Spatial-Contextual Discrepancy Information Compensation for GAN Inversion

Ziqiang Zhang[1], **Yan Yan**[1*], **Jing-Hao Xue**[2], **Hanzi Wang**[1]
[1]Xiamen University, China
[2]University College London, UK
zhangzq@stu.xmu.edu.cn, yanyan@xmu.edu.cn, jinghao.xue@ucl.ac.uk, hanzi.wang@xmu.edu.cn

Abstract

Most existing GAN inversion methods either achieve accurate reconstruction but lack editability or offer strong editability at the cost of fidelity. Hence, how to balance the distortion-editability trade-off is a significant challenge for GAN inversion. To address this challenge, we introduce a novel spatial-contextual discrepancy information compensation-based GAN-inversion method (SDIC), which consists of a discrepancy information prediction network (DIPN) and a discrepancy information compensation network (DICN). SDIC follows a "compensate-and-edit" paradigm and successfully bridges the gap in image details between the original image and the reconstructed/edited image. On the one hand, DIPN encodes the multi-level spatial-contextual information of the original and initial reconstructed images and then predicts a spatial-contextual guided discrepancy map with two hourglass modules. In this way, a reliable discrepancy map that models the contextual relationship and captures fine-grained image details is learned. On the other hand, DICN incorporates the predicted discrepancy information into both the latent code and the GAN generator with different transformations, generating high-quality reconstructed/edited images. This effectively compensates for the loss of image details during GAN inversion. Both quantitative and qualitative experiments demonstrate that our proposed method achieves the excellent distortion-editability trade-off at a fast inference speed for both image inversion and editing tasks. Our code is available at https://github.com/ZzqLKED/SDIC.

Introduction

Over the past few years, a variety of powerful GAN models, such as PGGAN (Karras et al. 2017) and StyleGAN (Karras, Laine, and Aila 2019; Karras et al. 2020), have been developed to generate high-quality images based on the latent code in the latent space. Based on these models, we can manipulate some attributes of generated images by modifying the latent code. However, such manipulation is only applicable to the images given by the GAN generator due to the lack of inference capability in GANs (Xia et al. 2022).

Recently, GAN inversion methods (Xia et al. 2022) have been proposed to manipulate real images. These methods usually follow an "invert first, edit later" procedure, which first inverts the real image back into a latent code of a pre-trained GAN model, and then a new image can be reconstructed or edited from the inverted latent code.

Some GAN inversion methods (Richardson et al. 2021; Tov et al. 2021) invert a real image into the native latent space of StyleGAN (i.e., the $\mathcal{W}$ space) and achieve good editability. However, such a way inevitably leads to the loss of image details during inversion. As a result, the reconstructed images are often less faithful than the original images. Although some methods (Abdal, Qin, and Wonka 2019; Kang, Kim, and Cho 2021) extend the $\mathcal{W}$ space to the $\mathcal{W}^+$/$\mathcal{W}$* space or perform per-image optimization to enhance the reconstruction fidelity, their editability can be greatly affected. Such a phenomenon is also known as the distortion-editability trade-off, indicating the conflict between image reconstruction fidelity and image editing quality.

To alleviate the distortion-editability trade-off, some recent methods (such as HFGI (Wang et al. 2022) and CLCAE (Liu, Song, and Chen 2023)) recover the missing information by enriching the latent code and the feature representations of a particular layer in the GAN generator. In this way, they can achieve better fidelity than traditional GAN inversion methods. However, HFGI considers the distortion information only at the pixel level, which easily introduces significant artifacts; CLCAE generates images based on contrastive learning, and it may still lose some image details and degrade the editability. Therefore, existing GAN inversion methods still suffer from the gap in image details between the original image and the reconstructed/edited image.

To address the above problems, we propose a novel spatial-contextual discrepancy information compensation-based GAN inversion method (SDIC), which consists of a discrepancy information prediction network (DIPN) and a discrepancy information compensation network (DICN). SDIC adopts a "compensate-and-edit" paradigm, which first compensates both the latent code and the GAN generator with the spatial-contextual guided discrepancy map, and then performs image inversion or editing.

Specifically, DIPN, which consists of a two-branch spatial-contextual hourglass module and a discrepancy map learning hourglass module, is designed to encode the multi-level spatial-contextual information of the original and initial reconstructed images, and predict a spatial-contextual guided discrepancy map. In DIPN, a spatial attention mech-

*Corresponding author.
Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

anism is leveraged to enable the network to adaptively select important parts for feature fusion. As a result, DIPN can accurately learn a reliable discrepancy map, which effectively captures the contextual relationship and fine-grained image details. Then, DICN is introduced to incorporate the discrepancy information into both the latent code and the GAN generator, generating high-quality reconstructed/edited images.

In summary, our main contributions are given as follows:

- We propose a novel GAN inversion method, which successfully exploits the multi-level spatial-contextual information of the original image to compensate for the missing information during inversion. Based on the "compensate-and-edit" paradigm, our method can generate high-quality and natural reconstructed/edited images containing image details without introducing artifacts.
- We design DIPN to accurately predict the spatial-contextual guided discrepancy map with two hourglass modules and DICN to leverage this map to effectively compensate for the information loss in both the latent code and the GAN generator with different transformations. Therefore, our method can achieve an excellent distortion-editability trade-off.
- We perform qualitative and quantitative experiments to validate the superiority of our method in fidelity and editability against state-of-the-art GAN inversion methods.

The full version of this paper, including appendix, can be found at https://arxiv.org/abs/2312.07079.

