Abstract—There are many recent research efforts to fine-tune a pre-trained generator with a few target images to generate images of a novel domain. Unfortunately, these methods often suffer from overfitting or under-fitting when fine-tuned with a single target image. To address this, here we present a novel single-shot GAN adaptation method through unified CLIP space manipulations. Specifically, our model employs a two-step training strategy: reference image search in the source generator using a CLIP-guided latent optimization, followed by generator fine-tuning with a novel loss function that imposes CLIP space consistency between the source and adapted generators. To further improve the adapted model to produce spatially consistent samples with respect to the source generator, we also propose contrastive regularization for patchwise relationships in the CLIP space. Experimental results show that our model generates diverse outputs with the target patchwise relationships in the CLIP space. Experimental results show that our model successfully generates new images corresponding to the target domain while preserving the content attribute of the source domain. More generated samples can be found in the experiment section.

One-Shot Adaptation of GAN in Just One CLIP

Gihyun Kwon and Jong Chul Ye, Fellow, IEEE

Manuscript received 5 December 2022; revised 1 April 2023; accepted 1 June 2023. Date of publication 23 June 2023; date of current version 5 September 2023. This work was supported in part by the National Research Foundation of Korea under Grant NRF-2020R1A2B8003001980, in part by Korea Medical Device Development Fund grant funded, in part by the Korea government (the Ministry of Science and ICT, the Ministry of Trade, Industry and Energy, the Ministry of Health and Welfare, the Ministry of Food and Drug Safety) under Grants 1711137899 and KMDF_PR_20200901_0015, in part by the Field-oriented Technology Development Project for Customs Administration through National Research Foundation of Korea funded, in part by the Ministry of Science & ICT and Korea Customs Service under Grant NRF-2021M3I1A1097938, and in part by the KAIST Key Research Institute (Interdisciplinary Research Group) Project. Recommended for acceptance by T. Hassner. (Corresponding author: Jong Chul Ye.)

This work involved human subjects or animals in its research. The author(s) confirm(s) that all human/animal subject research procedures and protocols are exempt from review board approval.

Gihyun Kwon is with the Department of Bio and Brain Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, South Korea (e-mail: cyclomon@kaist.ac.kr).

Jong Chul Ye is with the Graduate School of AI, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, South Korea (e-mail: jong ye@kaist.ac.kr).

Our Github source is as follows: https://github.com/cyclomon/OneshotCLIP. Digital Object Identifier 10.1109/TPAMI.2023.3283551

I. INTRODUCTION

RECENTLY, several studies have tried to fine-tune a pre-trained generator model with limited number of target images so that the fine-tuned model can generate images of a novel domain. Early methods [1], [2], [3] showed the results of fine-tuned models using about 100 training images. Subsequent studies [4], [5] have shown that Generative Adversarial Networks(GAN) domain adaptation is possible even in extreme situations using fewer than 10 training images. A recent method [6] attempted to transform a pre-trained source generator to generate novel domain images by fine-tuning with only one target image. Although above methods show good performance in limited-shot situations, when there are smaller number of available training images, the generation performance drops significantly due to the severe overfitting. The problem is more pronounced when only one target image is available, which is the situation we are trying to solve. Although a recently proposed model [6] has mitigated the overfitting problem, the model often suffers from underfitting so that the model cannot fully reflect the domain of the target image.

To address this, here we present a novel single-shot fine-tuning approach of the pre-trained generator by using a unified Contrastive Language–Image Pre-training (CLIP) [7] model space manipulation, which shows better perceptual quality compared to other state-of-the-art few-shot adaptation models. For example, Fig. 1 shows that our model successfully generates new images with only a single target by fine-tuning the source model trained on large datasets (e.g. FFHQ, LSUN church, LSUN cars, and AFHQ dog). One of the most important contributions of this work is the discovery of the importance of the unified CLIP space manipulation.

Specifically, our method consists of two steps: finding the reference image and fine-tuning the model, all in the CLIP space. More specifically, inspired by the idea of the recent single-shot adaptation method called Mind the GAP [6], the first step is to find the image from the pre-trained source generator that most closely resembles a given target image $I_{trg}$ and uses it as a reference image $I_{ref}$. In particular, unlike existing GAN inversion methods and Mind the GAP [6] which uses pixel-wise similarity, we propose a new strategy that precisely aligns the semantic attributes between $I_{ref}$ and $I_{trg}$ in the CLIP space. In the second step, we fine-tune the model using the patch-wise and
global discriminators. In this stage, \( I_{ref} \) obtained in the previous step plays an important role as a reference point that guides the adapted generator to maintain the principal attribute information (e.g., content shape) of the source generator. To further match the source-target domain consistency, we propose a new consistency loss that maintains the CLIP space relation between the generated samples. Also, in order to maintain detailed spatial information between source and target models, we propose to maintain patch-wise semantic relations with contrastive learning in the CLIP space. To further verify the performance of our model, we show that our CLIP space manipulation strategy allows flexible and effective attribute editing.

Our contribution can be summarized as follows:

- We proposed to use CLIP in the latent search stage to find a reference that is more suitable for model adaptation, which have not been tried in the previous work.
- We proposed patch- and sample-wise consistency regularization in the CLIP embedding space, which is also the very first trial in GAN adaptation task.
- Our method outperforms other limited-shot GAN adaptation methods in both of qualitative and quantitative results.
- Our method has flexible framework so that it can be easily applied to other tasks, such as attribute editing, text-guided adaptation, etc.

II. RELATED WORK

A. Generative Adversarial Networks

Recent advances in generative adversarial networks (GAN) have shown impressive performance in generating realistic images. Starting from the seminal work of GAN [8], Progressive GAN [9] and BigGAN [10] models showed significantly improved generation performance for human face dataset and natural images. Furthermore, StyleGAN [11] showed generation quality that is almost indistinguishable from the real data.

One of the great advantages of StyleGAN is the model flexibility, which has inspired numerous subsequent models through modifications of StyleGAN, such as StyleGAN with a modified structure for light weight model [12], a disentangled model with additional latent space by modifying the architecture of StyleGAN [13], [14], a model with additional encoder network [15], [16], etc. Recently proposed StyleGAN2 [17] showed further improved generation performance through more efficient model structure and training strategies. StyleGAN-ADA [18], also another version of StyleGAN2, showed better generational performance through a novel training strategy using adaptive augmentation in discriminator training. Besides, various methods have been proposed to perform diverse tasks by leveraging the characteristics of StyleGAN, such as semantic segmentation [19], [20] for dataset generation, 3D image rendering [21], [22], inverse problems [23], [24], etc.

