dataset is refined through CLIP-based filtering process and pose reconstruction-based filtering process. After that, EG3D model [2] is fine-tuned on the filtered dataset. We faithfully follow the implementation details in the paper. We generate 3,000 target images per target prompts and fine-tune the pretrained EG3D model [2] with a batch size of 20 making steps on 50k∼100k training images.

Table S1. List of text prompts used to collect target images for training our pose-preserved diffusion model.

Strategies	Full text prompts
Identity mixing	*"a FHD photo of a human face"*
Pose-guranteed prompts	*"a 3D render of a face in Pixar style"* *"a 3D render of a face of Lizardman monster in fantasy movie"* *"a 3D render of a face of Minotaurus in fantasy movie"* *"a 3D render of a head of a Lego man 3D model"* *"a 3D render of a Stone Golem head in fantasy movie"* *"a FHD photo of a face of a Skeleton in fantasy movie"* *"a FHD photo of a face of Monkey"* *"a FHD photo of white Greek statue face"*
Different domain generator	*"a FHD photo of a Cat face"*
Different domain pose-guranteed prompts	*"a 3D render of a face of a Cat in Zootopia style"* *"a 3D render of a face of a Pig in Zootopia style"* *"a 3D render of a face of a Sheep in Zootopia style"* *"a FHD photo of a face of a Wol"*

Table S2. List of complete text prompts that corresponds to each concise prompt.

Text types	Concise prompts	Full text prompts
Characters	SpongeBob	*"a FHD photo of a face of SpongeBob"*
	Sesame Street	*"a FHD photo of a fluffy character from 'Sesame Street' "*
	Rango	*"a 3D render of a face of Rango from 'Rango'"*
	Jack Skellington	*"a FHD photo of a face of Jack Skellington from 'The Nightmare Before Christmas'"*
	Minions	*"a 3D render of Doraemon"*
	Homer Simpson	*"a 3D render of Homer simpson"*
	Doraemon	*"a 3D render of yellow Minion with two eyes from the 'Despicable Me' franchise"*
	Sonic	*"a FHD photo of a face of Sonic the Hedgehog from 'Sonic the Hedgehog'"*
	Shaun the Sheep	*"a FHD photo of Shaun the Sheep"*
	Teddy bear	*"a FHD photo of a Teddy bear"*
	Barbie doll	*"a FHD photo of a barbie doll"*
	Scooby-Doo	*"a 3D render of Scooby-Doo"*
Animals	Elephant	*"a FHD photo of a face of a Elephant"*
	Turtle	*"a FHD photo of a face of a Turtle"*
	Horse	*"a FHD photo of a face of a Horse"*
	Iguana	*"a FHD photo of a face of a Iguana"*
	Cameleon	*"a FHD photo of a face of a Cameleon"*
	Bear	*"a FHD photo of a face of a Bear"*
	Dog	*"a FHD photo of a face of a Dog"*
	Goat	*"a FHD photo of a face of a Goat"*
	Cow	*"a FHD photo of a face of a Cow"*

D.3. User study

To evaluate the quality of the generated samples and 3D shapes from the shifted generators from five different methods of text-guided domain adaptation for 3D generative models including our PODIA-3D, we conducted a user study using a survey platform. The study involved 60 participants, and a total of 10,500 votes were collected. We applied each of the 5 methods of text-guided domain adaptation for 3D generative models, including our own method, to adapt the EG3D [2] generator to 7 text prompts, each converting a human face to a style with a large domain gap from the FFHQ [14] or AFHQ-cat [13, 3] domains, namely Horse', Cow', Elephant', Iguana', Turtle', Sesame street', 'SpongeBob'. For each text prompt, we sampled 30 images and a 3D shape from each generator, and placed the results of each method side-by-side. To assist the participants in the user study, we included screenshots of Google search pages corresponding to each text prompt. A total of 60 people completed the survey, providing 10,500 votes. We asked the participants in the user study to rate the quality of the rendered 2D images on a scale of 1 to 5. They were presented with the following questions: (1) Do the rendered 2D images

Table S3. List of complete text prompts that correspond to each subclass prompt used for text-guided debiasing.