Related Work

GAN Inversion. Existing GAN inversion methods can be divided into three categories: optimization-based, encoder-based, and hybrid methods. Optimization-based methods (Abdal, Qin, and Wonka 2020; Bau et al. 2020; Gu, Shen, and Zhou 2020; Zhu et al. 2020b) directly optimize the latent code or the parameters of GAN to minimize the reconstruction error for each given image. Although these methods can reconstruct high-fidelity images, they usually suffer from high computational cost and poor editability. Encoder-based methods (Alaluf, Patashnik, and Cohen-Or 2021; Hu et al. 2022; Kang, Kim, and Cho 2021; Tov et al. 2021) train an encoder to learn the mapping from a given image to a latent code and perform some operations on the latent code. Compared with the optimization-based methods, the encoder-based methods show better editability at a faster inference speed, but their reconstruction quality is much lower. Hybrid methods (Bau et al. 2019; Zhu et al. 2020a) leverage an encoder to learn a latent code and then optimize the obtained latent code. Recently, some methods (Alaluf et al. 2022; Dinh et al. 2022) introduce the hypernetwork to iteratively optimize the parameters of the GAN generator, obtaining better reconstruction results.

Our method belongs to the encoder-based methods. Conventional encoder-based methods (such as pSp (Chang and Chen 2018) and e4e (Tov et al. 2021)) extend the $\mathcal{W}$ space to the $\mathcal{W}^+/\mathcal{W}*$ space to improve the reconstruction fidelity. However, the low-dimensional latent code still limits the reconstruction performance. Moreover, the editing flexibility of the $\mathcal{W}^+/\mathcal{W}*$ space is reduced. In contrast, we propose to compensate the latent code and the GAN generator by exploiting the multi-level spatial-contextual information. In this way, the expressiveness of the latent code and generator is largely improved, alleviating the information loss due to inversion. Meanwhile, we explicitly control the proximity of the latent code to the $\mathcal{W}$ space, providing better editability.

Latent Space Editing. The latent spaces of pre-trained StyleGAN generators show good semantic interpretability, which enables diverse and flexible image editing. A number of methods have been developed to identify meaningful semantic directions for the latent code. Supervised methods (Abdal et al. 2021; Goetschalckx et al. 2019; Hutchinson et al. 2019; Shen et al. 2020) require pre-trained attribute classifiers or data with attribute annotations. InterFaceGAN (Shen et al. 2020) employs a support vector machine to identify the hyperplane that splits two binary attributes and considers the normal of the hyperplane as the manipulation direction. Unsupervised methods (Härkönen et al. 2020; Shen and Zhou 2021; Voynov and Babenko 2020; Wang and Ponce 2021) can discover unknown manipulation directions but require manual labeling of the found operational directions. GANspace (Härkönen et al. 2020) discovers multiple manipulation directions using principal component analysis. In this paper, we employ InterFaceGAN and GANspace for latent space editing due to their good editing performance.

Spatial-Contextual Information. Convolutional neural network (CNN) encodes both low-level features at the early stages and high-level features at the later stages. The low-level features are rich in spatial details while the high-level features capture the contextual information, which encodes the visual knowledge from an object/patch and its surrounding backgrounds/patches. The spatial-contextual information plays an important role in many computer vision tasks, such as object detection and semantic segmentation. Some methods (Choi et al. 2010; Li et al. 2016) exploit the contextual information between the object and its surrounding background to improve the object detection performance. A few works (Chang and Chen 2018) improve the quality of the detailed parts of the disparity map by exploiting multi-scale spatial-contextual information.

Unfortunately, the spatial-contextual information is not well exploited in exisiting GAN inversion methods. In this paper, we introduce the spatial-contextual information (obtained from the original image) to the discrepancy map prediction between the original image and the initial reconstructed image. In this way, our method can preserve more appearance details and generate clearer edges, greatly reducing the artifacts of the reconstructed/edited images.

Proposed Method

Overview

Motivation. Conventional GAN inversion methods (Alaluf, Patashnik, and Cohen-Or 2021; Collins et al. 2020; Kang, Kim, and Cho 2021; Pidhorskyi, Adjeroh, and Doretto 2020; Tov et al. 2021) invert a real image into the latent space of a pretrained GAN model and attain good editability. However, they usually suffer from the low fidelity of generated images due to severe information loss during the inversion process.