B. Image Manipulation

Manipulating specific attribute or domain of a given image is an important topic in computer vision. CycleGAN [25] successfully converted the image to other domains with novel cycle consistency, which was followed by several improved image translation methods [26], [27]. The recent contrastive unpaired translation model (CUT) [28] approach achieved state-of-the-art performance by preserving the spatial attributes between the input and the generated output by applying contrastive learning to the patchwise feature embedding space.

In addition, multi-domain models that enable arbitrary image transformation between various domains have also been developed. StarGAN [29] is the first work for multi-domain translation through a single common generator, and StarGANv2 [30] extended the multi-domain single-output translation capability of StarGAN into multi-domain diverse image translation.

In addition, various attribute manipulation methods leveraging the disentangled characteristic of StyleGAN have been proposed lately. These include a method to find specific direction for independent attribute by separating the principal component of the latent space [31], or to find a latent space that can edit a meaningful area of generated images by additionally using label information [32] or semantic map [33]. Since these StyleGAN-based processing methods inevitably require mapping the given image into StyleGAN’s latent space, various StyleGAN inversion methods have also been proposed. These include methods which train additional encoder model suitable for StyleGAN [34], [35], or latent optimization methods for a given image [36], [37], etc.

Recently, with the introduction of the CLIP model [7] by OpenAI, image manipulation methods through text conditions, such as text-to-image generation [38] and text-guided style transfer [39], have received a lot of attention. In addition, by exploiting the disentangled characteristic of StyleGAN, StyleCLIP [40] proposed a method for editing the latent attribute of StyleGAN through text, and StyleGAN-NADA [41] proposed a method of adapting the model to a novel domain through texts. Especially, StyleGAN-NADA further showed the versatility of model adaptation for a given target image.

C. Few-Shot Domain Adaptation

The purpose of adapting a generative model using few-shot images is to guide the model to a target domain while inheriting the diversity of source model trained on large-scale data. One of the first works of GAN adaptation is TransferGAN [1], which provided a novel insight of few-shot generation as a kind of transfer learning. Various subsequent few-shot GAN adaptation models focused on preventing the generator from overfitting by introducing regularization components during the fine-tuning process. FreezeD [2] performed GAN adaptation by freezing some layers of the discriminator in order to maintain the prior features of generator model. Others proposed to mix the weights of fine-tuned model and source model [42], or maintain singular vectors of the pre-trained weights [43]. In addition, EWC [5] used Fisher information to identify important layers of generator features and freeze them to prevent overfitting.

Recently, Ojha et al. [4] proposed an improved GAN adaptation method even in few-shot setting of less than 10 training samples by exploiting the cross-domain consistency and patch-based adversarial training. In the next version, Xiao et al. [44]
III. METHODS

By extending the existing approaches, our goal is to find improved regularization and training strategies, which can adapt the model robustly without over- and under-fitting problems regardless of the target image types or pre-trained source domains. One of the most important contributions of this work is the discovery of the importance of the unified CLIP space manipulation.

Specifically, our method consists of two steps. First, similar to the previous work [6], we search a latent code $w_{ref}$ in the latent space of the source domain generator $G_s$ so that it can generate reference image $I_{ref} = G_s(w_{ref})$ that is most similar to a given single-shot target image $I_{trg}$. However, in contrast to Mind the GAP [6] which uses pixel-wise similarity, we found that the CLIP space similarity significantly improves the search. The resulting image $I_{ref}$ is then used as a reference point to be aligned with $I_{trg}$ in the next step of model fine-tuning. In the second step, we fine-tune the pre-trained generator to create a target generator $G_t$ which follow the domain information of $I_{trg}$. At this time, our goal is to guide $G_t$ to maintain the diverse content attribute of $G_s$ by aligning the target image $I_{trg}$ and the reference image $I_{ref}$ obtained in the previous step. Again we reinforce this by using regularizations in the CLIP space to enforce semantic consistency between source and target generators. More detailed descriptions are as follows.

A. Step 1: Clip-Guided Latent Search

In order to find the image most consistent with the target image, the previous adaptation model Mind the Gap [6] leveraged the existing StyleGAN inversion method II2S [37]. However, we found that when the domain of the $I_{trg}$ (e.g. abstract sketch) is far from the source domain (e.g. FFHQ), the existing inversion model failed to reconstruct the attribute of $I_{trg}$.

We conjecture that this performance degradation is caused by the existing inversion model focusing only on pixel-wise similarity. To correct this, we propose to guide the generated $I_{ref}$ to follow the semantic attributes of $I_{trg}$ using the pre-trained CLIP model. More specifically, as shown in Fig. 2(left), we include additional loss that reduces the cosine distance between the CLIP space embeddings of the two images while using the pixel-wise loss between the $I_{ref}$ and $I_{trg}$. In addition, we employ the augmentation for $I_{ref}$ to avoid artifacts. The resulting optimization problem is formulated as:

$$\arg \min_{w \in W} D_{CLIP}(I_{ref}, I_{trg}) + ||I_{ref} - I_{trg}||_2 + \lambda_{reg} ||w - \bar{w}||_2,$$

where $I_{ref} = G_s(w)$, $D_{CLIP} \cdot \cdot$ is cosine distance between CLIP embeddings, $\lambda_{reg}$ is perceptual loss, and $\bar{w}$ is the averaged style code of the source generator $G_s$. Here, we additionally use $l_2$ regularization between $w$ and $\bar{w}$ to avoid reconstructing unrealistic images. Furthermore, to improve the efficiency of the search, rather than using random starting point for $w$, we set $\bar{w}$ as a starting point of $w$. After optimization step, we use the final latent codes $w$ as the reference latent $w_{ref}$.

The results of the proposed CLIP-guided optimization are shown in Fig. 2(right). When compared to the existing inversion methods II2S [37] and e4e [34], we can observe that the images from our method reflect the attributes (e.g. gender, component shape) of the target image while the baselines fail. Since our purpose is to obtain $I_{ref}$ representing the source domain while accurately having the attributes of the target image, the proposed method is more suitable in generating a reference. In Table 1, we quantitatively compare the inversion results. Although our method is not purposing on finding the latents for perfect reconstruction, our method shows decent performance when compared to baseline II2S in pixel-level metrics (MSE, LPIPS). When we compare the CLIP similarity score, our model outperformed the baseline. This indicates that our reconstructed outputs have higher semantic level similarity to the source images.

B. Step 2: Generative Model Fine-Tuning

1) Cross-Domain Semantic Consistency: In the next step, we create a target generator $G_t$ by fine-tuning the weights of the pre-trained generator. Recall that Ojha et al. [4] used a patch discriminator $D_{patch}$ so that the images generated by $G_t$ have the texture of the target image $I_{trg}$. However, the use of the...
and the target model \(G_t\) with the same latents \(w\). Then we obtain the embedded feature vector using a pre-trained CLIP model, and the cosine similarity scores between the embedded vectors in the CLIP space are calculated. The similarity scores between both domains \((c_i^0, c_i^0)\) are aggregated through the \(I_2\) regression. To guide the texture of the \(G_t\) to match the target \(I_{trg}\), we additionally use the patch discriminator.

discriminator alone easily leads to overfitting, so it is necessary to use additional regularization so that \(G_t\) can inherit the generation diversity of \(G_s\). To address this, the previous works attempted to match the distributions of the features between the source and target generators. Unfortunately, under the single-shot condition, such regularization based on the distribution of generator features does not work and still causes the overfitting.