Superclasses	Subclasses	Subclasses prompts
Bear	Black Bear	*"a FHD photo of a face of a Black Bear"*
	Brown Bear	*"a FHD photo of a face of a Brown Bear"*
	Giant Panda	*"a FHD photo of a face of a Giant Panda"*
	Polar Bear	*"a FHD photo of a face of a Polar Bear"*
Dog	Vizsla	*"a FHD photo of a face of a Vizsla"*
	Pug dog	*"a FHD photo of a face of a Pug dog"*
	German Shepherd	*"a FHD photo of a face of a German Shepherd"*
	Golden Retriever	*"a FHD photo of a face of a Golden Retriever"*
	Beagle	*"a FHD photo of a face of a Beagle"*
	Rottweiler	*"a FHD photo of a face of a Rottweiler"*
	Dachshund	*"a FHD photo of a face of a Dachshund"*
	Siberian Husky	*"a FHD photo of a face of a Siberian Husky"*
	Schnauzer	*"a FHD photo of a face of a Schnauzer"*
	Cocker Spaniel	*"a FHD photo of a face of a Cocker Spaniel"*
	Welsh Corgi	*"a FHD photo of a face of a Welsh Corgi"*
Zootopia	Rabbit in Zootopia Style	*"a FHD photo of a Rabbit in Zootopia Style with big eyes"*
	Fox in Zootopia Style	*"a FHD photo of a Fox in Zootopia Style with big eyes"*
	Buffalo in Zootopia Style	*"a FHD photo of a Buffalo in Zootopia Style with big eyes"*
	Cheetah in Zootopia Style	*"a FHD photo of a Cheetah in Zootopia Style with big eyes"*
	Sheep in Zootopia Style	*"a FHD photo of a Sheep in Zootopia Style with big eyes"*
	Gazelle in Zootopia Style	*"a FHD photo of a Gazelle in Zootopia Style with big eyes"*
	Tiger in Zootopia Style	*"a FHD photo of a Tiger in Zootopia Style with big eyes"*
	Bear in Zootopia Style	*"a FHD photo of a Bear in Zootopia Style with big eyes"*
	Koala in Zootopia Style	*"a FHD photo of a Koala in Zootopia Style with big eyes"*
3D animation characters	Toy Story	*"a FHD photo of a character in film 'Toy Story' style"*
	Moana	*"a FHD photo of a character in film 'Moana' style"*
	How to Train Your Dragon	*"a FHD photo of a character in film 'How to Train Your Dragon' style"*
	Brave	*"a FHD photo of a character in film 'Brave' style"*
	Coco	*"a FHD photo of a character in film 'Coco' style"*
	Ratatouille	*"a FHD photo of a character in film 'Ratatouille' style"*
	Rise of the Guardians	*"a FHD photo of a character in film 'Rise of the Guardians' style"*
	Tangled	*"a FHD photo of a character in film 'Tangled' style"*
	UP	*"a FHD photo of a character in film 'UP' style"*
	Moana	*"a FHD photo of a character in film 'Moana' style"*
Human face	Kids	*"a FHD photo of a kids"*
	Teenagers	*"a FHD photo of a teenagers"*
	20s	*"a FHD photo of a person in their 20s"*
	30s	*"a FHD photo of a person in their 30s"*
	40s	*"a FHD photo of a person in their 40s"*
	50s	*"a FHD photo of a person in their 50s"*
	60s	*"a FHD photo of a person in their 60s"*
	70s	*"a FHD photo of a person in their 70s"*
	80s	*"a FHD photo of a person in their 80s"*

accurately reflect the semantics of the target text? (text-2D image correspondence), (2) Are the rendered 2D images realistic? (photorealism), and (3) Are the rendered 2D images diverse within the image group? (diversity). These questions are similar to those used in [16]. Additionally, participants were asked to rate the accuracy of text-correspondence and the sense of depth and details of the 3D shapes on a scale of 1 to 5, based on the following questions: (4) Does the extracted 3D shape accurately reflect the semantics of the target text? (text-3D shape correspondence), and (5) Do the 3D shapes have a great sense of depth and detail? (sense of depth and details of 3D shapes). Finally, the mean score for each method was then calculated.

