Any Fashion Attribute Editing: Dataset and Pretrained Models

Shumin Zhu*, Xingxing Zou*, Wenhan Yang, and Wai Keung Wong✉

Abstract—Fashion attribute editing is essential for combining the pertise of fashion designers with the potential of generative artif intelligence. In this work, we focus on 'any' fashion attribute edit 1) the ability to edit 78 fine-grained design attributes commonly served in daily life; 2) the capability to modify desired attributes w keeping the rest components still; and 3) the flexibility to continuo edit on the edited image. To this end, we present the Any Fash Attribute Editing (AFED) dataset, which includes 830K high-quality fa ion images from sketch and product domains, filling the gap fo large-scale, openly accessible fine-grained dataset. We also prop Twin-Net, a twin encoder-decoder GAN inversion method that ot diverse and precise information for high-fidelity image reconstruct This inversion model, trained on the new dataset, serves as a bust foundation for attribute editing. Additionally, we introduce F sPCA to identify semantic directions in latent space, enabling ac rate editing without manual supervision. Comprehensive experime including comparisons with ten state-of-the-art image inversion m ods and four editing algorithms, demonstrate the effectiveness of Twin-Net and editing algorithm. All data and models are availabl https://github.com/ArtmeScienceLab/AnyFashionAttributeEditing .

Index Terms—Fashion Attribute Editing Dataset, Encoder-based GAN Inversion, Attribute Editing in Latent Space

1 Introduction

Generative AI is projected to boost profits in the fashion sector significantly [1]. Its success hinges on the performance of models in generation and editing tasks [2], [3], [4]. Although there is a trend favouring Diffusion Models for their ability to produce high-quality, diverse images [5], [6], [7], [8]. However, they need multiple inference steps to generate a single sample, which makes them less efficient than GAN-based methods [9]. Meanwhile, applying diffusion models to editing tasks also presents challenges, such as finding accurate features for specific semantics in a larger latent space [10]. *Therefore, this study focuses on exploring solutions within the GAN-based framework.*

(Shumin Zhu, Xingxing Zou* contributed equally to this work.) (Corresponding author: Wai Keung Wong✉.)*

Xingxing Zou is with the School of Fashion and Textiles, The Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: xingxing.zou@polyu.edu.hk)

Wenhan Yang is with PengCheng Laboratory, Shenzhen, Guangdong, P.R. China. (e-mail: yangwh@pcl.ac.cn)

Shumin Zhu, Wai Keung Wong is with the School of Fashion and Textiles, The Hong Kong Polytechnic University, and also with the Laboratory for Artificial Intelligence in Design, Hong Kong (e-mail: shumin.zhu@connect.polyu.hk; calvin.wong@polyu.edu.hk)

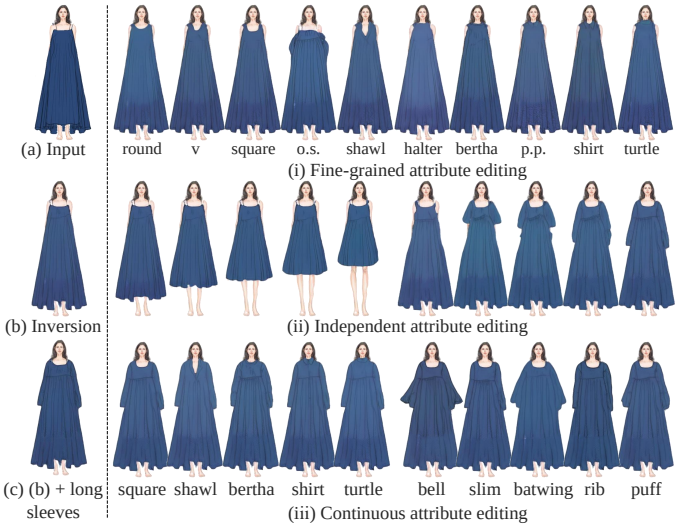


Fig. 1: Any Fashion Attributes Editing: Conditioned on the inverted result (b) of input (a), we target: (i) Fine-grained attribute editing; (ii) Independent attribute editing: edited results while preserving the rest without deformation; (iii) Continuous attribute editing: editing on the edited result.

Substantial progress has been made through various datasets [11], [12]. The Fashion-Gen dataset [13] introduced 293,008 images and item descriptions for high-resolution generation. Fu et al. [14] developed the SHHQ dataset with 230,000 images to evaluate editing algorithms like InterFaceGAN [15], StyleSpace [16], and SeFa [17]. Drawing from these datasets, various fashion image editing methodologies have emerged: SemanticStyleGAN [18] enables controlled generation of diverse styles, while DragGAN [19] and FreeDrag [20] offer interactive editing. However, many challenges still remain in fashion attribute editing.