Similar to Step 1, we found that CLIP space regularization solves the issues. Specifically, we found that maintaining the similarity distribution in the CLIP space as in Fig. 3 is more effective in preventing overfitting because it considers the semantic information of the generated images. Specifically, when arbitrary latent variables \([w_i]_n\) are sampled, we first calculate sample-wise similarity scores for \(G_s\) and \(G_t\) defined as:

\[
c_n = D_{CLIP}(G_s(w_i), G_s(w_j)),
\]

\[
c_n^t = D_{CLIP}(G_t(w_i), G_t(w_j)),
\]

where \(n\) is the reordered index of \((i, j)\), \(i \neq j\) and \(D_{CLIP}\) denotes the cosine similarity in the pre-trained CLIP embedding space. With the calculated similarity scores, the similarity loss between \(G_s\) and \(G_t\) can be computed. Different from the existing work [4] which use softmax with KL divergence for distribution matching, we simply used \(I_2\) distance between the similarity scores, since we observed that the use of KL divergence in the CLIP space reduces the training stability. Accordingly, our loss function for semantic consistency between \(G_s\) and \(G_t\) is formulated as:

\[
L_{con} = \frac{1}{w-p_c(w)} \sum_n ||c_n - c_n^t||_2.
\]

2) Patch-Wise Semantic Consistency: Although we can avoid the overfitting problem using the novel loss \(L_{con}\), the loss is a regularizer for the sample-wise semantics of the generated images so that their local features are ignored. Therefore, for better consistency of fine details between source and target generators, we propose a new patch-wise consistency loss.

To preserve the local attribute between two domains, we start from the idea of patch-wise contrastive loss (PatchNCE) in CUT [28]. Recall that CUT employs the contrastive learning to embed the features of the generator. Instead of using generator features directly, additional MLP header network is used in CUT to embed the features into another space. However, when we applied this directly to our framework, we observed that the training fails due to the imbalance between the header and the pre-trained generator. For this problem, we also found that CLIP is essential so that we propose to use the pre-trained CLIP model as a patch-wise embedding network.

Specifically, after cropping patches at random locations from the images generated by \(G_s\) and \(G_t\), we embed the image patches with the CLIP encoder as shown in Fig. 4(a). Then, we reduce the distance between the positive patches cropped at the same location, and push away the negative patches cropped from other locations. More specifically, if we set arbitrary location \(s_0\), the cropped patch of the outputs from the generators \(G_s\) and \(G_t\) are denoted as \([G_s(w)]_{s_0}\) and \([G_t(w)]_{s_0}\), respectively. Then if we set other \(N\) locations, we can obtain \(N\) patches from source domain as \([G_s(w)]_{s_i}\) where \(i \in \{1, \ldots, N\}\). With the cropped patches, we can calculate the patch-wise loss such as:

\[
L_{patch} = -\log \frac{\exp(v \cdot v^+)}{\exp(v \cdot v^+) + \sum_{i=1}^{N} \exp(v \cdot v_i^+)},
\]

where \(v = E([G_t(w)]_{s_0}), v^+ = E([G_s(w)]_{s_0})\) are positive vectors, and \(v^i = E([G_s(w)]_{s_i})\) are negative vectors. Here, \(E\) refers to the pre-trained CLIP encoder for the image patch, and the dot mark \(\cdot\) represents cosine similarity.

3) Reference Target Alignment: The losses suggested in the previous part serves to prevent overfitting when we sample arbitrary latents. This should be alternated with the additional loss for generating images through reference latent \(w_{ref}\) found in Step 1. Since \(I_{ref}\) obtained in Step 1 has the attribute most similar to \(I_{trg}\) and is an image representing the source domain, the domain-adapted output \(G_t(w_{ref})\) should be matched to \(I_{trg}\). Therefore, as shown in Fig. 4(b), we match the images in both of pixel and perceptual perspective as proposed in [6]. Also, we
further guide the \( G_t(w_{ref}) \) with global discriminator \( D_{glob} \) to make the image much closer to \( I_{trg} \).

4) Overall Training: Overall, our network is trained by alternately minimizing the two losses. First, when we use reference latent \( w_{ref} \), our loss is defined as:

\[
L_{ref} = \| G_t(w_{ref}) - I_{trg} \|_2 + \text{lpips}(G_t(w_{ref}), I_{trg}) + L_{adv}(G_t, D_{glob}),
\]

where StyleGAN2 adversarial loss is defined as \( L_{adv}^g(G, D_{glob}) = D_{glob}(G(w_{ref})) - D_{glob}(I_{trg}) \). For a global discriminator, we fine-tuned the pre-trained StyleGAN2 discriminator. Second, when arbitrary latents \( w \) are sampled, our loss is:

\[
L_{rand} = \lambda_{con}L_{con} + \lambda_{patch}L_{patch} + L_{adv}^p(G_t, D_{patch}).
\]

In this case, \( L_{adv}^p \) is defined as \( L_{adv}^p(G, D) = D_{patch}(G(w)) - D_{patch}(I_{trg}) \). For \( D_{patch} \), we used the proposed network in Ojha et al. [4], in which \( D_{patch} \) is a subset of \( D_{glob} \). More specifically, we extract the intermediate feature from \( D_{glob} \), then obtain the logit of \( D_{patch} \) with mapping the feature through several conv layers. Since simultaneously training using two losses is observed to reduce training stability and consume excessive memory, we therefore alternately train the model 3 iterations only using \( L_{rand} \) (Figs. 3, and 4(a)), then 1 iteration only using \( L_{ref} \) (Fig. 4(b)).

C. Text-Guided Image Manipulation

For text-guided image manipulation, we replaced the adversarial loss of our model with the directional CLIP loss. More specifically, in the first step we find the reference image \( I_{ref} \) that most closely matches the target text constraint \( t_{trg} \). Specifically, our goal is to find latent code \( w_{ref} \) for the source domain generator \( G_s \), as follows:

\[
w_{ref} := \arg \min_w \mathbb{E}_{I \sim W} [D_{CLIP}(I_{ref}, t_{trg}) + \lambda_{reg} \| w - \bar{w} \|_2],
\]

where \( I_{ref} = G_s(w) \), \( D_{CLIP}(\cdot, \cdot) \) is the cosine distance between CLIP embeddings, and \( \bar{w} \) is averaged style code of the source generator. After the optimization step, we use the final latent code \( w \) as the reference latent \( w_{ref} \).