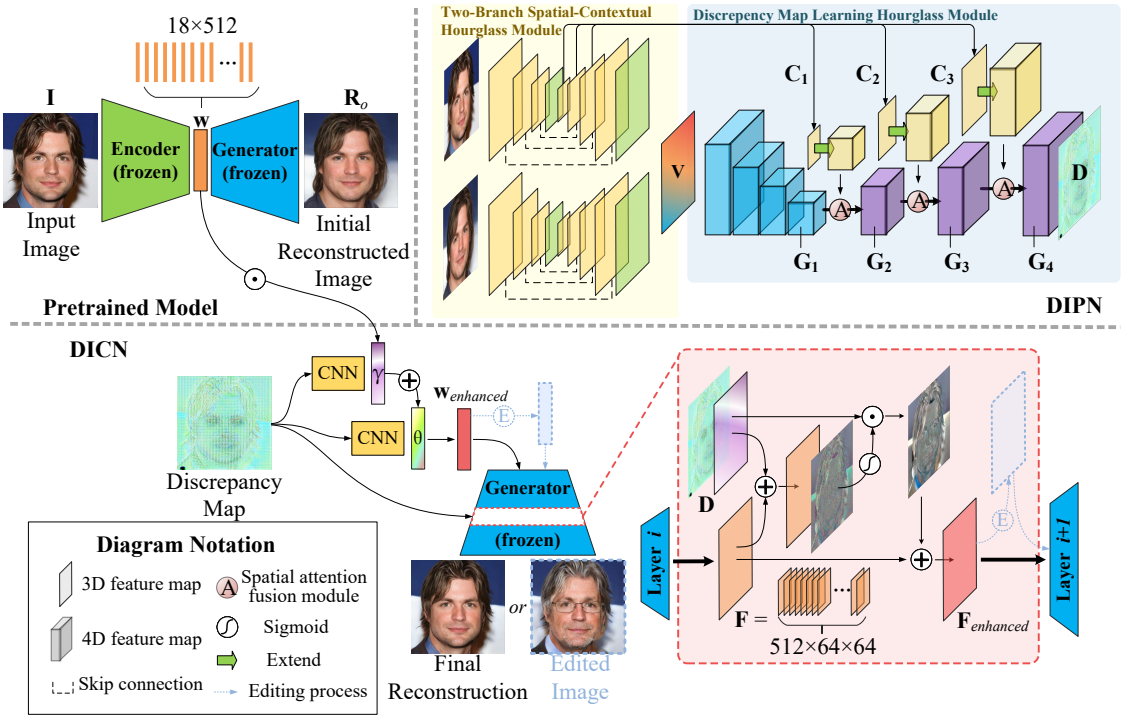

Figure 1: The architecture of SDIC, which consists of DIPN and DICN. DIPN contains a two-branch spatial-contextual hourglass module and a discrepancy map learning hourglass module. First, the original image $\mathbf{I}$ and the initial reconstructed image $\mathbf{R}_o$ (obtained by a pre-trained e4e model) are fed into DIPN to predict the discrepancy map. Then, the discrepancy map is fed into DICN for feature compensation in both the latent code and the GAN generator.

To deal with this, some recent methods (such as HFGI) follow an "edit-and-compensate" paradigm, which computes the distortion map (between the original image and the initial edited image), and then encodes the distortion map to obtain the latent map. In this way, the latent map can be combined with the latent code to compensate for the loss of image details. However, the images generated by these methods easily suffer from artifacts.

The problem of artifacts can be ascribed to the fact that the latent map only encodes the pixel-level spatial information (note that the distortion map is computed by subtracting the initial edited image from the original image). As a result, the latent map ignores high-level contextual information (i.e., the relationship between individual pixels and their surrounding pixels) and involves some attribute-specific disturbance caused by the adoption of the initial edited image.

Design. To address the above problems, we exploit both the spatial and contextual information of the original image. This enables the network to learn the contextual relationship between pixels and spatial details, significantly reducing the artifacts. To this end, we propose a novel SDIC method. Instead of using the "edit-and-compensate" paradigm in previous methods, SDIC adopts a "compensate-and-edit" paradigm, which enables high-quality and flexible editing of attributes. Notably, SDIC effectively exploits the spatial-contextual guided discrepancy information between the original image and the initial reconstructed image and leverages this information to compensate both the latent code and the GAN generator. In this way, our method achieves a good distortion-editability trade-off.

The network architecture of SDIC is given in Fig. 1. SDIC consists of a discrepancy information prediction network (DIPN) and a discrepancy information compensation network (DICN). Given the original image and the initial reconstructed image, DIPN (which contains a two-branch spatial-contextual hourglass module and a discrepancy map learning hourglass module) predicts a discrepancy map. In particular, we incorporate the spatial and contextual information of the original image into the different layers of the discrepancy map learning hourglass module. As a result, the discrepancy map not only encodes fine-grained image details but also models the contextual relationship. The discrepancy map is subsequently fed into DICN to perform feature compensation for the information loss in both the latent code and the GAN generator, obtaining the enhanced latent code and the enhanced latent map.

The attribute editing takes a similar process as the inversion process except that the enhanced latent code and enhanced latent map are modified by attribute editing operations in DICN.

Discrepancy Information Prediction Network

Two-Branch Spatial-Contextual Hourglass Module. The two-branch spatial-contextual hourglass module takes

both the original image $\mathbf{I} \in \mathbb{R}^{3\times H\times W}$ and the initial reconstructed image $\mathbf{R}_o \in \mathbb{R}^{3\times H\times W}$ generated from the pre-trained model (we use the popular e4e model (Tov et al. 2021)) as inputs and extracts their spatial and contextual information, where H and W denote the height and width of the image, respectively. Generally, this module consists of two parallel hourglass branches, where two branches have the same structures. Each branch consists of a convolutional block and a U-Net style upsampling block (Ronneberger, Fischer, and Brox 2015).

Specifically, given an input image ($\mathbf{I}$ or $\mathbf{R}_o$), it is first fed into a convolutional block (consisting of five 3×3 convolutional layers). In the convolutional block, the second, fourth, and fifth layers perform convolution with stride 2, reducing the feature map size to 1/2, 1/4, and 1/8 of the original size, respectively. Such a way gradually expands the receptive fields and captures coarse-to-fine information at different scales.

Next, the reduced feature map is fed into a U-Net style upsampling block with skip-connection at different scales. Technically, the upsampling block contains three 4×4 convolutional layers with stride 1. Each convolutional layer is preceded by an upsampling layer that uses nearest-neighbor interpolation to upsample the feature map. By doing so, we can repeatedly upsample the feature map until its size reaches $3 \times H \times W$. In this way, the sizes of the feature map are gradually resized to 1/4 and 1/2 of the input image size. Meanwhile, a 3×3 convolutional layer is also applied to merge the skip connection and the final upsampled feature. The receptive fields are gradually reduced during upsampling. Thus, more fine-grained information can be obtained. Note that during upsampling, the three feature maps $\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3$, whose sizes correspond to 1/8, 1/4, and 1/2 of the original image size, respectively, serve as the multi-resolution feature maps for the subsequent discrepancy map learning hourglass module. These feature maps model the multi-level spatial-contextual information at different resolutions. Generally, the higher-resolution feature map captures more spatial details while lower-resolution feature map encodes more contextual information.

Finally, the module outputs two feature maps with the size of $C \times H \times W$ (C=48) corresponding to the original image and the initial reconstructed image, respectively.

Discrepancy Map Learning Hourglass Module. The two feature maps from the two-branch spatial-contextual hourglass module are concatenated together to obtain the feature volume $\mathbf{V} \in \mathbb{R}^{2C\times H\times W}$, which is taken as the input of the discrepancy map hourglass learning module. This module consists of a downsampling block and a fusion block. Based on the feature volume $\mathbf{V}$, we apply a downsampling block to obtain the initial discrepancy information. Then, the fusion block combines the initial discrepancy information with the spatial-contextual information of the original image from the two-branch spatial-contextual hourglass module to predict the discrepancy map.