As shown in Fig. 1, fashion includes numerous fine-grained attributes, with at least 50 identified in previous research [21], [22], [23]. The goal of the editing task is to accurately disentangle the semantic directions of these attributes in latent space. **Accurate image inversion is foundational; if the inversion model cannot reconstruct a specific attribute, it cannot be edited in subsequent steps.** Thus, the increasing number of attributes heightens the complexity of the editing process and raises the requirements for the inversion model. Current works [24], [25], [26] successfully reconstruct 'common' attributes, such as 'long sleeves', but struggle with 'uncommon' ones, like 'peter

pan collar'. In addition to accurate editing, fashion design demands independent and continuous editing. Independent editing modifies a specific attribute while keeping the rest of the image unchanged; for example, in Fig. 1 (ii), the free length is altered without affecting other parts. Continuous editing refers to re-editing another attribute on an already edited image; for instance, in Fig. 1 (iii), (c) is (b) edited with long sleeves, followed by changing the neckline to a 'Square Neckline', 'Shawl', or 'Bertha' etc, in (c). These requirements necessitate a more sophisticated disentanglement of attributes. Lastly, although product images are key in the design process, sketch images are edited more frequently in the early stages of fashion design.

To this end, we first test existing open-source encoder-based GAN inversion models, but they fail to accurately reconstruct the fine-grained fashion attributes. Meanwhile, upon examining fashion training datasets, we found a lack of open source availability [14], [13]. In other words, the number of released images is insufficient to train a satisfactory GAN inversion model [27], [28]. Additionally, existing datasets primarily consist of product images from online stores, with a scarcity of sketches (examples in Fig. 1), which are standard in fashion design. *To this end, the basis for the next step is constructing a new dataset that includes product and sketch fashion images with fine-grained attributes first.*

After obtaining the new dataset, we further examine the encoder-based GAN inversion methods [24], [25], [29], [30], [31], [32], [33], [34], [35]. Although these methods, such as HFGI [31] and StyleRes [30], aim to reconstruct high-fidelity images, they face challenges in handling varied semantic attributes. This is mainly because current encoder-based image inversion methods struggle to reconstruct fine-grained attributes due to semantic ambiguity. *An inversion model with higher fidelity and editability is needed to integrate the semantics of fine-grained fashion attributes.*

Finally, we explore both supervised and unsupervised attribute editing techniques [36], [37], [38], [39], [40], [41], [16]. Supervised methods [37], [42], [43], [44], [45], such as InterFaceGAN [15], rely on classifiers that are limited in scope. Unsupervised methods, including GANSpace [38], SeFA [17], and ReSeFA [46], reduce the reliance on classifiers, but may require manual oversight. Interactive methods, such as DragGAN [19], offer direct manipulation but have limitations with specific fashion attributes. *To handle a larger number of attributes, a more precise editing algorithm is needed to disentangle the semantic directions.*

In summary, we thus propose solutions from three aspects for Any Fashion Attribute Editing. First, we introduce the AFED **dataset**, which comprises 830,000 full-body fashion images to address limitations in current datasets. It includes 230,000 human fashion images, along with 300,000 solid-color women's clothing sketches and 300,000 printed men's clothing sketches. Second, we propose a novel encoder-based GAN **inversion method** called Twin-Net, designed for high-fidelity image inversion suitable for fine-grained fashion attribute editing. Unlike previous methods, our approach supplements missing information with more precise semantic attribute directions. Building on strong image inversion performance, we introduce a few-shot **editing algorithm** that uses paired attribute images to estimate disentangled, fine-grained attribute directions in latent space, enabling flexible manipulation at the semantic level. This method successfully disentangles directions for editing 78 fine-grained fashion attributes.

TABLE 1: Listed of popular fashion datasets. Gray color indicates datasets are friendly for fashion attribute editing. The abbreviation 'p.' refers to 'product', and 's.' refers to 'sketch', respectively. 'Full-Body Ratio' refers to the proportion of images in the dataset that depict the model's full body relative to the total number of images.

Dataset	Total Image	Mean Resolution	Ratio of Full-Body Image	Domain
ATR [47]	7,700	400 ×600	76%	p.
Mark1et1501 [48]	32,668	128 ×64	100%	p.
DeepFashion [49]	146,680	1101 ×750	6.8%	p.
LIP [50]	50,462	196 ×45	37%	p
Prada[49], [51]	78,979	256 ×256	-	p.
ZalandoGAN [52]	120,000	128 ×128	0%	p.
VITON [53]	16,253	256 ×192	0%	p.
FashionGEN [13]	293,008	1360 ×1360	-	p.
FashionAI [22]	324K	512 ×512	-	p.
iMaterialist [21]	1M+	600 ×600	1.4%	p.
POG [54]	583k	n ×797	0%	p.
MVP [55]	49,211	256 ×192	72.5%	p.
VITON-HD [12]	27358	1024 ×768	0%	p.
Dress code [11]	107,584	1024 ×768	50%	p.
SHHQ [14]	231,176	1024 ×512	100%	p.
DeepFashion-MM [56]	44,096	1024 ×512	29%	p.
AFED	**830,000**	1024 ×512	**100%**	**p. & s.**