Next, we fine-tune the generator model using the directional CLIP loss proposed in StyleGAN-NADA [41]:

\[
\Delta T = E_T(t_{trg}) - E_T(t_{src}),
\]

\[
\Delta I = E_I(G_t(w)) - E_I(G_s(w)),
\]

\[
L_{dir} = 1 - \frac{\Delta I \cdot \Delta T}{\| \Delta I \| \| \Delta T \|}.
\]

where \( E_T \) is text encoder of CLIP, and \( E_I \) is image encoder of CLIP, \( w \) is randomly sampled latent, and \( t_{source} \) is the source text condition representing the text description of the domain of \( G_s \). In our case, we used the text “Photo” as \( t_{src} \). In order to prevent overfitting problems, again we use our proposed consistency losses \( L_{con} \) and \( L_{patch} \).

Our full loss function for \( G_t \) fine-tuning is then defined as:

\[
L_{text} = L_{con} + L_{patch} + L_{dir} + L_{dir}^{ref},
\]

where \( L_{dir}^{ref} \) is the directional clip loss with using reference latent \( w_{ref} \) instead of sampled \( w \). Due to the absence of target images, we do not use discriminator network. We trained the model for 1,000 iterations.

Similar to Mind the GAP [6], instead of showing direct output \( G_t(w) \), we also use \( G_t(w_{mix}) \), where the mixed style code \( w_{mix} \) was constructed by replacing the last 11 vectors of sampled latent \( w \) with \( w_{ref} \).

IV. EXPERIMENTS

A. Experimental Details

Our training starts with the pre-trained StyleGAN2 [17]. To verify the versatility of our model, we conducted three different experiments using pre-trained generators trained on three datasets (FFHQ, LSUN church, LSUN cars). The models trained on FFHQ, LSUN church, and LSUN cars generate 1024x1024, 256x256, and 512x512 resolution images, respectively. We used the Adam optimizer [45] for all cases, and set the learning rate to 0.02. We used batch size 2 for FFHQ and LSUN cars, and batch size 4 for LSUN church. The training time took about 30 minutes with 2,000 iterations using a single RTX 2080 GPU. This is the time including 3 minutes of CLIP-guided optimization in step 1. For \( L_{patch} \) calculation, we used patch size of 128x128, 64x64, and 32x32 in adapting the pre-trained models on FFHQ, LSUN church, and LSUN cars, respectively. For hyperparameters, we set \( \lambda_{reg} \) in step 1 as 0.01, and set \( \lambda_{con}, \lambda_{patch} \) as 10, 1, respectively. Similarly to previous work [6], we froze ‘toRGB’ layer and mapping network in the refining stage.

For the patch discriminator \( D_{patch} \), we used a subset of \( D_{glob} \) as proposed by Ojha et al. [4]. More specifically, using a pre-trained discriminator of StyleGAN2, we first extract the intermediate features using the first residual conv layers of \( D_{glob} \), and then map the \( s \) to the final logit through an additional conv layer \( F^{\text{conv3x3-LeakyReLU}} \). In order to deal with various patch window sizes, multiple \( l \) and \( F^{l} \) were used. Specifically, when we adapt the model pre-trained with FFHQ, we used \( l = [5, 6] \); for the pretrained model with LSUN church [46], we used \( l = [3, 4] \); for the pretrained models with LSUN cars [47] and AFHQ dog [30], \( l = [4, 5] \) was used.

For the CLIP-guided latent optimization, in contrast to the baseline methods such as I2S [37] and e4e [34], which use extended StyleGAN latent space \( W^+ \in \mathbb{R}^{18x512} \), the optimization was performed in the single latent space in \( W \in \mathbb{R}^{512} \) since it produced better perceptual quality and preserved \( I_{trg} \) attribute. For inference, in the case of adapting FFHQ-pretrained model, we inverted the CelebA-HQ [9] images into latent codes \( w_{rec} \) using e4e [34] and generated images \( G_t(w_{rec}) \). In the case of LSUN church and LSUN cars models, we do not have proper validation image sets. Therefore we showed the results with same sampled latent \( w' \), in which the source image is \( G_s(w') \) and adapted outputs are \( G_t(w') \).

With the given latent code \( w \), we could show the adapted output with \( G_t(w) \). However, we observed that the generated images often have minor artifacts in background. To alleviate them, we used style mixing trick in which we replace the fine
Fig. 5. Comparison results with baseline models. The images in the uppermost row are the target images, and the images in the second row are source content images. The results in the following rows are generated by the models adapted with target images. Our results contain the texture of the target images while preserving the content attributes of the source images.

layer style code to average style vector $\bar{w}$ of $G_t$. In case of FFHQ model adaptation, instead of using $w$ which is composed of 18 vectors, we use mixed style code $w_{\text{mix}}$ in which the last 7 vectors are substituted to $\bar{w}$. In the LSUN car model fitting experiment, the last 7 vectors in the sampled code $w$, which consisted of 16 vectors, were replaced; and in adapting the LSUN church model, the last 6 vectors in the sampled code $w$, which consisted of 14 vectors, were replaced. For adapting the model with AFHQ dog, the last 9 vectors in the sampled code $w$, which consisted of 16 vectors, were replaced.

For training iterations, our default setting of training iteration is 2,000. However, if the content shape of target image is not much far from the source domain (e.g. realistic portrait), we trained the model with only 1,000 iterations. We selected perceptually better image between the outputs from models trained on 1,000 and 2,000 iterations. In case of adapting LSUN church models, we used 1,500 iterations since the model converged faster than the others.

For target images, we selected random image from the portrait face dataset [48] in FFHQ model adaptation. For others, we randomly collected publicly available images from the internet.

B. Qualitative Results

Fig. 5(up) shows the image result generated by the target generator adapted from the source generator pre-trained with
FFHQ. For comparison, we also show the images generated from the state-of-the-art baseline models: Few-shot GAN adaptation (FSA) [4], StyleGAN-NADA (NADA) [41], DualStyleGAN (DST) [49], TargetCLIP (TCL) [50], JoJoGAN [51], Relaxed Spatial Structural Alignment (RSSA) [44], and Mind the Gap (MTG) [6]. In all cases, the results show that the baseline models of few-shot GAN adaptation and StyleGAN-NADA suffer from overfitting problems, and in some cases, the model training for adaptation failed.

In one-shot style transfer model of JojoGAN and DualstyleGAN, the model outputs show proper style change, but the output images often have similar facial identities, which indicate the models still suffer from overfitting problems. For one-shot attribute transfer model of TargetCLIP, the model generated diverse outputs, but it does not contain the proper style of target images.