Specifically, the feature volume is first fed into the downsampling block, which uses three downsampling layers to increase the receptive fields. Each downsampling layer consists of a 3×3 ×3 3D convolution with stride 2 and a 3 ×3×3 3D convolution with stride 1. As a result, we get the discrepancy features $\mathbf{G}_1 \in \mathbb{R}^{C\times 48\times H/8\times W/8}$ after downsampling. $\mathbf{G}_1$ is then fed into a fusion layer (consisting of a spatial attention fusion module and an upsampling layer) to obtain the spatial-contextual guided discrepancy feature $\mathbf{G}_2$ at a larger resolution. Each upsampling layer consists of a 4×4×4 3D transposed convolution with stride 2 and two 3×3×3 3D convolutions with stride 1. Similar operations are repeated to obtain higher resolution features $\mathbf{G}_3$ and $\mathbf{G}_4$.

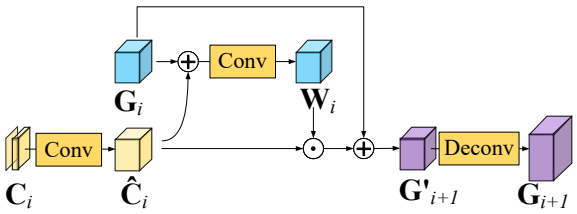

Figure 2: The architecture of the spatial attention fusion module.

Instead of adding or concatenating the discrepancy feature and the spatial-contextual feature, we incorporate a spatial attention fusion module (Woo et al. 2018). This module enables us to adaptively select important regions of features for fusion. The architecture of the spatial attention fusion module is given in Fig. 2.

Technically, $\mathbf{C}_i$ (i=1, 2, 3) is first upsampled by a 3D convolutional layer to obtain $\widehat{\mathbf{C}}_i$ which has the same dimensions as $\mathbf{G}_i$. Then, we add $\widehat{\mathbf{C}}_i$ and $\mathbf{G}_i$ to obtain the enhanced feature, which is fed into a 5×5×5 3D convolutional layer to get spatial attention weights $\mathbf{W}_i$, i.e.,

$$\mathbf{W}_i = \sigma(f^{5\times5\times5}(\mathbf{G}_i + \widehat{\mathbf{C}}_i)), \tag{1}$$

where σ denotes the Sigmoid function and $f^{5\times5\times5}$ denotes the 5×5×5 3D convolution operation.

The attention weights reflect the importance of each spatial location that needs to be emphasized. Hence, we fuse the discrepancy feature and the spatial-contextual feature as

$$\mathbf{G}^{'}_{i+1} = \mathbf{W}_i \odot \widehat{\mathbf{C}}_i + \mathbf{G}_i, \tag{2}$$

where '$\odot$' denotes the Hadamard product.

Next, $\mathbf{G}^{'}_{i+1}$ is fed into a 5×5×5 3D deconvolution layer to obtain $\mathbf{G}_{i+1}$. Finally, we get $\mathbf{G}_4 \in \mathbb{R}^{1\times3\times H\times W}$ and then perform a 3×3 convolutional operation on $\mathbf{G}_4$ to generate the final discrepancy map $\mathbf{D} \in \mathbb{R}^{3\times H\times W}$.

Discrepancy Information Compensation Network

As we previously mentioned, due to the information loss during inversion, the information involved in the latent code $\mathbf{w}$ (with the size of 18×512) is inadequate. Therefore, many existing methods extract additional information to compensate $\mathbf{w}$. However, such a way cannot guarantee the preservation of image details because of the low dimensionality of $\mathbf{w}$. In this paper, we introduce to compensate for the information loss in both $\mathbf{w}$ and the early layer of the GAN generator. The architecture of DICN is illustrated in Fig. 1.

On the one hand, to compensate the latent code $\mathbf{w}$ with the discrepancy map $\mathbf{D}$, we leverage the conventional linear affine transformation. As shown in Fig. 1, we apply two 3×3 convolutional layers to the discrepancy map obtained from DIPN, predicting the scaling parameter $\gamma \in \mathbb{R}^{18\times512}$ and the displacement parameter $\theta \in \mathbb{R}^{18\times512}$, i.e.,

$$\gamma = f^g(\mathbf{D}), \theta = f^t(\mathbf{D}), \tag{3}$$

where f^g and f^t denote the convolution layers.

Based on the above, we apply a channel scaling operation to $\mathbf{w}$ using the scaling parameter γ, followed by a channel displacement operation using the displacement parameter θ. This process effectively filters out uninformative features while compensating for insufficient detailed features. The affine transformation expands the representation space of the generator and facilitates the extraction of high-fidelity features in StyleGAN. The above process is expressed as

$$\mathbf{w}_{enhanced}(i) = \gamma_i \odot \mathbf{w}_i + \theta_i, \tag{4}$$

where $\mathbf{w}_{enhanced}(i)$ is the i-th row of the enhanced latent code $\mathbf{w}_{enhanced}$; $\mathbf{w}_i$ is the i-th row of $\mathbf{w}$; γ_i and θ_i are the i-th rows of γ and θ, respectively.

On the other hand, to compensate for the information loss in the generator, instead of using the affine transformation, we take a similar fusion way as done in DIPN. We apply the discrepancy map to compensate the output of the early layer of the generator (we choose layer 7 in this paper as suggested by HFGI). We denote the output of this layer as the latent map $\mathbf{F}$ (with the size of $512 \times 64 \times 64$). We adopt the spatial attention fusion module to adaptively select the important parts and suppress the unimportant parts of features. Then, we add the attention map to $\mathbf{F}$, that is,

$$\mathbf{F}_{enhanced} = \sigma(f^{c1}(\mathbf{F} + f^{c2}(\mathbf{D}))) \odot f^{c2}(\mathbf{D}) + \mathbf{F}, \tag{5}$$

where f^{c1} and f^{c2} denote convolutional blocks (f^{c1} contains two 3×3 convolution layers and f^{c2} contains four 3×3 convolution layers). $\mathbf{F}_{enhanced}$ denotes the enhanced latent map which is further fed to the next layer of the generator.