Our main contributions are as follows:

- We present the Any Fashion Editing Dataset (AFED), which contains 830K high-quality full-body fashion images from two domains. This dataset creates a new challenge scene of any fashion editing with complex fashion attributes, making it ideal for arbitrary fashion attribute editing research. It inspires the design of the method in two directions: 1) relax the trade-off between inversion and editing; 2) better fine-grained semantics disentanglement. **All data, source code, models** will be released for academic use.
- We propose Twin-Net, a GAN-based framework for high-quality fashion image inversion and editing. Different from existing methods taking random noise as a condition for editability, our approach leverages factorized semantics as conditions, preserving the person's identity while allowing flexible, unsupervised editing of fashion attributes.
- We design PairsPCA, a method that is capable of mining the clear mapping relationship between latent and semantic meanings consistently. This modeling facilitates the targeted editing of individual attributes without unintended effects on others and also enables attribute editing with the least human effort.
- Comprehensive experiments validate the effectiveness of our proposed methods. A total of 14 baselines were compared, including seven GAN inversion methods, three diffusion-based editing methods, and four GAN-based attribute editing methods. We trained 27 models across three datasets, comprising StyleGAN models and models for eight GAN inversion methods (including Twin-Net). The experiments evaluated reconstruction quality, editing capability, accuracy, and disentanglement. The results demonstrate the effectiveness of our proposed framework and methods.

2 Related Work

2.1 Fashion Attribute Editing Dataset

A comprehensive list of popular fashion datasets with potential for editing tasks is presented in Tab. 1. However, most are not conducive to our objectives: 1) Datasets for attribute recognition [47], [48], [49], [50], [22], [21], [52] face challenges due to complex images—varied lighting, poses,

Fig. 2: Initial inversion results on the SHHQ dataset are presented as follows: (a) is the original image from the SHHQ dataset, (b) is a manually edited image highlighting the desired fashion attribute based on (a), and (c) is the inverted result from the officially released pre-trained encoder. This triad illustrates the pre-trained encoder's ineffectiveness in capturing and generating the specified fashion attributes.

backgrounds, and types (half-body, full-body, single item, etc.). This complexity makes them unsuitable for our editing tasks, despite their potential for robust fine-grained editing, as image alignment significantly impacts performance [46]. 2) Datasets for virtual try-on [53], [55], [52], [12], [11] or fashion compatibility learning [54] often feature single items against clean backgrounds. However, most of them have low resolution [53], [55], [52], which hinders our goal of high-fidelity image inversion. High-resolution virtual try-on datasets, such as VITON-HD, do not include full-body images of models. The DressCode dataset is suitable for our editing task, but it only contains 53,792 available full-body images of models, and all models' faces are missing. 3) High-resolution datasets for fashion image synthesis, such as [13], DeepFashion-MM [56], SHHQ [14], show promise. However, preliminary tests reveal issues with successful inversion on SHHQ's public model (Fig. 2). Moreover, SHHQ's accessible dataset is limited to 40,000 images, and while FashionGEN has consistent full-body images, they constitute only a small fraction of its total size. DeepFashion-MM contains just 12,788 full-body images. Even combined, these datasets fall short of the required volume. *To this end, we introduce the AFED dataset.*

2.2 Encoder-based GAN Inversion.

Encoder-based GAN inversion methods [57], [24], [58], [25], [29], [30], [31], [32], [33], [34], [59], [35] involve learning an encoder that maps an image to latent space and a generator that reconstructs the image from the latent code. This enables real-time inference, enhancing practical applicability. Some methods [29], [32] use iterative forward passes, leveraging training set knowledge during projection to improve editability, albeit at some cost to fidelity. Other studies [24], [25], [57] project latent codes via single or iterative forward passes, which may lead to significant information loss and imprecise reconstructions. To mitigate this, some methods [31], [30] employ two encoders: one for encoding an image to a low-rate latent space and another for reconstructing missing details through high-rate features. While these approaches improve high-fidelity inversion, they often lack clear semantic information, relying instead on randomly generated data with no defined meaning. For example, HFGI [31] uses random augmentation of residuals between the input and inverted image as missing information, while StyleRes [30] uses random latent codes as editing directions. Such vague information is insufficient for effective image inversion with fine-grained attributes. *In contrast, we propose incorporating semantic attribute direction during training.*

2.3 Attribute Editing in Latent Space.

Many algorithms have been proposed to identify latent directions for attribute editing [36], [37], [38], [39], [60], [40], [41], [61], [16], [62]. Supervised methods [37], [42], [43], [44], [45], such as InterFaceGAN [15] employs a linear SVM in the latent space, and [63] learns a mapping in the latent space with the aid of an age classifier. However, these methods are limited by their dependency on attribute classifiers, restricting editing to a small set of attributes. Unsupervised methods have been proposed to address these limitations [38], [19], [64], [65], [66], [16], [67], [46], [17]. For instance, GANSpace [38] performs PCA on latent codes to identify directions of maximum variations, followed by manual filtering of directions. SeFA [17] optimizes latent directions to maximize variations after projection on the affine matrix A. ReSeFa [46] extends SeFA to factorize the latent semantics learned by GANs concerning an arbitrary image region. DragGAN [19] enables users to 'drag' any points of the image to reach target points interactively. *Different from these methods, we perform attribute editing with disentangled semantic direction using minimal supervision.*