For RSSA, the model showed diverse generated outputs which indicates that the model partially solved the overfitting problem. However, the generated outputs still show some facial components (e.g. eyes), which indicate that the model could not fully address the overfitting problem.

In the case of Mind the Gap, unlike previous baselines, the model generates diverse images while preserving the content of the source domain images. However, the generated images did not properly reflect the domain texture of the target image in most cases, and only relatively easy characteristics such as global color or background were mainly changed. On the other hand, our model generates images that accurately reflect the complex texture (e.g. brushstroke, sketch line) of the target image while preserving the contents of the source domain images.

In order to verify the versatility of the proposed model, we conducted additional experiments with domain adapted models using source generator pre-trained with LSUN cars and LSUN church datasets. We also show the results from baselines models for comparison in Fig. 5(down). Again, we observed that our model perceptually improved resulted compared to other baselines. Since JojoGAN, DST, TargetCLIP, and RSSA model mostly concentrate on human face dataset, we did not include the qualitative outputs.

In Fig. 6, we show qualitative results from models trained with sketches, sunglasses, and babies datasets. Our model also outperforms baseline methods, showing successful translation outputs while preserving the content structure of the sources.

For further evaluation, we show additional results on adapting FFHQ pre-trained models in Fig. 7. In Fig. 8, we also show the results on adapting the models pre-trained on LSUN Car, LSUN church, and AFHQ dog datasets. The generated images have semantic style of the target images while preserving the content information of source images.

C. Quantitative Results

For evaluation of our proposed model, we conducted quantitative experiments in Table II. First, we measured the generation quality with FID [52] scores. For fair comparison, we performed adaptation for sketches [53], FFHQ-sunglasses, and -babies data [11], which are benchmarks for GAN adaptation provided in Few-shot adaptation [4]. For Babies, it contains 2,492 images, and for Sunglasses dataset, it has 2,683 images. For sketches dataset, it has 295 images. To experiment with the one-shot setting, we randomly selected 5 images from each dataset and used them as targets. With each adapted model, we randomly generated 1,000 samples and calculated FID between the real and generated data. We report average score of 2 training runs for each target image (total 10 runs for each score). For the experiments of all baseline methods, we also used the same...
Fig. 8. Additional results with adapting the models pre-trained on LSUN Cars (Up), LSUN church (Middle), and AFHQ Dog (Bottom) datasets. The images in the uppermost row are the source content images. The results in the following rows are generated by the models adapted with the target images.

TABLE II
QUANTITATIVE RESULTS ON VARIOUS METRICS OF FID, LPIPS, AND IDENTITY SCORE BLUE-SECOND BEST, RED-BEST

| Dataset  | FID↓ | LPIPS↑ | ID↑ |
|----------|------|--------|-----|
| TGAN     | 116.8| 96.92  | 152.7| 0.336| 0.100|
| Mine     | 117.9| 85.26  | 168.2| 0.413| 0.150|
| FD       | 128.5| 108.1  | 176.3| 0.392| 0.146|
| FSA      | 145.8| 90.34  | 119.6| 0.437| 0.163|
| NADA     | 134.8| 137.8  | 102.7| 0.435| 0.175|
| Jojo     | 128.9| 121.0  | 117.6| 0.519| 0.376|
| TCL      | 147.3| 72.93  | 105.8| 0.536| 0.400|
| DST      | 105.0| 68.56  | 141.7| 0.507| 0.243|
| MTG      | 107.2| 73.34  | 123.6| 0.554| 0.343|
| RSSA     | 112.7| 108.6  | 123.8| 0.598| 0.396|
| Ours     | 83.87| 64.61  | 105.2| 0.568| 0.403|

dataset and followed same experimental protocol for reporting FID scores. In the experiments for FFHQ-babies and sunglasses, we used smaller parameters ($\lambda_{con} = 0.1, \lambda_{patch} = 0.01$) as it showed better performance.

In order to further evaluate the generation diversity, we also calculated intra-sample LPIPS distance [54]. Following the experimental protocol used in our baseline paper [4], for each adapted model, we randomly generate two batches and calculated LPIPS distance between the batches. We used averaged score of 1,000 times of LPIPS distances between the batches, and each batch contains 4 samples. We experimented on the models shown in our main paper, which are adapted on portrait painting dataset. For thorough evaluation, we selected 10 different models for each method and report the averaged scores.

To evaluate the content preservation performance, we report the face identity score [55] which are calculated between the images generated from both models $G_s(z)$ and $G_t(z)$, in which $z$ are the same latent codes.

As baselines, we adopted several few-shot adaptation models such as TransferGAN (TGAN) [1], FreezeD (FD) [2], MineGAN (Mine) [56], Few-shot adaptation (FSA) [4], StyleGAN-NADA (NADA) [41], DualStyleGAN (DST) [49], Target-CLIP (TCL) [50], JoJoGAN [51], Relaxed Spatial Structural Alignment (RSSA) [44], and Mind the Gap (MTG) [6]. The quantitative results show that our model obtained the best FID score in sketches and sunglasses datasets, and the second best in the babies dataset. Our model also scored the best in LPIPS generation diversity and identity scores. The results show that our model shows the best performance among all baselines, with respect to generation quality, diversity, and content preservation.

D. User Study

For further evaluation of our proposed model, we additionally conducted a user study in Table III. To quantitatively measure the detailed preference from users, we used a custom-made mean opinion scoring system. As baselines, we used the models of Few-shot adaptation, StyleGAN-NADA, and Mind the Gap.

For user study, we provided the generated images to users, and asked them to score the images according to three evaluation criteria: 1) Do the generated images properly reflect the domain style of the target image? 2) Does the content of the generated images properly match with the source domain images? 3) Do the generated images have sufficient diversity? Users can choose the scores among 5 options: 1-very bad, 2-bad, 3-neutral, 4-good, 5-very good. We randomly recruited 20 users using Google Form, who come from the age group between 20 s and 40 s. For the generated images used in our user study, we fine-tuned 10 models using 10 different target images per each methods. For each fine-tuned models, we generated 10 different images using the same source images per each models. Therefore, we show total 400 images (10 fine-tuned models x 10 generated images x 4 methods) to each user. For fair comparison, we collected...
user opinions using fine-tuned models with the same target and source images.

In Table III, we can see that our model outperforms all baseline models in terms of perceptual preference scores.

For detailed statistical analysis, we show the individual mean score for each questions in Fig. 9. Our user study has total of 8 questions. In overall questions, our proposed model obtained higher scores compared to the baseline methods.

E. Text-Guided Image Manipulation Results

In Fig. 10, we show comparison results of model adaptation with single text condition. Although baseline StyleGAN-NADA can modulate the overall texture of outputs, the generated faces have almost same identity regardless of the input images, suggesting that the model suffers from overfitting. In contrast, our model can synthesize reasonable outputs that reflect both text condition and source content. The results showed that our CLIP space regularization can avoid overfitting, further confirming that our proposed model have a strong advantage in versatility.