Note that some methods (such as FS (Yao et al. 2022), CLCAE, and HFGI) also operate on both the latent code and the generator. However, the differences between these methods and our method are significant. FS and CLCAE follow the "compensate-and-edit" paradigm, where they train an additional encoder to generate a new latent code while obtaining a new latent map to replace one layer of the generator. Although such a way improves the fidelity, the editability is affected (since the new latent space is greatly different from the $\mathcal{W}$ space that is proven to have excellent editability). HFGI follows the "edit-and-compensate" paradigm and leverages the pixel-level distortion map (from the original image and the initial edited image) to compensate the latent map by a linear transformation, limiting the editing performance. Moreover, the above methods ignore spatial-contextual information, resulting in losing some image details or introducing artifacts. In contrast, we generate a spatial-contextual guided discrepancy map (from the original image and the initial reconstructed image) to compensate the latent map by a nonlinear transformation, adaptively fusing features for better compensation and expanding the representation space.

Attribute Editing Operations

We perform operations on both the enhanced latent code and the enhanced latent map for attribute editing. To be specific, the enhanced latent code $\mathbf{w}_{enhanced}$ is first modified by the mainstream latent space editing method (Härkönen et al. 2020; Shen et al. 2020) to obtain the edited latent code (used as the input of the GAN generator). Then, the enhanced latent map is modified as follows. We first obtain the initial reconstructed image $\mathbf{R}_o$ and the initial edited image $\mathbf{E}_o$. Then the latent map $\mathbf{F}^R$ of the reconstructed image and the latent map $\mathbf{F}^E$ of the edited image at layer 7 of the generator are extracted. Assume that the enhanced latent maps of $\mathbf{F}^R$ and $\mathbf{F}^E$ are represented as $\mathbf{F}^E_{enhanced}$ and $\mathbf{F}^R_{enhanced}$, respectively. During the attribute editing, we expect that the difference between $\mathbf{F}^R_{enhanced}$ and $\mathbf{F}^E_{enhanced}$ should be close to that between $\mathbf{F}^R$ and $\mathbf{F}^E$ in the latent space to ensure editability. Hence, instead of generating $\mathbf{F}^E_{enhanced}$ via Eq. (5), we add the difference between $\mathbf{F}^R$ and $\mathbf{F}^E$ to $\mathbf{F}^R_{enhanced}$ for predicting $\mathbf{F}^E_{enhanced}$, that is,

$$\mathbf{F}^E_{enhanced} = \mathbf{F}^R_{enhanced} + \mathbf{F}^E - \mathbf{F}^R, \tag{6}$$

where $\mathbf{F}^R_{enhanced}$ is obtained via Eq. (5).

Joint Loss

During the training stage, the parameters of the generator are frozen so that we can focus on optimizing the encoding process. We design a joint loss to achieve high reconstruction quality and good editability.

For the reconstruction quality, we define the reconstruction loss for the original image $\mathbf{I}$ and the reconstructed image $\mathbf{R}_f$ as

$$\begin{aligned}\mathcal{L}_{rec} = \mathcal{L}_2(\mathbf{I}, \mathbf{R}_f) + \lambda_{LPIPS}\mathcal{L}_{LPIPS}(\mathbf{I}, \mathbf{R}_f) \\ +\lambda_{ID}\mathcal{L}_{ID}(\mathbf{I}, \mathbf{R}_f),\end{aligned} \tag{7}$$

where $\mathcal{L}_2(\mathbf{I}, \mathbf{R}_f)$ denotes the Euclidean distance between $\mathbf{I}$ and $\mathbf{R}_f$ to evaluate the structural similarity; $\mathcal{L}_{LPIPS}(\mathbf{I}, \mathbf{R}_f)$ denotes the LPIPS loss (Zhang et al. 2018) to evaluate the perceptual similarity; $\mathcal{L}_{ID} = 1 - < \mathrm{F}(\mathbf{I}), \mathrm{F}(\mathbf{R}_f) >$ explicitly encourages the encoder to minimize the cosine similarity between $\mathbf{I}$ and $\mathbf{R}_f$, which can measure the identity consistency. Here, $\mathrm{F}(\cdot)$ represents the feature extractor (we use the pre-trained ArcFace model (Deng et al. 2019) for the face domain and the pre-trained ResNet-50 model (Tov et al. 2021) for other domains). λ_{LPIPS} and λ_{ID} indicate the balancing weights.

To ensure the editability, we also incorporate the editing loss, which is defined as

$$\mathcal{L}_{edit} = \mathcal{L}_1(\mathbf{w}, \mathbf{w}_{enhanced}) + \mathcal{L}_1(\mathbf{F}, \mathbf{F}_{enhanced}), \tag{8}$$

where $\mathcal{L}_1(\cdot, \cdot)$ denotes the L_1 norm. This loss is leveraged to constrain the distances between $\mathbf{w}$ and $\mathbf{w}_{enhanced}$ as well as those between $\mathbf{F}$ and $\mathbf{F}_{enhanced}$. In this way, we can keep the latent codes close to the $\mathcal{W}$ space, beneficial to maintain the editability, as suggested in (Tov et al. 2021).

Finally, the joint loss is expressed as

$$\mathcal{L}_{joint} = \mathcal{L}_{rec} + \lambda_{edit}\mathcal{L}_{edit}, \tag{9}$$

where λ_{edit} indicates the balancing weight.