3 Any Fashion Attribute Editing Dataset

3.1 Dataset Construction

The AFED dataset consists of three sub-datasets, as illustrated in Fig. 3. It includes 300,000 (a) solid-color women's clothing sketches (AFED-S-Color), 300,000 (b) printed men's clothing sketches (AFED-S-Print), and 230,000 (c) human fashion images (AFED-H-Product). For the AFED-S-Color and AFED-S-Print, the data were generated using a mature pipeline[1] previously utilized to create full-body sketch images with fine-grained fashion attributes. Generating these sketches took about 250 hours on a single GTX 3090 GPU.

Regarding the AFED-H-Product, we initially collected over 1,000,000 raw fashion images from the internet, encompassing various clothing styles and fine-grained attributes. During this process, we focused on gathering images with 'common' and 'uncommon' attributes. During data processing, we considered seven significant factors [14]: attributes balance and diversity, resolution, body position, body part occlusion, human posture, multiple people, and background. These factors are crucial for curating high-quality, fine-grained fashion datasets. Specifically, we constructed the dataset considering the following aspects: 1) Intentional collection of fashion images with 'uncommon' attributes during the data collection. 2) Retention of fashion images with resolutions greater than 1024×512. 3) Ensuring consistency through adjustments such as segmentation, cropping, and alignment of people in the images. 4) Eliminating images missing any body parts, as we focus on the whole body. 5) Removal of uncommon model display poses, such as sitting or severe sideways positions, through

1. https://code-create.com.hk/aida/

(a) AFED-S-Color (b) AFED-S-Print

(c) AFED-H-Product

Fig. 3: Image samples from AFED dataset.

manual inspection to ensure the learnability of the data distribution. 6) Retention of only the unoccluded full-body image of a single person in instances where fashion show images featured multiple individuals. 7) Standardization of representation and elimination of the influence of complex backgrounds by utilizing segmentation masks [68] to modify the image background to pure white. The manual collection process is hindered by its slow pace and high costs. For reference, a team of four carried out this collection manually over a year, averaging 4,800 images per week.

3.2 Dataset Comparison

Statistically, the AFED dataset is 2.8 times larger than FashionGEN [13], and 3.6 times larger than SHHQ [14], significantly surpassing the other datasets compared. Although iMaterialist has a large volume of data, its low resolution makes it unsuitable for fashion attribute editing tasks. This limitation also affects other datasets, such as FashionAI [22], POG [54], and MVP [55], etc. In contrast, AFED, DeepFashion [49], SHHQ [14], DeepFashion-MM [56], and FashionGEN [13] provide high-definition images. Regarding the proportion of full-body images, DeepFashion-MM provides 29% full-body images, while DeepFashion's proportion is 6.8%. Both AFED and SHHQ offer a full-body image ratio of 100%. Notably, AFED is the only dataset that includes sketches and products, making it suitable for editing both image domains. Additionally, we observed a significant imbalance in the distribution of fashion attributes within the existing datasets. Attributes such as length, 'V Neckline', and 'Round Neckline' dominate, comprising over 30% of the dataset. For instance, in the FashionGEN dataset [13], the proportion of sleeve length among all clothing images is 0.7642, while the 'Round Neckline' is 0.3296, and 'Peter Pan' is merely 0.0009, indicating an extremely long-tailed distribution. Using FashionGEN as a reference, we consider attributes with a proportion over 0.1 as 'common' (16 in total). The attribute distribution is shown in Fig. 4.

4 Approach

This section mainly introduces the proposed approach for achieving fashion attribute editing, including subsection 4.1 Inversion Method and subsection 4.2 Editing Algorithm. Specifically, we introduce Twin-Net for image inversion, designed to enhance fidelity and editability by supplementing missing information. Twin-Net, inspired by StyleRes [30], uses two pipelines for generating high-quality edited and inverted images, respectively. Unlike StyleRes, Pipeline 2 of Twin-Net takes the inverted image, i.e., the output of Pipeline 1, as input. This setting is motivated by the finding that the result inverted twice of input will lose more information than when inverted only once and thus, in other words, it could supplement more information during the training process (see visualized results of $\Delta^{'}_{rec}$ and $\Delta_{rec_{gt}}$ in Fig. 5). Meanwhile, it directly calculates differences to supplement missing information, thereby supplementing the missing information more intuitively. Additionally, we incorporate a Distortion Alignment Module, used in HFGI [31], to analyze and correct differences between original and reconstructed images during editing. Unlike HFGI's random transformations, Twin-Net uses two encoder-generator sets for optimizing image quality. After developing the inversion model, we propose PairsPCA, a few-shot editing algorithm that facilitates precise semantic editing. Further details are presented in the following.