To further show the superiority of our text-guided image manipulation model, we conducted another user study. We followed the same protocol used in our one-shot adaptation experiments. Again, we provided the generated images to users, and asked them to score the images according to three evaluation criteria: 1) Do the generated images properly reflect the domain style of the target text? 2) Does the content of the generated images properly match with the source domain images? 3) Do the generated images have sufficient diversity?

Users can choose the scores among 5 options: 1-very bad, 2-bad, 3-neutral, 4-good, 5-very good. We randomly recruited 20 users using Google Form, who come from the age group between 20 s and 40 s. We provided users with 40 generated images per each text condition. We compared our model with StyleGAN-NADA. Table IV shows that our text-guided model outperforms StyleGAN-NADA in human perception scores.

F. Latent Space Editing

Recall that our purpose is to train a fine-tuned model $G_t$ to generate images with the target texture while preserving the latent space characteristics of the source domain. To verify this, we tested whether the latent space of $G_t$ sufficiently includes the disentangled characteristic of the source domain through attribute editing.

In this part, we adopted the latent space editing technique in StyleCLIP [40], which is a state-of-the-art attribute editing method. With StyleCLIP, we can edit the face attributes with various text conditions. Among the three editing approaches proposed in StyleCLIP, we used the editing through global direction which can be universally applied. When there is a latent code $w$ corresponding to the source image, we can manipulate it with StyleCLIP to get a edited latent $\hat{w}$, then we put the code as an input to the fine-tuned generator $G_t$ to get the attribute edited output in the target domain.
Fig. 12. Latent space editing results. We manipulated the attribute of the faces using a tool in the StyleCLIP. The results show that adapted model \( G_t \) still maintains the latent space characteristic of source domain generator.

Fig. 13. Ablations study results. (a) The absence of \( L_{con} \) leads to overfitting, generating the same face identity regardless of the input images. (b) Without \( L_{patch} \), detailed component shapes are misaligned with the source image. (c) When II2S is used for reference research instead of our proposed CLIP guided optimization (CLIPopt), several facial attributes are not correctly reflected. (d) Generated images from \( G_t \) adapted using \( L_{con} \) with Softmax and KL divergence. (e) Generated images from \( G_t \) adapted using consistency loss on generator features [4] instead of our proposed CLIP space consistency \( L_{con} \). (f) Results from \( G_t \) adapted using contrastive regularization proposed in CUT [28] instead of our \( L_{patch} \).

We show experimental results of attribute editing in Fig. 12. When editing through various text conditions, we show that domain-adapted generator \( G_t \) can also generate edited outputs without entanglement. This shows that our adapted generator maintains the content characteristics of the source domain.

G. Ablation Study

1) Qualitative Comparison: In order to check how the various components proposed in our method operate, we conducted ablation studies by removing each component in Fig. 13.

Specifically, (a) when our proposed semantic consistency loss \( L_{con} \) is removed, we can see that the facial attributes of the generated faces have identical shape due to the overfitting of adapted model.

(b) When removing the patch-wise consistency loss \( L_{patch} \), we can see the generated faces have different face identities from the source domain images. Although using only the suggested \( L_{con} \) can prevent overfitting, we found that when \( L_{patch} \) is not used, the generated images are excessively deformed due to lack of consistency between the local regions of source and output images.

(c) To evaluate the superiority of our proposed CLIP-guided optimization, we trained the model using baseline II2S [37] instead of our the proposed CLIP space optimization. At this time, the model generates decent result images, but we can see that there are misalignments between the attributes (e.g., eyes, shape, beard) of the generated and source images. We conjecture that the problem occurs because the reference image generated with II2S does not properly cover the attribute of the target image.

(d) When we use \( L_{con} \) composed of Softmax & KL Divergence instead of our proposed \( L_2 \) regression, we observed that the results suffer from severe overfitting as the loss does not work properly.

(e) Next, when we use the consistency loss on generator features proposed by Ojha et al. [4] instead of our CLIP-space consistency loss \( L_{con} \), the model suffers from severe overfitting due to training failure.

(f) We show the results using contrastive regularization on generator features with additional header network \( F \) similar to CUT [28], instead of using our \( L_{patch} \). In this case, we observed that overfitting problem is resolved as \( L_{con} \) correctly operates, but the local features of output images are not properly matched to the source images, due to training imbalance between generator and header networks. We can see that using our full settings can preserve the attribute of source images the best.

2) Quantitative Comparison: In Table V, we show quantitative ablation study results for our main proposed components. We measured various quantitative metrics with adapted models on sketches dataset. We followed the same experimental settings of our main quantitative experiments. For additional study, we include two different ablation settings: (g) To show the effectiveness of reference-target matching, we experimented with removing the matching between \( I_{ref} \) and \( I_{trg} \). (h) To further evaluate our proposed mixing trick, we measured the performance with removing the mixing trick. When the proposed components are removed from our full settings, the models showed degraded performance in all aspects including generation quality (FID), generation diversity (LPIPS), and content preservation (ID). In contrast, our full settings showed the best quantitative performance among all baseline settings.

For further ablation study, we show additional results in Table VI. When we use smaller regularization loss (\( \lambda_{con} = 1, \lambda_{patch} = 0.1 \)), our model shows degraded diversity and identity score, which indicate that the model shows overfitting. When using larger regularization loss (\( \lambda_{con} = 100, \lambda_{patch} = 10 \)), the model shows degraded score in FID. With using different layers...
TABLE VI
ABLATION STUDY RESULTS ON HYPERPARAMETER SETTINGS

| Metrics | FID↓ | LPIS↑ | ID↑ |
|---------|------|-------|-----|
| $\lambda_{con} = 1$ | 89.87 | 0.506 | 0.211 |
| $\lambda_{con} = 100$ | 86.51 | 0.536 | 0.490 |
| $\lambda_{patch} = 0.1$ | 82.82 | 0.476 | 0.320 |
| $\lambda_{patch} = 10$ | 103.09 | 0.543 | 0.475 |
| $D_{patch,l} = [3, 4]$ | 90.64 | 0.547 | 0.441 |
| $D_{patch,l} = [7, 8]$ | 86.91 | 0.546 | 0.428 |
| Ours-best | 83.87 | 0.572 | 0.443 |

Our best setting showed the best quantitative score.