E. Additional Results

E.1. Results of text-driven 3D domain adaptation

We present additional results of qualitative comparison between our text-guided domain adaptation outcomes and other baselines in Fig S2 and Fig S3. Our method empow-

ers the EG3D [2] model to generate multi-view consistent images in a broad range of text-guided domains (animals, characters) with high text-image correspondence, diversity and excellent quality of 3D shape, while other baselines don't.

E.2. SD vs DGD vs PPD

Fig. S4 presents additional comparison results of text-guided image-to-image translation and text-guided domain adaptation, evaluated using Stable diffusion (SD)[28], the depth-guided diffusion (DGD)[28], our pose-preserved diffusion (PPD) models, and specialized-to-general (S-to-G).

PPD enables high pose-consistency and text-image correspondence in image translation, which leads to successful domain adaptation of the EG3D [2] 3D generator while preserving 3D shapes. Furthermore, S-to-G sampling helps to address the issue of detail bias.

E.3. Text-guided debiasing for human face images

Fig. S5 demonstrates that our debiasing method enables the generation of a more agebalanced representation for FFHQ images that bias towards individuals in their 20s and 30s without requirements of an- notated datasets or pre-trained attribute classifiers.

F. Discussion

Limitation. We investigate that domain adaptation to non-living objects, such as chairs and hamburgers, which lack directional information, leads to a low level of text-image correspondence as represented in Fig. S6. Our text-guided domain adaptation process relies on the performance of text-to-image diffusion models, which means that any limitations of the chosen diffusion models are also present in our pipeline. For this work, we have utilized both the Stable Diffusion [28] and Depth-Guided Diffusion [28] models. As stated in the Stable diffusion model card, the model has certain limitations that include the inability to achieve complete photorealism, compositionality, proper face generation, and the generation of images with languages other than English. These limitations may have an impact on the performance of our method.

Social impacts. Our novel PODIA-3D technique enables the creation of high-quality 3D samples in text-guided domains, even in cases where there are substantial gaps from the source domain. Importantly, this can be achieved without requiring any artistic skills. However, it is crucial to acknowledge that this technology has the potential to be misused for creating visuals that are upsetting or insulting. As per the Stable diffusion model card [28], the model can be misused in various ways such as creating inaccurate, hurtful, or offensive depictions of individuals, and cultures. Therefore, we strongly advise individuals to use our approach judiciously and solely for its intended purposes.

G. List of Abbreviations

Table S4 describes the significance of various abbreviations, anacronyms and notations used throughout the papers.

Table S4. List of Abbreviations.