4.1 Inversion Method: Twin-Net

Given an input image $\mathbf{X} \in \mathbb{R}^{H\times W\times 3}$, our goal is to produce a high-quality edited image $\tilde{\mathbf{X}}^{h}_{edit} \in \mathbb{R}^{H\times W\times 3}$ and reconstructed image $\tilde{\mathbf{X}}^{h}_{rec} \in \mathbb{R}^{H\times W\times 3}$, respectively. The training framework uses two sub-networks, Pipeline 1 for generating the $\tilde{\mathbf{X}}^{h}_{edit}$ and Pipeline 2 for $\tilde{\mathbf{X}}^{h}_{rec}$, as shown in Fig. 5. Each subnetwork is composed of a pre-trained basic encoder E_0 [25], a basic generator G_0 [69], and a distortion consultation branch. Both E_0 and G_0 remain fixed during the training phase, while the distortion consultation branch is trained to approximate missing information within a high-rate feature space and share weights in Pipeline 1 and Pipeline 2.

Specifically, the encoder E_0 first maps the input image $\mathbf{X}$ into the latent space $\mathcal{W}+$, producing the low-rate latent code $\mathbf{W}^{+} \in \mathbb{R}^{18\times 512}$. Then, the latent code $\mathbf{W}^{+}$ is feded to the generator G_0 to obtain the low-fidelity reconstructed image $\tilde{\mathbf{X}}^{'}_{rec} \in \mathbb{R}^{H\times W\times 3}$. It is worth noting that, compared with $\mathbf{X}$, the low-fidelity $\tilde{\mathbf{X}}^{'}_{rec}$ lost the attribute of 'one-shoulder' as shown in Fig. 5. This finding shows that single-pass inversions in low-rate latent spaces lead to reconstruction inaccuracies due to information loss.

Our goal is to achieve a high-quality edited image within Pipeline 1; accordingly, we introduce an editing direction into the latent code. We aim for controllable changes in the desired direction while keeping other aspects constant. Random directions utilized in previous research can lead to uncontrollable edits and alignment challenges. In contrast, overly predefined directions can limit edit flexibility. Inspired by [46], we factorize the latent semantics of the generator G_0 learned in the clothing region in an unsupervised manner. Specifically, given the generator G_0 and the generated image $\mathbf{A} = G_0(\mathbf{z})$, where $\mathbf{z} \in \mathbb{R}^{512}$ is drawn from the $\mathcal{Z}$ space (the input latent space of StyleGAN as described by Karras et al. [70]), we designate the clothing partition of A as A^f and the last partition of A as A^b. That is $A^f \cup A^b = A$ and $A^f \cap A^b = \emptyset$, where the subscripts f and b are short for "foreground" and "background" respectively. Prior to