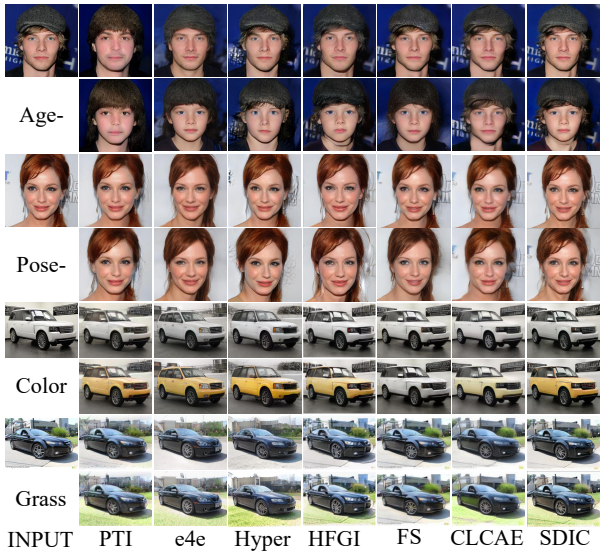

Figure 3: Qualitative comparison between SDIC and several state-of-the-art methods on image inversion and editing tasks. More results are shown in *Supplement B*.

Experiments

Experimental Settings

Datesets. We evaluate our method on two domains: human faces and cars. For the face domain, we adopt the widely-used FFHQ dataset (Karras, Laine, and Aila 2019) for training and the CelebA-HQ dataset (Karras et al. 2017; Liu et al. 2015) for testing. For the car domain, we used the Stanford car dataset (Krause et al. 2013) for training and testing.

Comparison Methods. We compare our SDIC method with various GAN inversion methods, including optimization-based method PTI (Roich et al. 2022) and encoder-based methods (e4e (Tov et al. 2021), HyperStyle (Alaluf et al. 2022), HFGI (Wang et al. 2022), FS (Yao et al. 2022), and CLCAE (Liu, Song, and Chen 2023)). All the results of these comparison methods are obtained by using the trained models officially provided by the corresponding authors.

Implementation Details. In this paper, we adopt InterfaceGAN (Shen et al. 2020) for face image editing and GANSpace (Härkönen et al. 2020) for car image editing. We use the pre-trained StyleGAN generator and the e4e encoder in our method. The sizes of the input and output of the network are both 1024×1024. λ_{LPIPS}, λ_{ID}, and λ_{edit} are empirically set to 0.8, 0.2, and 0.5, respectively. We use the ranger optimizer (Yong et al. 2020) with a learning rate of 0.001 and a batch size of 2. Our model is trained 100,000 steps on the NVIDIA GeForce RTX 3080 GPU.

Reconstruction Results

Quantitative Evaluation. We quantitatively compare our method with state-of-the-art GAN inversion methods. The results are shown in Table 1. We use the $SSIM$, $PSNR$, and L_2 to measure the reconstruction error, and the $LPIPS$ (Zhang et al. 2018) for the perceptual quality. We also use CurricularFace (Huang et al. 2020) to extract the features of two images and calculate their cosine similarity as the ID distance, which can measure the identity similarity between each reconstruction image and the original input image. These metrics are evaluated on the first 1,000 images of CelebA-HQ. In addition, we also report the inference time obtained by these methods.

As we can see, our method significantly outperforms both the encoder-based methods (HFGI, FS, CLCAE, and HyperStyle) and the optimization-based method (PTI) in terms of reconstruction quality (including ID, $SSIM$, $PSNR$, $LPIPS$, and L_2). Notably, our method achieves a much faster inference speed than the optimization-based method. In a word, SDIC obtains the highest fidelity at a fast inference speed among all the competing methods.

Qualitative Evaluation. Fig. 3 gives the qualitative comparison between SDIC and several state-of-the-art methods. For the face domain, SDIC effectively preserves background and foreground details. In the first row of Fig. 3, only SDIC preserves the indentation on the cheeks. In the third row of Fig. 3, SDIC and PTI successfully reconstruct the earrings and bangs. For the car domain, SDIC shows superior preservation of image details such as car lights, front, and reflective parts compared with other encoder-based methods. The above results show the effectiveness of SDIC.

Editing Results

Quantitative Evaluation. There are no intuitive measures to evaluate the editing performance. Therefore, we calculate the ID distance (Huang et al. 2020) between the original image and the manipulation one. Meanwhile, we conduct a user study to evaluate the editing results as done in HFGI and CLCAE. Specifically, we collect 56 edited images of faces and cars for all the competing methods and ask 30 participants to choose the images with high fidelity and appropriate manipulation. The results are given in Table 1. Our proposed method achieves the same ID as PTI and greatly outperforms the other competing methods in terms of $User\ Study$.

Qualitative Evaluation. The image editing results are given in Fig. 3. Compared with PTI, e4e, and Hyper, SDIC retains more detailed information in the editing results (e.g., the hat in the second row of Fig. 3, the shadow and gap on the ground in the eighth row of Fig. 3) while maintaining high image quality. In contrast, HFGI exhibits numerous artifacts (e.g., the neck in the second row and the right-side hair in the fourth row of Fig. 3). FS loses facial details and cannot edit car images well. CLCAE shows poor editing results (e.g., distortion in the neck in the fourth row of Fig. 3) and small attributes changes (e.g., minimal age reduction in the second row and almost no color change in the sixth row of Fig. 3). In general, our method effectively balances fidelity and editability by incorporating spatial-contextual information.

