DATID-3D: Diversity-Preserved Domain Adaptation Using Text-to-Image Diffusion for 3D Generative Model (Supplementary Material)

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

A. Additional Results

A.1. Videos

We provide an accompanying supplementary video that better visualizes and demonstrates that our methods, DATID-3D, enables the shifted generator to synthesize multi-view consistent images with high fidelity and diversity in a wide range of text-guided targeted domains at gwang-kim.github.io/datid_3d.

A.2. Results of text-driven 3D domain adaptation

More results for text-driven 3D domain adaptation using the EG3D [4] generators pre-trained on FFHQ [14] or AFHQ Cats [5, 13] are illustrated in Figures S1 and S2, respectively. Without additional images and knowledge of camera distribution, our framework allows the synthesis of diverse, high-fidelity multi-view consistent images in a wide range of text-guided domains beyond the training domains.

A.3. Results of pose-controlled synthesis

The results of our pose-controlled image and 3D shape synthesis in the text-guided domain are shown in Figure S3. For more results, see the provided supplementary video.

A.4. Additional qualitative comparison results.

In Figure S4, we provide the more qualitative comparison of our method with two baselines, StyleGAN-NADA* [20] and HyperDomainNet* [1]. By exploiting text-to-image diffusion models and adversarial training, our framework helps the shifted generator to synthesize more photorealistic and varied images.

A.5. Additional quantitative comparison results.

We additionally evaluate Kernel Inception Distance (KID) [2] to calculate the distance between the distributions of generated samples and test images in the target domain because when the dataset is small, Frechet inception distance (FID) [10] can be easily biased while KID adopts unbiased design. As used in the user study, EG3D pre-trained on 512^2 images in FFHQ [14] and four text prompts converting a human face to 'Pixar', 'Neanderthal', 'Elf' and 'Zombie' styles, respectively, are employed for evaluation. We generate 3,000 images generated through text-to-image diffusion models with a different random seed per text prompt. As presented in Table S1, our results demonstrate the superior KID as compared to the baselines.

Table S1. Quantitative comparisons with the baselines in diversity, text-image correspondence and photo realism.

	KID↓
StyleGAN-NADA*	0.156
HyperDomainNet*	0.133
Ours	**0.012**

B. Details on Methods

B.1. Algorithms

Text-guided target dataset generation. The algorithm for text-guided target dataset generation is described in Algorithm 1. With each random latent vector $\boldsymbol{z}_i \in \mathcal{Z}$ and camera parameter $\boldsymbol{c}_i \in \mathcal{C}$, we synthesize a source image $\boldsymbol{x}_i^{\text{src}}$ using pre-trained 3D generator G_θ. Then, guided by a text prompt $\boldsymbol{y}$, we perform text-guided image-to-image manipulation (`T_I2I`) to generate $\boldsymbol{x}_i^{\text{trg}}$ from $\boldsymbol{x}_i^{\text{src}}$ using the text-to-image diffusion model $\boldsymbol{\epsilon}_\phi$. In `T_I2I`, we first embed $\boldsymbol{x}^{\text{src}}$ into $\boldsymbol{q}_0$ through E^V and perturb it to generate $\boldsymbol{q}_{t_r}^{\text{trg}}$ through the stochastic forward DDPM (Denoising Diffusion Probabilistic Models) process [11] while the return step $t_r < T_p$, where T_p is the pose-consistency step. Then, we execute the sampling process to obtain $\boldsymbol{q}_0^{trg}$ from the noisy latent $\boldsymbol{q}_{t_r}^{\text{trg}}$ using $\boldsymbol{\epsilon}_\phi$. s controls the scale of gradients from

[†]Corresponding author.

Figure S1. Variety of text-guided adaption results. We fine-tuned EG3D [4], pre-trained on 512^2 images in FFHQ [14], to generate diverse samples for a variety of concepts.

a target prompt y and a negative prompt y_{neg}. Finally, the target image $\boldsymbol{x}^{\text{trg}}$ is obtained using the VQGAN decoder D^V. By repeating the above process N times, we can construct a target dataset $\mathcal{D}$.