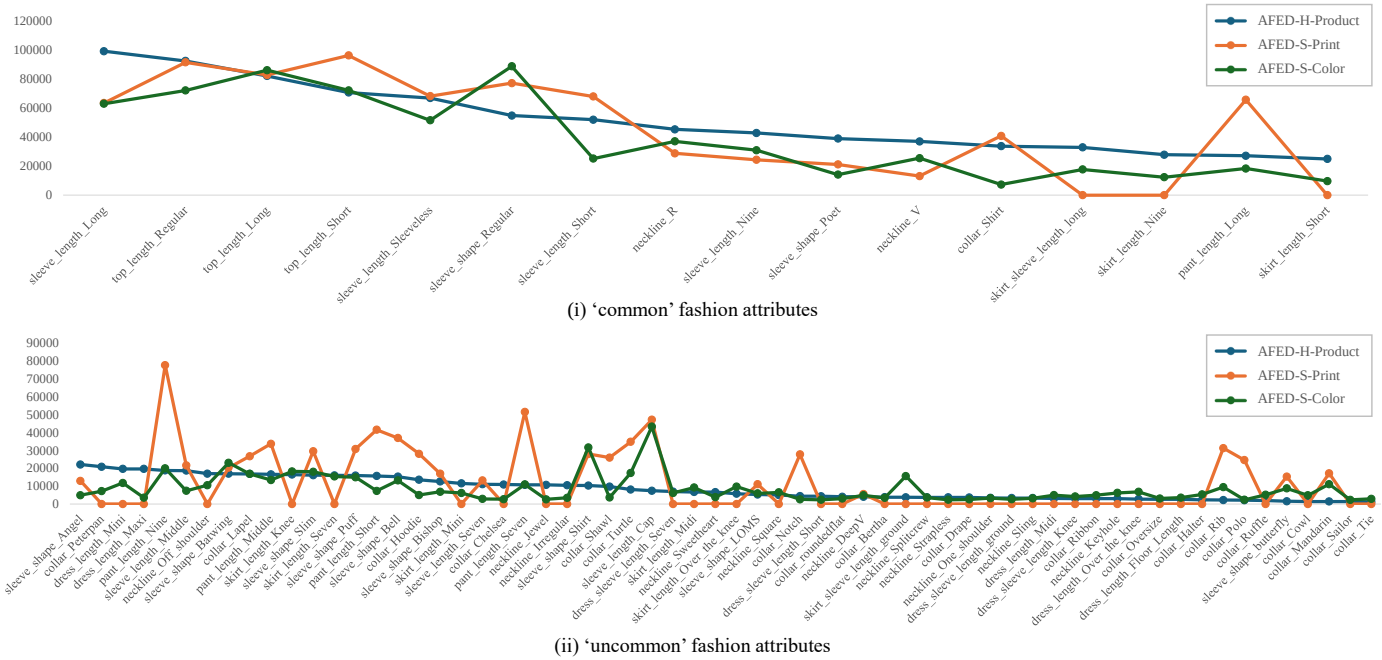


Fig. 4: Statistic distribution of the fine-grained attributes in the AFED dataset.

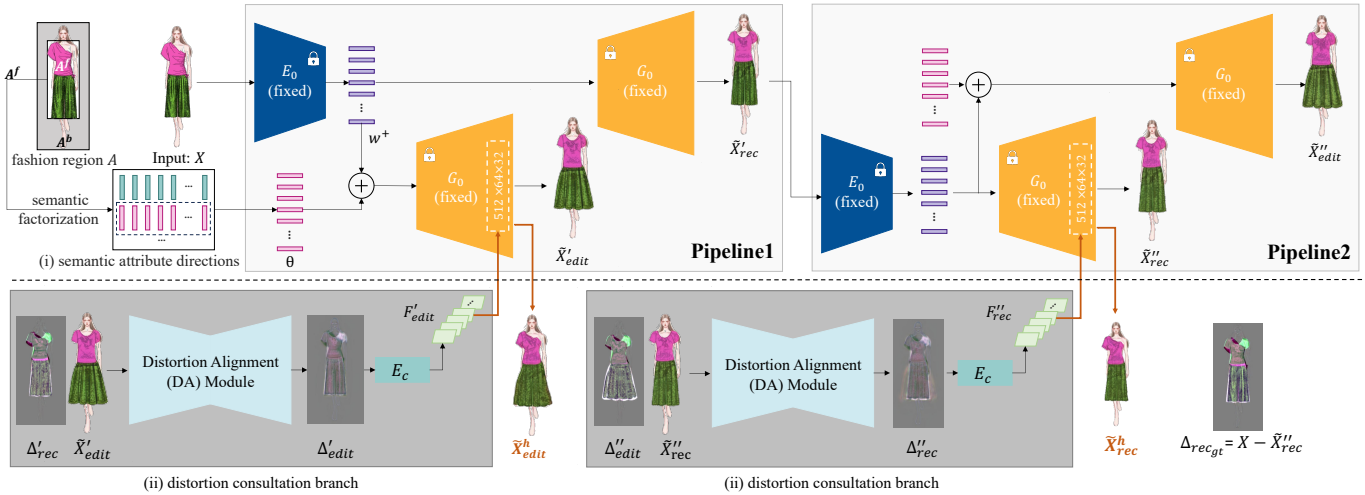


Fig. 5: The training framework of Twin-Net.

training, we compute the semantic directions associated with region $\mathbf{A}^f$ by solving Equation 1.

$$(\boldsymbol{J}_b^T \boldsymbol{J}_b + a\boldsymbol{I})^{-1} \boldsymbol{J}_f^T \boldsymbol{J}_f \boldsymbol{n} = \lambda \boldsymbol{n} \tag{1}$$

where $a = \tau tr(\boldsymbol{J}_b^T \boldsymbol{J}_b)$ and $\tau = 1e^{-3}$ is a small scaling factor and Jacobian matrix $\boldsymbol{J}_b$ represents the derivative of pixel values with respect to the latent code. $\boldsymbol{n}$ is eigenvectors corresponding to the largest eigenvalues λ. For training, the top 8 directions are selected, and the degree is set to 6. To avoid the influence of color during the editing process, we use layer-wise editing [15] to add the direction to the first 8 layers of StyleGAN. Fig 6 shows the attribute editing effects of the top 8 direction vectors on three sub-datasets during the training process. It can be seen that, since only the clothing region was factorized, when adding semantic directions, one or more fashion attributes were edited (for example, in the AFED-S-Color dataset, direction $\mathbf{n}_1$ edited both clothing length and sleeve length; in the AGED-H-Product dataset, direction $\mathbf{n}_1$ also edited the pose), while the identity information (ID) of the model is well preserved.