Ablation Studies

Influence of Spatial-Contextual Information. We compare the reconstruction results obtained by our method with and without the two-branch spatial-contextual hourglass module. The results are given in Fig. 4(a). For our method without the two-branch spatial-contextual hourglass module,

	Inversion						Editing	
Method	$ID \uparrow$	$SSIM \uparrow$	$PSNR \uparrow$	$LPIPS \downarrow$	$L_2\ (MSE) \downarrow$	$Time\ (s) \downarrow$	$ID \uparrow$	$User\ Study \uparrow$
PTI	0.832	0.703	24.355	0.110	0.012	201.357	**0.726**	22.500%
e4e	0.495	0.537	19.390	0.206	0.050	**0.033**	0.452	18.750%
HyperStyle	0.737	0.624	22.513	0.104	0.025	0.710	0.663	21.668%
HFGI	0.606	0.641	22.372	0.136	0.026	0.108	0.610	22.918%
FS	0.815	0.648	23.797	0.934	0.015	0.568	0.492	17.083%
CLCAE	0.708	0.725	25.665	0.083	0.012	0.125	0.518	22.918%
SDIC	**0.871**	**0.815**	**27.672**	**0.057**	**0.007**	0.321	**0.726**	**65.832%**

Table 1: Quantitative comparison for inversion/editing quality on the face domain. The value of *User Study* is the percentage of users that choose this method. The best results are in bold.

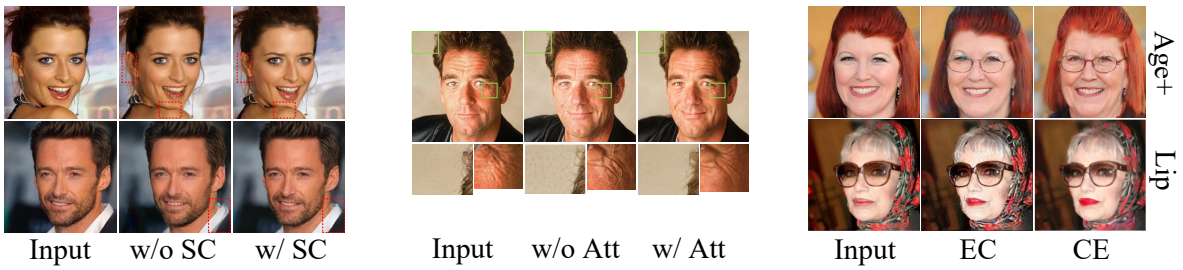

Figure 4: Ablation study results on the face domain. In the figure, "w/" and "w/o" denotes "with" and "without", respectively. "SC" denotes "Spatial-Contextual". "EC" and "CE" denotes the "Edit-and-Compensate" paradigm and the "Compensate-and-Edit" paradigm, respectively.

we concatenate the two input images and directly feed them into the discrepancy map learning hourglass module. We select the two images under different conditions (i.e., side face and deformed mouth). Without the two-branch spatial-contextual hourglass module, our method shows substantial artifacts on the edge of the portrait and low fidelity for the mouth. In contrast, with the module, our method is more robust under different conditions, effectively removing artifacts and preserving more spatial details.

Influence of Spatial Attention Fusion in DICN. To compensate the latent map with the discrepancy information, we leverage the spatial attention fusion module. We evaluate the performance of our method with the spatial attention fusion module and with the conventional affine transformation. The results are given in Fig. 4(b).

Compared with our method with the spatial attention fusion module, our method with conventional affine transformation tends to give worse results. Some details of the generated image are overemphasized. For example, the wrinkles at the corners of the eyes are not natural while many reticulated lines appear in the background. This is because the affine transformation is only a linear transformation, leading to limited compensation results. In contrast, our model with the spatial attention fusion module adaptively selects informative parts and suppresses uninformative parts, generating high-quality images with a natural and smooth appearance.

Influence of "Compensate-and-Edit" Paradigm. We compare the "compensate-and-edit" and the "edit-and-compensate" paradigms. Specifically, we design a variant of our method based on the "edit-and-compensate" paradigm. That is, this variant predicts the discrepancy information between the initial edited and original images with DIPN and reconstructs the image with DICN. A comparison between this variant and our method is shown in Fig. 4(c).

The variant gives worse editing results than our method. It considers the attribute changes as low-fidelity regions that do not match the original image. Thus, the network tends to correct them. Such a way reduces the effectiveness of the editability. In contrast, our method retains more detail after editing by the paradigm of "first compensating and then editing". The above experiments show the effectiveness of the "compensate-and-edit" paradigm.

More ablation studies and applications of our method can refer to *Supplement A and C*.

Conclusion

In this paper, we introduce a novel SDIC method, consisting of DIPN and DICN. Following the "compensate-and-edit" paradigm, SDIC first generates the spatial-contextual guided discrepancy information between the original image and the initial reconstructed image by DIPN and then compensates both the latent code and the GAN generator with the discrepancy information by DICN. Experimental results show that our method can strike a good distortion-editability trade-off at a fast inference speed, which shows the effectiveness and efficiency of our method.

One limitation of our proposed method is the difficulty in handling large manipulation cases (see *Supplement D* for failure cases). We intend to explicitly align the edited image and the original image in future work.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants 62372388, 62071404, and U21A20514, and by the Open Research Projects of Zhejiang Lab under Grant 2021KG0AB02.

References

Abdal, R.; Qin, Y.; and Wonka, P. 2019. Image2StyleGAN: How to embed images into the StyleGAN latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4432–4441.

Abdal, R.; Qin, Y.; and Wonka, P. 2020. Image2StyleGAN++: How to edit the embedded images? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8296–8305.

Abdal, R.; Zhu, P.; Mitra, N. J.; and Wonka, P. 2021. Styleflow: Attribute-conditioned exploration of StyleGAN-generated images using conditional continuous normalizing flows. *ACM Transactions on Graphics (ToG)*, 40(3): 1–21.

Alaluf, Y.; Patashnik, O.; and Cohen-Or, D. 2021. ReStyle: A residual-based StyleGAN encoder via iterative refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 6711–6720.

Alaluf, Y.; Tov, O.; Mokady, R.; Gal, R.; and Bermano, A. 2022. HyperStyle: StyleGAN inversion with hypernetworks for real image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 18511–18521.