CLIP and pose reconstruction-based filtering. The algorithm for CLIP and pose reconstruction-based filtering process is presented in Algorithm 2. For all $(\boldsymbol{x}_i^{\text{src}}, \boldsymbol{x}_i^{\text{trg}})$ in the raw target dataset $\mathcal{D}$, we first compute the CLIP distance score d_{CLIP} between the target image $\boldsymbol{x}_i^{\text{trg}}$ and the target prompt y. If $d_{\text{CLIP}} > \alpha$, then replace $\boldsymbol{x}_i^{\text{trg}}$ with a new one through `T_I2I` and repeat the CLIP-based filtering again. Otherwise, we convert $\boldsymbol{x}_i^{\text{trg}}$ to a reconstructed image $\boldsymbol{x}_i^{\text{rec}}$ using the Reconstructor latent diffusion $\boldsymbol{\epsilon}_{\phi^{\text{rec}}}$. Then, we calculate the pose difference score d_{pose} between the reconstructed image $\boldsymbol{x}_i^{\text{rec}}$ and the target image $\boldsymbol{x}_i^{\text{trg}}$. If $d_{\text{pose}} > \beta$, then replace $\boldsymbol{x}_i^{\text{trg}}$ with a new one through `T_I2I` and repeat the CLIP-based filtering again. Otherwise, we can finish the filtering for $\boldsymbol{x}_i^{\text{trg}}$ and save a set of $(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{src}}, \boldsymbol{x}_i^{\text{trg}})$ to $\mathcal{D}_f$. In practice, it sometimes takes a too long time to repeat the process until $\boldsymbol{x}_i^{\text{trg}}$ passes, we only repeat it by K_{f} times, which was set to 5 for our experiments.

Figure S2. Variety of text-guided adaption results. We fine-tuned EG3D [4], pre-trained on 512^2 images in AFHQ Cats [5, 13] to generate diverse samples for a variety of concepts.

Diversity-preserved domain adaptation. The algorithm for diversity-preserved domain adaptation is provided in Algorithm 3. We first clone the pre-trained 3D generator G_θ to $G_{\theta'}$ and initialize pose-conditioned discriminator D_ψ. For $i = 1, 2, ..., N$, we first sample a random latent vector and camera parameter. Then, we compute ADA loss for the generator $\mathcal{L}_{\text{ADA}}^{\theta'}$ with generated images $G_{\theta'}(\boldsymbol{z}_i, \boldsymbol{c}_i)$ using D_ψ and the stochastic non-leaking augmentation A. Also, we calculate the density regularization loss $\mathcal{L}_{\text{den}}$ with randomly chosen points v from the volume $\mathcal{V}$ for each rendered scene. With these two losses, the generator is updated. Next, we compute ADA losses for the discriminator, $\mathcal{L}_{\text{ADA}}^{\psi,\text{fake}}$ and $\mathcal{L}_{\text{ADA}}^{\psi,\text{real}}$, with generated images $G_{\theta'}(\boldsymbol{z}_i, \boldsymbol{c}_i)$ and real targets $\boldsymbol{x}_i^{\text{trg}}$, respectively. Combining these two losses, the discriminator is updated. We repeat this process for K epochs.

C. Implementation Details

C.1. 3D generative model

We adopt EG3D [4], the state-of-the-art 3D generative model pre-trained on 512^2 images in FFHQ [14] and AFHQ Cats [5, 13] as our source generator. Its generator is composed of a backbone, decoder, volume rendering, and super-resolution parts. The backbone consists of the StyleGAN2