Consequently, we could do manipulation at the semantic level in the low-rate latent space. To obtain edited image, a semantic attribute direction $\boldsymbol{n} \in \mathbb{R}^{18\times512}$ is added to $\mathcal{W}+$ and achieve the low-rate edited latent code $\mathbf{W}^+_{edit}$,

$$\mathbf{W}^+_{edit} = \mathbf{W}^+ + s\boldsymbol{n} \tag{2}$$

where $\mathbf{W}^+_{edit} \in \mathbb{R}^{18\times512}$, and s constrains the degree of the edits. Then, the generator G_0 takes $\mathbf{W}^+_{edit}$ as input to generate a low-fidelity edit image $\tilde{\mathbf{X}}'_{edit} = G_0(\mathbf{W}^+_{edit})$, $\tilde{\mathbf{X}}'_{edit} \in \mathbb{R}^{H\times W\times3}$. It can be observed that the obtained $\tilde{\mathbf{X}}_{edit}$, as shown in Fig. 5, is also inaccurate, having similarly lost the 'one-shoulder' attribute like $\tilde{\mathbf{X}}'_{rec}$. In other words, edits conducted within the low-rate latent space result in imprecision, attributed to the loss of information.

Since there is no ground truth for $\tilde{\mathbf{X}}'_{edit}$, we propose using the difference between $\mathbf{X}$ and $\tilde{\mathbf{X}}'_{rec}$ as a substitute to supplement the missing information of $\tilde{\mathbf{X}}'_{edit}$. This approach leverages the inherent relationship between these two image

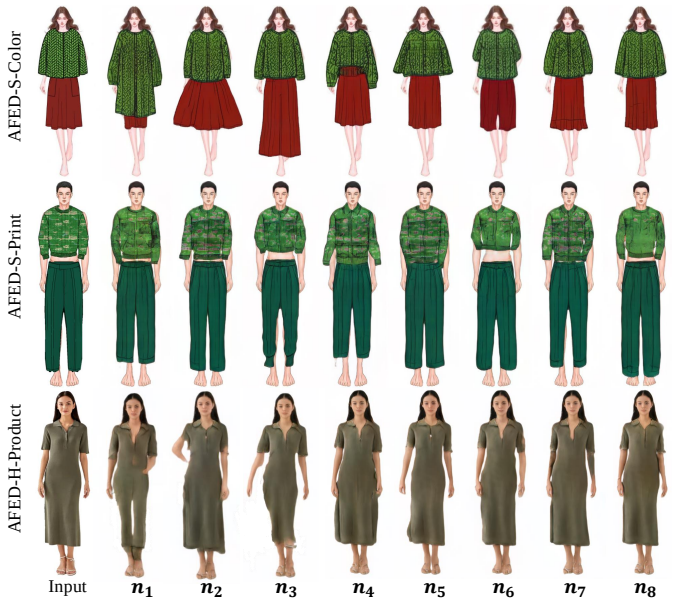


Fig. 6: Visualization of editing using the top 8 direction vectors on three sub-datasets during training. For each direction, one or more attributes are edited.

pairs. The difference can be expressed as $\Delta^{'}_{rec} = \mathbf{X} - \tilde{\mathbf{X}}^{'}_{rec}$, $\Delta^{'}_{rec} \in \mathbb{R}^{H\times W\times 3}$. Certainly, the missing information between $\mathbf{X}$ and $\tilde{\mathbf{X}}^{'}_{rec}$ will not align with the missing information between the ground truth of $\tilde{\mathbf{X}}^{'}_{edit}$ and the generated $\tilde{\mathbf{X}}^{'}_{edit}$. Directly incorporating this information $\Delta^{'}_{rec}$ as supplementary data into the generated $\tilde{\mathbf{X}}^{'}_{edit}$ may introduce artifacts in the resultant image $\tilde{\mathbf{X}}^{h}_{edit}$.

To better approximate the missing information of the reconstructed image to the edited image, a distortion alignment module (DA) is adopted in the distortion consultation branch, $\Delta^{'}_{edit} = DA(\Delta^{'}_{rec}, \tilde{\mathbf{X}}^{'}_{edit})$, $\Delta^{'}_{edit} \in \mathbb{R}^{H\times W\times 3}$. Subsequently, the consultation encoder E_c projects the $\Delta^{'}_{edit}$ into a high-rate latent map $\mathbf{F}^{'}_{edit} = E_c(\Delta^{'}_{edit})$, $\mathbf{F}^{'}_{edit} \in \mathbb{R}^{512\times 64\times 32}$. Taking the high-rate feature map $\mathbf{F}_{edit}$ and edit latent code $\mathbf{W}^{+}_{edit}$ as input, the generator G_0 is expected to generate a high-fidelity edit image $\tilde{\mathbf{X}}^{h}_{edit} \in \mathbb{R}^{H\times W\times 3}$ as:

$$\tilde{\mathbf{X}}^{h}_{edit} = G_0(\mathbf{W}^{+}_{edit}, \mathbf{F}^{'}_{edit}) \tag{3}$$

Notably, the optimization of the distortion alignment module DA and the consultation encoder E_c is not feasible since the absence of ground truth for $\tilde{\mathbf{X}}^{h}_{edit}$. They will share weights with the DA and E_c in Pipeline 2, respectively.