TABLE VII
QUANTITATIVE RESULTS ON 3-SHOT AND 10-SHOT ADAPTATION TASKS

| Metrics | 3-shot | 10-shot |
|---------|--------|---------|
| | FID↓ | LPIS↑ | ID↑ | FID↓ | LPIS↑ | ID↑ |
| FSA | 78.15 | 0.488 | 0.229 | 63.85 | 0.400 | 0.142 |
| MTG | 101.2 | 0.565 | 0.326 | 94.62 | 0.533 | 0.225 |
| Ours | 70.13 | 0.567 | 0.410 | 71.53 | 0.556 | 0.401 |

For evaluation, we use sketches dataset. Our model outperformed few-shot adaptation baselines.

Fig. 14. Additional comparison results with state-of-the-art image style transfer models. The result from our model reflects accurate semantic information of the target image while baseline models fail.

Fig. 15. Hierarchical layer-wise stylization outputs. If we stylize the deep layer, the output identity follows the target. With middle layer stylization, the output images contain mixed features of source and target images. With shallow layer, the outputs follow the style of target images, while preserving the content of sources.

J. Layer-Wise Stylization

To further show the possibility of selective stylization of our proposed method, we show the experiment result in Fig. 15. Originally, our model is not purposing on layer-selective style transfer since we fine-tune the whole model parameters using target image. However, we can obtain the selective stylization outputs with substituting the latents of selective layers into our inverted latent $w_{ref}$. We set the low-resolution deep layers as layer 1 - 6, middle layers as layer 7 - 12, and shallow layers as layer 13 - 18.

When we change the deep layer style, we can see that the facial identity of output images mostly follow the target identity. For middle layer stylization, the outputs have both of source and target facial attributes. In shallow layer stylization, the outputs have style of target images, while preserving the content of sources. Regardless of selected layers, since our method include fine-tuning the whole parameters, the output images contain the overall style of target domain.

K. Failure Cases

Although our proposed method shows state-of-the-art performance in content-preserving one-shot domain transfer, our method shows limited performance in several extreme cases. Since our method rely on pre-trained generator model, the output quality can vary according to the target images. As shown in Fig. 16, if we use targets which are significantly far from the source domain (e.g., human face $\rightarrow$ artistic painting), the model performance is degraded. In order to solve the problem, as a future work, we will extend our method to global natural image domain using the generator pre-trained on ImageNet data.

SANet [58], and CST [59]. In the case of the existing style transfer models, the model failed to reflect the semantic information of target images, as they excessively concentrate on applying the overall color and texture of the target images. We can see that there are several artifacts such as the background color of the target image applied to the face, or the background color applied to entire image areas as shown in Fig. 14. In the case of our model, we can see that the generated images accurately reflect the characteristics of target domain by considering both the texture and semantic information of the targets.
V. DISCUSSION AND CONCLUSION

In this article, we proposed a new framework that can transform a pre-trained StyleGAN to generate target domain images by fine-tuning with a single target image. The key idea is the CLIP space manipulation through a two step approach. Specifically, we proposed an optimization method in CLIP space to find the reference image which have the most similar attribute to the target image in the source domain. The reference image was then used as an anchor point to maintain the content attribute of target generator. In addition, to prevent model overfitting, we proposed two different regularization losses: semantic consistency loss and patch-wise consistency both in CLIP space. Our experimental results showed that the proposed method leads to better quantitative and qualitative results than the existing methods.

Our model showed superior performance compared to the baseline models in one-shot domain adaptation experiments, and the results on various adapted models further confirmed that our model has enough versatility. As a further extension, we are planning to try domain adaptation on generative models for natural images. In addition, we plan to explore whether our proposed regularization can be applied to other generative framework such as diffusion models.

REFERENCES

[1] Y. Wang, C. Wu, L. Herranz, J. van de Weijer, A. Gonzalez-Garcia, and B. Raducanu, “Transferring GANs: Generating images from limited data,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 218–234.
[2] S. Mo, M. Cho, and J. Shin, “Freeze the discriminator: A simple baseline for fine-tuning GANs,” 2020, arXiv:2002.10964.
[3] A. Noguchi and T. Harada, “Image generation from small datasets via batch statistics adaptation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 2750–2758.
[4] U. Ojha et al., “Few-shot image generation via cross-domain correspondence,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 10 743–10 752.
[5] Y. Li, R. Zhang, J. Lu, and E. Shechtman, “Few-shot image generation with elastic weight consolidation,” 2020, arXiv:2012.02780.
[6] P. Zhu, R. Abdal, J. Fosser, and P. Wonka, “Mind the gap: Domain gap control for single shot domain adaptation for generative adversarial networks,” 2021, arXiv:2110.08598.
[7] A. Radford et al., “Learning transferable visual models from natural language supervision,” 2021, arXiv:2103.00020.
[8] I. J. Goodfellow et al., *Generative Adversarial Nets*, ser. NIPS’14. Cambridge, MA, USA: MIT Press, 2014, pp. 2672–2680.
[9] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of GANs for improved quality, stability, and variation,” in Proc. Int. Conf. Learn. Representations, 2018.
[10] A. Brock, J. Donahue, and K. Simonyan, “Large scale GAN training for high fidelity natural image synthesis,” 2018, arXiv:1809.11096.
[11] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 4401–4410.
[12] S. Belousov, “MobileStyleGAN: A lightweight convolutional neural network for high-fidelity image synthesis,” 2021, arXiv:2104.04767.
[13] Y. Alharbi and P. Wonka, “Disentangled image generation through structured noise injection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 5134–5142.
[14] G. Kwon and J. C. Ye, “Diagonal attention and style-based GAN for content-style disentanglement in image generation and translation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 13 980–13 989.
[15] H. Kim, Y. Choi, J. Kim, S. Yoo, and Y. Uh, “Exploiting spatial dimensions of latent in GAN for real-time image editing,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 852–861.
[16] S. Pidhorskyi, D. A. Adjeroh, and G. Doretto, “Adversarial latent autoencoders,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 14 104–14 113.
[17] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of StyleGAN,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8110–8119.
[18] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, “Training generative adversarial networks with limited data,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 12 104–12 114.
[19] Y. Zhang et al., “DatasetGAN: Efficient labeled data factory with minimal human effort,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 10 145–10 155.
[20] R. Abdal, P. Zha, Y. J. Mitra, and P. Wonka, “Labels4Free: Unsupervised segmentation using StyleGAN,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 13 970–13 979.
[21] A. Tewari et al., “StyleRig: Rigging StyleGAN for 3D control over portrait images,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 6142–6151.
[22] Y. Shi, D. Angarwal, and A. K. Jain, “Lifting 2D StyleGAN for 3D-aware face generation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 6258–6266.
[23] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, “Pulse: Self-supervised photo upsampling via latent space exploration of generative models,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 2437–2445.
[24] T. Yang, P. Ren, X. Xie, and L. Zhang, “GAN prior embedded network for blind face restoration in the wild,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 672–681.
[25] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 2223–2232.
[26] H. Fu, M. Gong, C. Wang, K. Batmanghelich, K. Zhang, and D. Tao, “Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 2427–2436.
[27] S. Benaim and L. Wolf, “One-sided unsupervised domain mapping,” in Proc. Adv. Neural Inf. Process. Syst., I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017, pp. 752–762. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/5990e1005a220ee2ebc542eb9d9eb1e-Paper.pdf
[28] T. Park, A. A. Efros, P. Zhang, and J.-Y. Zhu, “Contrastive learning for unpaired image-to-image translation,” in Proc. Eur. Conf. Comput. Vis., A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., Cham, Springer International Publishing, 2020, pp. 319–345.
[29] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, “StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 8789–8797.
[30] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, “StarGAN v2: Diverse image synthesis for multiple domains,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 8188–8197.
[31] E. Häkkinen, A. Hertzmann, J. Lehtinen, and S. Paris, “GANSpace: Discovering interpretable GAN controls,” in Proc. Adv. Neural Inf. Process. Syst., H. Larochelle, M. Ranzato, R. Hodell, M. P. Balcan, and H. Lin, Eds., Curran Associates, Inc., 2020, pp. 9841–9850. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/66ef43209667adbb64e6149852b5cc3e-Paper.pdf
[32] Y. Shen, J. Gu, X. Tang, and B. Zhou, “Interpreting the latent space of GANs for semantic face editing,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Los Alamitos, CA, USA, 2020, pp. 9240–9249. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPR42600.2020.00926