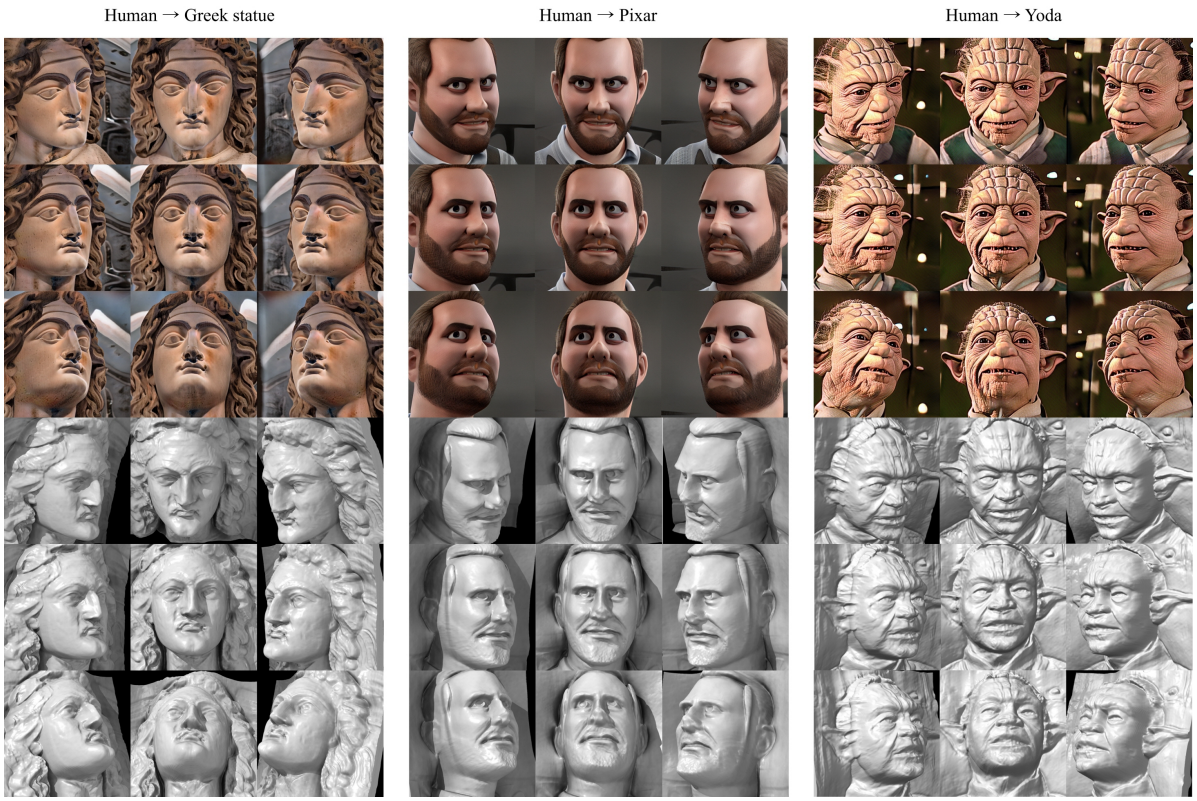


Figure S3. Pose-controlled images and 3D shapes in text-guided domain through our method. See the supplementary videos at gwang-kim.github.io/datid_3d

Figure S4. Qualitative comparison with the 3D extension of existing 2D text-guided domain adaptation methods (the star mark (*)). Our DATID-3D yielded diverse samples while other baselines did not.

Algorithm 1: Text-guided target dataset generation

Input: G_θ, $\boldsymbol{\epsilon}_\phi$, E^V, D^V, y, y^{neg}, t_r, s, N, $*$
Output: $\mathcal{D} = \{(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{src}}, \boldsymbol{x}_i^{\text{trg}})\}_{i=1}^N$

Function T_I2I ($\boldsymbol{x}^{\text{src}}$, y, y^{neg}, $\boldsymbol{\epsilon}_\phi$, $*$) **:**
  $\boldsymbol{q}_0 = E^V(\boldsymbol{x}^{\text{src}})$, $\boldsymbol{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $\boldsymbol{q}_{t_r}^{\text{trg}} = \sqrt{\bar{\alpha}_{t_r}}\boldsymbol{q}_0 + \sqrt{1-\bar{\alpha}_{t_r}}\boldsymbol{n}$
  for $t = t_r, t_r - 1, \ldots, 1$ **do**
    $\boldsymbol{\epsilon}_\phi^{\text{comb}} = s\boldsymbol{\epsilon}_\phi(\boldsymbol{q}_t^{\text{trg}}, t, y) + (1-s)\boldsymbol{\epsilon}_\phi(\boldsymbol{q}_t^{\text{trg}}, t, y^{\text{neg}})$
    $\boldsymbol{q}_{t-1}^{\text{trg}} = \text{Sampling}(\boldsymbol{q}_t^{\text{trg}}, \boldsymbol{\epsilon}_\phi^{\text{comb}}, t)$
  $\boldsymbol{x}^{\text{trg}} = D^V(\boldsymbol{q}_0^{\text{trg}})$
  return $\boldsymbol{x}^{\text{trg}}$
$\mathcal{D} = \{\}$
for $i = 1, 2, \ldots, N$ **do**
  $\boldsymbol{z}_i \in \mathcal{Z}$, $\boldsymbol{c}_i \in \mathcal{C}$
  $\boldsymbol{x}_i^{\text{src}} = G_\theta(\boldsymbol{z}_i, \boldsymbol{c}_i)$
  $\boldsymbol{x}_i^{\text{trg}}$=T_I2I ($\boldsymbol{x}_i^{src}$, y, y^{neg}, $\boldsymbol{\epsilon}_\phi$, $*$)
  Append $(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{src}}, \boldsymbol{x}_i^{\text{trg}})$ to $\mathcal{D}$.