Abbreviation	Meaning
SD, $\epsilon_\phi^{\text{SD}}$	stable diffusion
DGD, $\epsilon_\phi^{\text{DGD}}$	depth-guided diffusion
PPD, $\epsilon_\phi^{\text{PPD}}$	pose-preserved diffusion
S-to-G	specialized-to-general
T-I2I	text-guided image-to-image translation
t	diffusion timestep
t_r	return diffusion timestep
T	total diffusion timesteps
$\boldsymbol{x}^{\text{src}}$, $\boldsymbol{x}_0^{\text{src}}$	source image
$\boldsymbol{x}_t^{\text{src}}$	noised source image at diffusion timestep t
$\boldsymbol{x}^{\text{trg}}$, $\boldsymbol{x}_0^{\text{trg}}$	target image for training PPD
$\boldsymbol{x}^{\text{tda}}$	target image for domain adaptation
$y^{\text{p-guar}}$	pose-guaranteed prompts
y^{trg}	target text prompt for training PPD
y^{tda}	target text prompt for domain adaptation
$y_i^{\text{tda},\mathcal{A}}$	ith subclass texts for domain adaptation
$\mathcal{D}^{\text{tda}}$	pose-aware target dataset
$\mathcal{D}^{\text{PPD}}$	PPD training data
$\mathcal{D}^{\text{deb}}$	debiased target dataset
$\boldsymbol{z}$	random latent vectors for 3D generator
$\boldsymbol{c}$	random camera parameters for 3D generator
$\boldsymbol{d}_i^{\text{src}}$	ith source depth map
$\boldsymbol{q}_i^{\text{trg}}$	ith target diffusion latent
G	3D generator
E^V	VQGAN encoder
D^V	VQGAN Decoder
N^{src}	the number of source images
N^{sub}	the number of subclasses
η	PPD ratio
$\mathcal{L}_{\text{ADA}}$	ADA loss
$\mathcal{L}_{\text{den}}$	density regularization loss
$\mathcal{X}^{\text{tda}}$	target domain
$\mathcal{A}$	attribute

References

[1] Aibek Alanov, Vadim Titov, and Dmitry Vetrov. Hyperdomainnet: Universal domain adaptation for generative adversarial networks. *arXiv preprint arXiv:2210.08884*, 2022.

[2] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022.

[3] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

[4] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceed-*

ings of the IEEE/CVF conference on computer vision and pattern recognition, pages 12873–12883, 2021.

[5] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *arXiv preprint arXiv:2108.00946*, 2021.

[6] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[7] he Resource for Biocomputing Visualization and Informatics (RBVI). Ucsf chimerax. In *https://www.cgl.ucsf.edu/chimerax/*.

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020.

[9] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[10] Xinyu Huang, Peng Wang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its application. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2702–2719, 2019.

[11] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. If you use this software, please cite it as below.

[12] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *arXiv preprint arXiv:2006.06676*, 2020.

[13] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021.

[14] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[15] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[16] Gwanghyun Kim and Se Young Chun. Datid-3d: Diversity-preserved domain adaptation using text-to-image diffusion for 3d generative model. *arXiv preprint arXiv:2211.16374*, 2022.

[17] Youngjung Kim, Hyungjoo Jung, Dongbo Min, and Kwanghoon Sohn. Deep monocular depth estimation via integration of global and local predictions. *IEEE transactions on Image Processing*, 27(8):4131–4144, 2018.

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[20] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022.

[21] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

[22] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[23] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[24] 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. *arXiv preprint arXiv:2103.00020*, 2021.

[25] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[26] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *ICCV*, 2021.

[27] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3), 2022.

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[30] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

[31] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[32] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[33] Kunpeng Song, Ligong Han, Bingchen Liu, Dimitris Metaxas, and Ahmed Elgammal. Diffusion guided domain adaptation of image generators. *arXiv preprint arXiv:2212.04473*, 2022.

[34] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*, 2020.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[36] Chaoyang Wang, Simon Lucey, Federico Perazzi, and Oliver Wang. Web stereo video supervision for depth prediction from dynamic scenes, 2019.

[37] Qiang Wang, Shizhen Zheng, Qingsong Yan, Fei Deng, Kaiyong Zhao, and Xiaowen Chu. Irs: A large naturalistic indoor robotics stereo dataset to train deep models for disparity and surface normal estimation. *arXiv preprint arXiv:1912.09678*, 2019.

[38] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. 2020.

[39] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. Monocular relative depth perception with web stereo data supervision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[40] Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. Structure-guided ranking loss for single image depth prediction. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[41] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1790–1799, 2020.

[42] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902*, 2022.

[43] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.