Bau, D.; Strobelt, H.; Peebles, W.; Wulff, J.; Zhou, B.; Zhu, J.-Y.; and Torralba, A. 2020. Semantic photo manipulation with a generative image prior. *arXiv preprint arXiv:2005.07727*.

Bau, D.; Zhu, J.-Y.; Wulff, J.; Peebles, W.; Strobelt, H.; Zhou, B.; and Torralba, A. 2019. Seeing what a GAN cannot generate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4502–4511.

Chang, J.-R.; and Chen, Y.-S. 2018. Pyramid stereo matching network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5410–5418.

Choi, M. J.; Lim, J. J.; Torralba, A.; and Willsky, A. S. 2010. Exploiting hierarchical context on a large database of object categories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 129–136.

Collins, E.; Bala, R.; Price, B.; and Susstrunk, S. 2020. Editing in style: Uncovering the local semantics of GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5771–5780.

Deng, J.; Guo, J.; Xue, N.; and Zafeiriou, S. 2019. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4690–4699.

Dinh, T. M.; Tran, A. T.; Nguyen, R.; and Hua, B.-S. 2022. Hyperinverter: Improving StyleGAN inversion via hypernetwork. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11389–11398.

Goetschalckx, L.; Andonian, A.; Oliva, A.; and Isola, P. 2019. Ganalyze: Toward visual definitions of cognitive image properties. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 5744–5753.

Gu, J.; Shen, Y.; and Zhou, B. 2020. Image processing using multi-code GAN prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3012–3021.

Härkönen, E.; Hertzmann, A.; Lehtinen, J.; and Paris, S. 2020. GANspace: Discovering interpretable GAN controls. *Conference on Neural Information Processing Systems (NeurIPS)*, 33: 9841–9850.

Hu, X.; Huang, Q.; Shi, Z.; Li, S.; Gao, C.; Sun, L.; and Li, Q. 2022. Style transformer for image inversion and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11337–11346.

Huang, Y.; Wang, Y.; Tai, Y.; Liu, X.; Shen, P.; Li, S.; Li, J.; and Huang, F. 2020. Curricularface: adaptive curriculum learning loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5901–5910.

Hutchinson, B.; Denton, E.; Mitchell, M.; and Gebru, T. 2019. Detecting bias with generative counterfactual face attribute augmentation. *arXiv preprint arXiv:1906.06439*.

Kang, K.; Kim, S.; and Cho, S. 2021. GAN inversion for out-of-range images with geometric transformations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 13941–13949.

Karras, T.; Aila, T.; Laine, S.; and Lehtinen, J. 2017. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.

Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4401–4410.

Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and improving the image quality of StyleGAN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8110–8119.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the International IEEE Workshop on 3D Representation and Recognition*, 554–561.

Li, J.; Wei, Y.; Liang, X.; Dong, J.; Xu, T.; Feng, J.; and Yan, S. 2016. Attentive contexts for object detection. *IEEE Transactions on Multimedia (TMM)*, 19(5): 944–954.

Liu, H.; Song, Y.; and Chen, Q. 2023. Delving StyleGAN inversion for image editing: A foundation latent space viewpoint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10072–10082.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *Proceedings of the*

IEEE/CVF International Conference on Computer Vision (ICCV), 3730–3738.

Pidhorskyi, S.; Adjeroh, D. A.; and Doretto, G. 2020. Adversarial latent autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14104–14113.

Richardson, E.; Alaluf, Y.; Patashnik, O.; Nitzan, Y.; Azar, Y.; Shapiro, S.; and Cohen-Or, D. 2021. Encoding in style: a StyleGAN encoder for image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2287–2296.

Roich, D.; Mokady, R.; Bermano, A. H.; and Cohen-Or, D. 2022. Pivotal tuning for latent-based editing of real images. *ACM Transactions on graphics (TOG)*, 42(1): 1–13.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 234–241.

Shen, Y.; Gu, J.; Tang, X.; and Zhou, B. 2020. Interpreting the latent space of GANs for semantic face editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9243–9252.

Shen, Y.; and Zhou, B. 2021. Closed-form factorization of latent semantics in GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1532–1540.

Tov, O.; Alaluf, Y.; Nitzan, Y.; Patashnik, O.; and Cohen-Or, D. 2021. Designing an encoder for StyleGAN image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4): 1–14.

Voynov, A.; and Babenko, A. 2020. Unsupervised discovery of interpretable directions in the GAN latent space. In *arXiv preprint arXiv:2002.03754*.

Wang, B.; and Ponce, C. R. 2021. The geometry of deep generative image models and its applications. *arXiv preprint arXiv:2101.06006*.

Wang, T.; Zhang, Y.; Fan, Y.; Wang, J.; and Chen, Q. 2022. High-fidelity GAN inversion for image attribute editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11379–11388.

Woo, S.; Park, J.; Lee, J.-Y.; and Kweon, I. S. 2018. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, 3–19.

Xia, W.; Zhang, Y.; Yang, Y.; Xue, J.-H.; Zhou, B.; and Yang, M.-H. 2022. GAN inversion: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yao, X.; Newson, A.; Gousseau, Y.; and Hellier, P. 2022. A Style-Based GAN Encoder for High Fidelity Reconstruction of Images and Videos. *European Conference on Computer Vision (ECCV)*.

Yong, H.; Huang, J.; Hua, X.; and Zhang, L. 2020. Gradient centralization: A new optimization technique for deep neural networks. In *European Conference on Computer Vision (ECCV)*, 635–652. Springer.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 586–595.

Zhu, J.; Shen, Y.; Zhao, D.; and Zhou, B. 2020a. In-domain GAN inversion for real image editing. In *European Conference on Computer Vision (ECCV)*, 592–608.

Zhu, P.; Abdal, R.; Qin, Y.; Femiani, J.; and Wonka, P. 2020b. Improved StyleGAN embedding: Where are the good latents? *arXiv preprint arXiv:2012.09036*.