Algorithm 2: CLIP and pose reconstruction-based filtering

Input: $\mathcal{D}$, $\boldsymbol{\epsilon}_{\phi^{\text{rec}}}$, $\boldsymbol{\epsilon}_\phi$, y^{src} , y y^{neg}, N, $*$
Output: $\mathcal{D}_f = \{(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{src}}, \boldsymbol{x}_i^{\text{trg}})\}_{i=1}^N$

$\mathcal{D}_f = \{\}$
for $i = 1, 2, \ldots, N$ **do**
  $(\boldsymbol{x}_i^{\text{src}}, \boldsymbol{x}_i^{\text{trg}}) \in \mathcal{D}$
  while *True* **do**
    if $d_{\text{CLIP}}(\boldsymbol{x}_i^{\text{trg}}, y) > \alpha$ **then**
      $\boldsymbol{x}_i^{\text{trg}}$=T_I2I ($\boldsymbol{x}_i^{\text{src}}$, y, y^{neg}, $\boldsymbol{\epsilon}_\phi$, $*$)
      continue
    else
      $\boldsymbol{x}_i^{\text{rec}}$ =T_I2I ($\boldsymbol{x}_i^{\text{src}}$, y^{src}, None, $\boldsymbol{\epsilon}_{\phi^{\text{rec}}}$, $*$)
      if $d_{\text{pose}}(\boldsymbol{x}_i^{\text{rec}}, \boldsymbol{x}_i^{\text{src}}) > \beta$ **then**
        $\boldsymbol{x}_i^{\text{trg}}$=T_I2I ($\boldsymbol{x}_i^{\text{src}}$, y, y^{neg}, $\boldsymbol{\epsilon}_\phi$, $*$)
        continue
      else
        break
  Append $(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{src}}, \boldsymbol{x}_i^{\text{trg}})$ to $\mathcal{D}_f$.

generator [15] and a mapping network with 8 hidden layers. The decoder is constructed as an MLP with a single hidden layer with soft plus activation and neural rendering [18] of features [19] using two-pass importance is utilized. The super-resolution module is implemented with two StyleGAN2 blocks with modulated convolutions. EG3D's discriminator is based on a StyleGAN2 discriminator with two changes, dual discrimination, and camera pose-conditioning.

Algorithm 3: Diversity-preserved domain adaptation

Input: G_θ (pre-trained 3D generator), $\mathcal{D}_f$ (filtered dataset), N (Number of data), K (total number of epochs), A (stochastic non-leaking augmentation), f, $*$
Output: $G_{\theta'}$

$G_{\theta'} \leftarrow \text{clone}(G_\theta)$, $D_\psi \leftarrow$ Initialize_D
for $k = 1, 2, \ldots, K$ **do**
  for $i = 1, 2, \ldots, N$ **do**
    $\boldsymbol{z}_i \in \mathcal{Z}$, $\boldsymbol{c}_i \in \mathcal{C}$, $v_i \in \mathcal{V}$
    // Step 1: Update $G_{\theta'}$
    $\mathcal{L}_{\text{ADA}}^{\theta'} = -f(D_\psi(A(G_{\theta'}(\boldsymbol{z}_i, \boldsymbol{c}_i)), \boldsymbol{c}_i)$
    $\mathcal{L}_{\text{den}}^{\theta'} = \|\sigma_{\theta'}(v_i) - \sigma_{\theta'}(v_i + \delta v_i)\|$
    $\theta' \leftarrow$ Update_$G(\theta', \mathcal{L}_{\text{ADA}}^{\theta'} + \lambda_{\text{den}}\mathcal{L}_{\text{den}})$
    // Step 1: Update D_ψ
    $\mathcal{L}_{\text{ADA}}^{\psi,\text{fake}} = f(D_\psi(A(G_{\theta'}(\boldsymbol{z}_i, \boldsymbol{c}_i)), \boldsymbol{c}_i)$
    $(\boldsymbol{c}_i, \boldsymbol{x}_i^{\text{trg}}) \in \mathcal{D}$
    $\mathcal{L}_{\text{ADA}}^{\psi,\text{real}} = f(-D_\psi(A(\boldsymbol{x}^{\text{trg}}), \boldsymbol{c}_i)$
      $+\lambda\|\nabla D_\psi(A(\boldsymbol{x}^{\text{trg}}), \boldsymbol{c}_i)\|^2)$
    $\psi \leftarrow$ Update_$D(\psi, \mathcal{L}_{\text{ADA}}^{\psi,\text{fake}} + \mathcal{L}_{\text{ADA}}^{\psi,\text{real}})$