[33] O. Kafri, O. Patashnik, Y. Alaluf, and D. Cohen-Or, “StyleFusion: A generative model for disentangling spatial segments,” 2021, arXiv:2107.07437.

[34] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, “Designing an encoder for StyleGAN image manipulation,” ACM Trans. Graph., vol. 40, no. 4, pp. 1–14, 2021.

[35] E. Richardson et al., “Encoding in style: A styleGAN encoder for image-to-image translation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 2287–2296.

[36] R. Abdal, Y. Qin, and P. Wonka, “Image2StyleGAN: How to embed images into the StyleGAN latent space?,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 4432–4441.

[37] P. Zhu, R. Abdal, Y. Qin, J. Femiani, and P. Wonka, “Improved StyleGAN embedding: Where are the good latents?,” 2020, arXiv:2012.09036.

[38] X. Liu, C. Gong, L. Wu, S. Zhang, H. Su, and Q. Liu, “FuseDream: Training-free text-to-image generation with improved clip+ GAN space optimization,” 2021, arXiv:2112.01573.

[39] G. Kwon and J. C. Ye, “CLIPstyler: Image style transfer with a single text condition,” 2021, arXiv:2112.00374.

[40] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, and D. Lischinski, “Styleclip: Text-driven manipulation of StyleGAN imagery,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 2085–2094.

[41] R. Gal, O. Patashnik, H. Maron, G. Chechik, and D. Cohen-Or, “StyleGAN-NADA: Clip-guided domain adaptation of image generators,” 2021, arXiv:2108.09946.

[42] J. N. Pinkney and D. Adler, “Resolution dependent GAN interpolation for controllable image synthesis between domains,” 2020, arXiv:2010.05334.

[43] E. Robb, W.-S. Chu, A. Kumar, and J.-B. Huang, “Few-shot adaptation of generative adversarial networks,” 2020, arXiv:2010.11943.

[44] J. Xiao, L. Li, C. Wang, Z.-J. Zha, and Q. Huang, “Few shot generative model adaption via relaxed spatial structural alignment,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 1194–11203.

[45] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[46] F. Yu, Y. Zhang, S. Song, A. Seff, and J. Xiao, “LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop,” 2015, arXiv:1506.03365.

[47] T. Kramberger and B. Potočnik, “LSUN-standford car dataset: Enhancing large-scale car image datasets using deep learning for usage in GAN training,” Appl. Sci., vol. 10, no. 14, Jul. 2020, Art. no. 4913. [Online]. Available: https://www.mdpi.com/2076-3417/10/14/4913

[48] M. Affifi, M. A. Brubaker, and M. S. Brown, “HistOGAN: Controlling colors of GAN-generated and real images via color histograms,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 7941–7950.

[49] S. Yang, L. Jiang, Z. Liu, and C. C. Loy, “Pastiche master: Exemplar-based high-resolution portrait style transfer,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 7693–7702.

[50] H. Chefer, S. Benaim, R. Paris, and L. Wolf, “Image-based clip-guided essence transfer,” in Proc. 17th Eur. Conf. Comput. Vis., Tel Aviv, Israel, Springer, Oct. 23–27, 2022, pp. 695–711.

[51] M. J. Chong and D. Forsyth, “JoLoGAN: One shot face stylization,” 2021, arXiv:2112.11641.

[52] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “GANs trained by a two time-scale update rule converge to a local Nash equilibrium,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 6626–6637.

[53] X. Wang and X. Tang, “Face photo-sketch synthesis and recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 11, pp. 1955–1967, Nov. 2009.

[54] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 586–595.

[55] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “ArcFace: Additive angular margin loss for deep face recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 4690–4699.

[56] Y. Wang, A. Gonzalez-Garcia, D. Berga, L. Herranz, F. S. Khan, and J. V. D. Weijer, “MineGAN: Effective knowledge transfer from GANs to target domains with few images,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 9332–9341.

[57] S. Liu et al., “AdaAttN: Revisit attention mechanism in arbitrary neural style transfer;” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 6649–6658.

[58] D. Y. Park and K. H. Lee, “Arbitrary style transfer with style-attentional networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 5880–5888.

[59] J. Svoboda, A. Anoosheh, C. Osendorfer, and J. Masri, “Two-stage peer-regularized feature recombination for arbitrary image style transfer,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 13 816–13 825.

Gibyun Kwon received the BS degree from the Department of Electronic Engineering, Hanyang University, Seoul, South Korea, in 2018 and the MS degree from the Department of Electric Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea in 2020. Currently he is working toward the PhD degree with the Department of Bio and Brain Engineering, Korea Advanced Institute of Science and Technology. His research interests include machine learning for image processing, especially for generative models. He authored and coauthored multiple papers in top conference proceedings including ICLR, CVPR, ICCV, MICCAI, and BMVC.

Jong Chul Ye (Fellow, IEEE) received the BSc and MSc degrees from Seoul National University, Korea, and the PhD degree from Purdue University, West Lafayette. He is a professor of the Graduate School of Artificial Intelligence (AI) of Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea in 2020. Currently he is working toward the PhD degree with the Department of Bio and Brain Engineering, Korea Advanced Institute of Science and Technology. His research interests include machine learning for image processing, especially for generative models. He authored and coauthored multiple papers in top conference proceedings including ICLR, CVPR, ICCV, MICCAI, and BMVC.