C.2. Text-to-image diffusion model

We employ Stable diffusion [21] as our text-to-image diffusion model. It is a latent-based diffusion model and leverages a pre-trained 123M CLIP ViT-L/14 [20] text encoder to provide the model with the condition of text prompts. The diffusion model where 860M UNet [22] with the text encoder are combined is lightweight and enables text-to-image synthesis on GPU at 10GB VRAM. We use Stable diffusion v1.4, where 977k steps were taken at 512×512 images paired with text captions from a subset of the LAION-5B [23] database.

For the diffusion sampling method, we choose PLMS [17], one of the state-of-the-art sampling methods, accelerating the diffusion process with high quality. We set the number of inference steps to 50, which enables us to generate a high-quality image in 1∼2 seconds. We generally set y^{neg} to None. Also, we generally set the return step t_r and the guidance scale s to 700 and 10, respectively.

C.3. Pose-extractor

As a pose-extractor, we use 6DRepNet [9] that demonstrates the state-of-the-art performance on BIWI [3] head pose estimation benchmark. This model predicts a pose vector on images that includes yaw, pitch, and roll vectors. We found that this model works well on both FFHQ [14] and AFHQ Cats [5, 13] images, thus we use the model for both types of images.

C.4. Fine-tuning details

We fine-tune the 3D generative models with a batch size of 20 until the models see 50,000∼200,000 images. We use a learning rate of 0.002 for both the generator and discriminator. For the discriminator's input, we blur images, progressively diminishing the blur degree following [4, 13] and don't use style mixing during training. We use ADA loss combined with R1 regularization with $\lambda = 5$. We set the strength of density regularization λ_{den} to 0.25.

C.5. 3D shape visualization

To visualize 3D shapes, we first extract iso-surfaces from the density field using marching cubes following [4]. Then, we view the 3D surfaces using UCSF Chimerax [8].

C.6. Text prompts

In the main paper and supplementary, we use a concise text prompt to refer to each text prompt. Full-text prompts corresponding to each concise text prompt are summarized in Table S2.

D. Experimental Details

D.1. Evaluation details

Baselines. In StyleGAN-NADA* that is a 3D extended version of StyleGAN-NADA [20], we fine-tune the 3D generator G^θ with the directional CLIP loss as follows:

$$\mathcal{L}^\theta_{\text{direction}} = 1 - \frac{\langle \Delta I, \Delta T \rangle}{\|\Delta I\| \|\Delta T\|}, \tag{S1}$$

where $\Delta I = E^C_I(\boldsymbol{x}^{\text{gen}}) - E^C_I(\boldsymbol{x}^{\text{src}}), \Delta T = E^C_T(y^{\text{tar}}) - E^C_T(y^{\text{src}})$. We implement the loss and optimization part based on the official StyleGAN-NADA codebase [20].

In HyperDomainNet* that is a 3D extended version of HyperDomainNet [1], in-domain angle consistency loss $\mathcal{L}_{\text{indomain}}$ is added to the directional CLIP loss for preserving the CLIP similarities among images before and after domain adaptation.

$$\mathcal{L}^\theta_{\text{indomain}} = \sum_{i,j}^{n} (\langle E^C_I(\boldsymbol{x}^{\text{gen}}_i), E^C_I(\boldsymbol{x}^{\text{gen}}_j) \rangle - \langle E^C_I(\boldsymbol{x}^{\text{src}}_i), E^C_I(\boldsymbol{x}^{\text{src}}_j) \rangle)^2, \tag{S2}$$

We implement the loss part based on the official HyperDomainNet [1].

KID. Based on the StyleGAN3 [13] codebase implementation, we calculate Kernel Inception Distance (KID) between 50,000 produced images and 3,000 training images.

User study. For the user study, we collect 9,000 votes from 75 people using a survey platform. We adapt the generator using each method for four text prompts converting a human face to 'Pixar', 'Neanderthal', 'Elf' and 'Zombie' styles as these prompts are used in the previous work, StyleGAN-NADA [20]. Then, for each text prompt, we sample 30 images from each generator and put the results of each method side-by-side. To quantify opinions, we requested users to rate the perceptual quality on a scale of 1 to 5 for 3 questions as we introduced in the main text. Finally, we report the mean of each score for each method, respectively.

Table S3. High diversity is ensured by sampling more target images (large n) with our CLIP and pose reconstruction-based filtering.

	KID ↓
$n = 100$	0.024
$n = 500$	0.015
$n = 1000$	0.013
$n = 3000$	0.012

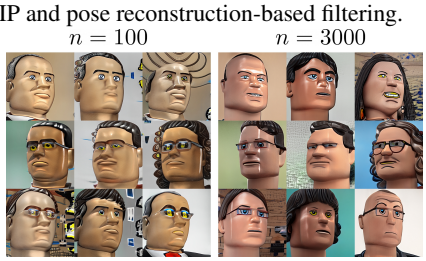
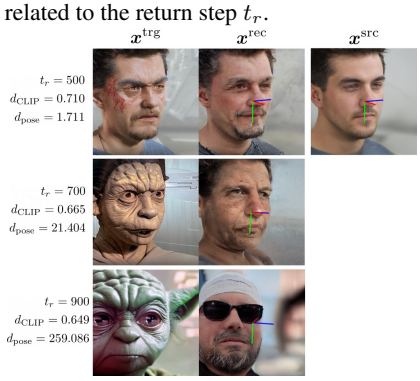


Table S4. Trade-off between image-text correspondence d_{CLIP} and pose-consistency d_{pose} related to the return step t_r.

t_r	d_{pose}↓	d_{CLIP}↓
500	**8.133**	0.689
600	23.381	0.672
700	86.516	0.657
800	263.081	0.654
900	327.478	**0.652**



Non-adversarial fine-tuning. One generator per instance is optimized like StyleGAN-NADA* [20], but the difference is that the guidance is from CLIP image encoding of the target images that were generated from text-to-image diffusion models, not CLIP text encoding.

D.2. 3D GAN inversion

For single-view manipulated 3D reconstruction, we invert a real image into the latent vector w in $\mathcal{W}^+$ space. To achieve this, we obtain the camera parameter c with pre-trained pose extractor [7, 9] and we initialize w as a mean of 10,000 ws that are mapped from zs which are randomly sampled from Normal distribution. Then, we generate a images with 3D generator and compute a feature distance between the generated image and the real image using VGG-19 network. Then, using Adam optimizer [16], we update the w minimizing the feature distance for 1,000 steps.

Table S2. List of full text prompts corresponding to each text prompt.

Source data type	Concise prompt	Full text prompt
FFHQ	Lego	*"a 3D render of a head of a lego man 3D model"*
	Greek Statue	*"a FHD photo of a white Greek statue"*
	Pixar	*"a 3D render of a face in Pixar style"*
	Orc	*"a FHD photo of a face of an orc in fantasy movie"*
	Elf	*"a FHD photo of a face of a beautiful elf with silver hair in live action movie"*
	Neanderthal	*"a FHD photo of a face of a neanderthal"*
	Skeleton	*"a FHD photo of a face of a skeleton in fantasy movie"*
	Zombie	*"a FHD photo of a face of a zombie"*
	Masquerade	*"a FHD photo of a face of a person in masquerade"*
	Peking opera	*"a FHD photo of face of character in Peking opera with heavy make-up"*
	Tekken	*"a 3D render of a Tekken game character"*
	Ston golem	*"a 3D render of a stone golem head in fantasy movie"*
	Devil	*"a FHD photo of a face of a devil in fantasy movie"*
	Baby	*"a FHD photo of a face of a cute baby"*
	Super Mario	*"a 3D render of a face of Super Mario"*
	Hobbit	*"a FHD photo of a face of Hobbit in Lord of the Rings "*
	Yoda	*"a FHD photo of a face of Yoda in Star Wars"*
AFHQ Cats	Golden statue	*"a photo of a face of an anmial golden statue"*
	Madagascar character	*"a 3D render of a face of a animal animation character in Madagascar style"*
	Eevee in Pokemon	*"a 3D render of a face of an eevee in Pokemon"*
	Lion in Zootopia style	*"a 3D render of a face of a lion in Zootopia style"*
	Cat in Zootopia style	*"a 3D render of a face of a cat in Zootopia style"*
	Wolf in Zootopia style	*"a 3D render of a face of a wolf in Zootopia style"*
	Fox in Zootopia style	*"a 3D render of a face of a fox in Zootopia style"*
	Sheep in Zootopia style	*"a 3D render of a face of a sheep in Zootopia style"*
	Pig in Zootopia style	*"a 3D render of a face of a pig in Zootopia style"*
	Hamster in Zootopia style	*"a 3D render of a face of a hamster in Zootopia style"*
	Racoon in Zootopia style	*"a 3D render of a face of a racoon in Zootopia style"*

Figure S5. Reconstructor successfully converted the target images into the images in the source domain (Human face) without unrealistic artifacts.

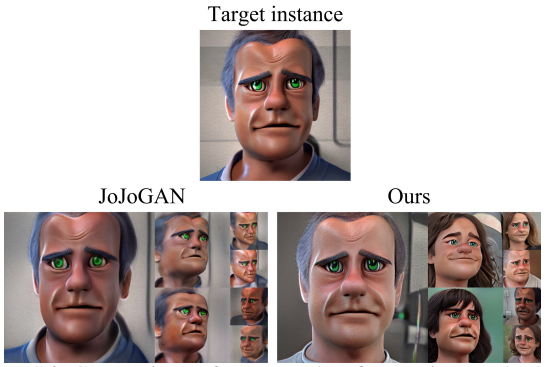


Figure S6. Comparison of our one-shot fine-tuning method with JoJoGAN [6], the state-of-the-art one-shot stylization method. Our method shows more diverse images with higher quality.

E. Additional Ablation Studies

Number of samples. We also analyzed the diversity, image quality, and training time depending on the number of samples. According to the quantitative (table) and qualitative (figures) results in Table S3, more sampled target images lead to improved image quality and diversity.

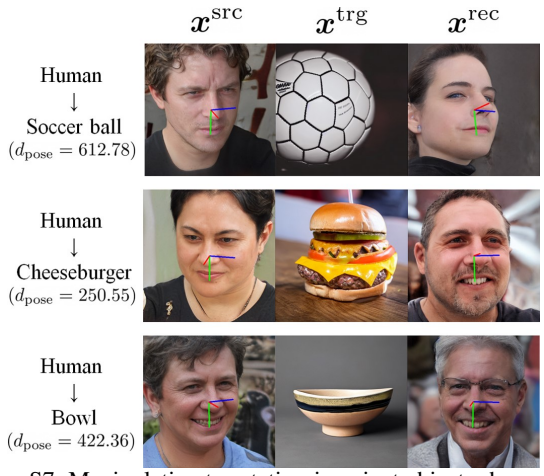


Figure S7. Manipulation to rotation-invariant objects shows high pose-difference scores.

Trade-off related to return step. The return step t_r is one of the important hyperparameters that determines the degree of text changes guided by image-to-image manipulation. We identified that there is a trade-off between image-text correspondence and pose consistency related to the return step. According to the quantitative (table) and qualitative (figures, 'Human face' $\rightarrow$ 'Yoda') results in Table S4, higher return step results in a lower CLIP distance score, but a higher pose difference score. Thus, we set t_r to 600$\sim$700 depending on the text prompts.

Effectiveness of the Reconstructor. Here, we compare the reconstruction performance between our proposed Reconstructor and original Stable diffusion. As a text prompt, the Stable diffusion uses "A photo of a human face" while the Reconstructor use "A photo of a <s> human face" that includes a specifier word. Our goal is to translate the manipulated target image back to the image in the source domain. As shown in Figure S5, the results from the stable diffusion reveal loss of pose information or artificial distortions because of its highly stochastic nature, whereas the Reconstructor successfully transforms the target images into the images in the source domain (Human face).

Effectiveness of one-shot fine-tuning using text-to-diffusion model. Here, we compare our one-shot fine-tuning method with the 3D extension of the state-of-the-art method of one-shot stylization for 2D generative models, JoJoGAN [6]. We add the camera sampling procedure to the domain adaptation pipeline in JoJoGAN. As presented in Figure S6, our one-shot fine-tuning method shows superior image quality and diversity for 3D generative models while the results from JoJoGAN severely overfit the target images.

F. Discussion

Limitation. We discovered that maintaining posture information in the target images created in Stage 1 is a crucial requirement for a successful text-driven 3D domain adaption. There are, however, certain inevitable circumstances that fit this requirement. The target object being rotation-invariant or in 2D space is one of the situations when pose information is lost. As shown in Figure S7, image manipulation of 'Human face' $\rightarrow$ 'Cheeseburger', 'Human face' $\rightarrow$ 'Soccer ball' and 'Human face' $\rightarrow$ 'Bowl' reports high pose-difference score, failing domain adaptation with flattened 3D shapes as described in Figure 11 in the main text.

Also, the supervision of our text-guided domain adaptation depends on the power of text-to-image diffusion models. So, the limitation of the chosen diffusion models is inherited in our pipeline. In this work, we adopt Stable diffusion [21]. According to the Stable diffusion model card, a limitation of the model includes falling short of achieving (1) complete photorealism, (2) compositionality, (3) proper face generation, (4) generating images with other languages except for English, and so on. These limitations can affect our performance of ours.

Diversity. The diversity of generated samples from the shifted generator depends on the diversity of the target dataset. For example, the target images from the text prompt 'Human face' $\rightarrow$ 'Super Mario' will be less diverse and more biased to the specific concept than the target images from 'Human face' $\rightarrow$ 'Pixar'. Thus, the domain adaptation results using the text prompt 'Human face' $\rightarrow$ 'Super Mario' are also less diverse than the results using 'Human face' $\rightarrow$ 'Pixar'. Also, as analyzed in [12], transfer learning of the generative models succeeds only when the target dataset has comparable or less diverse than the source dataset.

Social Impacts DATID-3D enables the generation of high-quality 3D samples in the text-guided domain as well as single-shot manipulated 3D reconstruction without artistic skills. Nevertheless, these can be applied maliciously to produce visuals that make people feel unpleasant or aggressive. This involves creating images that people are likely to find upsetting, frightening, or insulting, as well as information that reinforces stereotypes from the past or present. According to the Stable diffusion [21] model card, a misuse of the model includes (1) creating inaccurate, hurtful, or otherwise offensive depictions of individuals, their environment, cultures, and religions, (2) intentionally spreading stereotypical portrayals or discriminatory material, (3) impersonating individuals without their consent, (4) sexual content without viewer's permission, (5) depictions of horrifying violence and gore and so on. We thus strongly urge people to use our approach wisely and for the proper intended goals.

References

[1] Aibek Alanov, Vadim Titov, and Dmitry Vetrov. Hyperdomainnet: Universal domain adaptation for generative adversarial networks. *arXiv preprint arXiv:2210.08884*, 2022. 1, 6

[2] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 1

[3] Guido Borghi, Marco Venturelli, Roberto Vezzani, and Rita Cucchiara. Poseidon: Face-from-depth for driver pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4661–4670, 2017. 5

[4] 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. 1, 2, 3, 6

[5] 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. 1, 3, 5

[6] Min Jin Chong and David Forsyth. Jojogan: One shot face stylization. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVI*, pages 128–152. Springer, 2022. 7, 8

[7] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 6

[8] he Resource for Biocomputing Visualization and Informatics (RBVI). Ucsf chimerax. In *https://www.cgl.ucsf.edu/chimerax/*. 6

[9] Thorsten Hempel, Ahmed A Abdelrahman, and Ayoub Al-Hamadi. 6d rotation representation for unconstrained head pose estimation. *arXiv preprint arXiv:2202.12555*, 2022. 5, 6

[10] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 1

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020. 1

[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. 8

[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. 1, 3, 5, 6

[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. 1, 2, 3, 5

[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. 5

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[17] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 5

[18] B Mildenhall, PP Srinivasan, M Tancik, JT Barron, R Ramamoorthi, and R Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, 2020. 5

[19] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11453–11464, 2021. 5

[20] 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. 1, 5, 6

[21] 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. 5, 8

[22] 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. 5

[23] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022. 5