The goal of Pipeline 2 is to generate a high-quality reconstructed image $\tilde{\mathbf{X}}^{h}_{rec} \in \mathbb{R}^{H\times W\times 3}$. This enables the entire framework to be trainable, as there will be ground truth for the reconstructed image; that is, the input image $\mathbf{X}$ serves as the ground truth for $\tilde{\mathbf{X}}^{h}_{rec}$. Meanwhile, it ensures that the network can achieve high-quality inversion performance while maintaining the editability of the inverted code. Specifically, we take the low-fidelity reconstructed image $\tilde{\mathbf{X}}^{'}_{rec}$ from Pipeline 1 as input. The missing information between the high-quality edited image $\tilde{\mathbf{X}}^{h}_{edit}$ and the reconstructed edited image $\tilde{\mathbf{X}}^{''}_{edit}$ is used to simulate $\Delta^{''}_{rec}$ as the supervision signal to train its distortion alignment module. Then, the generator G_0 uses $\mathbf{W}^{'+}_{edit}$ (which adds a semantic attribute direction to the latent code $\mathbf{W}^{'+}$ using Equation 2) to generate the low-fidelity edited image $\mathbf{X}^{''}_{edit}$. Concurrently, the consultation encoder E_c projects $\Delta^{''}_{rec}$ into the high-rate feature map $\mathbf{F}^{''}_{rec} = E_c(\Delta^{''}_{rec})$. Finally, the generator G_0 takes $\mathbf{F}^{''}_{rec}$ and the low-rate latent code $\mathbf{W}^{'+}$ to produce the high-quality reconstructed image $\tilde{\mathbf{X}}^{h}_{rec}$ which can be written as:

$$\tilde{\mathbf{X}}^{h}_{rec} = G_0(\mathbf{W}^{'+}_{rec}, \mathbf{F}^{''}_{rec}) \tag{4}$$

Losses. To reconstruct the input image, the general reconstruction losses, including the L_2, perceptual loss [71], and identity loss [72] between the input $\mathbf{X}$ and output $\tilde{\mathbf{X}}^{h}_{rec}$ are adopted. Given the intricate line drawings and edges in fashion design, we propose a new loss function, the Sobel Loss, to enhance the details of the inversion result. The Sobel operator [73] S is adopted to compute the edge maps of $\tilde{\mathbf{X}}^{h}_{rec}$ and $\mathbf{X}$, respectively. Taking $\mathbf{X} \in \mathbb{R}^{H\times W\times 3}$, the L_1 loss between these two edge maps is calculated.

$$L_{sobel} = \left| S(\mathbf{X}) - S(\tilde{\mathbf{X}}^{h}_{rec}) \right|_1 \tag{5}$$

where S represents the Sobel operator. Furthermore, we impose a constraint on $\Delta^{''}_{rec}$ to approximate $\Delta_{rec_{gt}}$. This particular loss is referred to as the align loss, denoted as L_a.

$$L_a = \left| \Delta_{rec_{gt}} - \Delta^{''}_{rec} \right|_1 \tag{6}$$

4.2 Editing Algorithm: PairsPCA

Building on the successful image inversion and editing capabilities of Twin-Net, we further explore the potential for fine-grained fashion attribute editing in the $\mathcal{W}+$ space. Fashion attribute editing requires that the calculated attribute directions be accurate, disentangled, and continuous. To achieve this, we propose PairsPCA, a few-shot attribute editing algorithm. The details of this method are introduced in the following.

Inference of Editing. Our goal is to edit an attribute from a to b. Following the structure of the inversion method, i.e., Twin-Net, for a given image $\mathbf{X}_{real}$, we first obtain the latent code $\mathbf{W}^{+}_{real}$ of $\mathbf{X}_{real}$ through the encoder E_0. Then, the next step involves adding the desired direction $\boldsymbol{d}^{ab}$ to $\mathbf{W}^{+}_{real}$ as follows:

$$\mathbf{W}^{+}_{real_{ab}} = \mathbf{W}^{+}_{real} + m\boldsymbol{d}^{ab} \tag{7}$$

where $m>0$ controls the degree of the edit. The high-fidelity edited image is then computed as:

$$\tilde{\mathbf{X}}^{h}_{real_{ab}} = G_0(\mathbf{W}^{+}_{edit_{ab}}, \mathbf{F}^{'}_{edit}) \tag{8}$$

The calculation of $\mathbf{F}^{'}edit$ is elaborated in Subsection 4.1. It can be observed that the main point of editing is to add the desired direction $\boldsymbol{d}^{ab}$. As introduced above, during the training phase, the direction to be added represents semantic attribute directions, which provide a solid foundation for attribute manipulation at the semantic level. To further

enable fine-grained attribute editing, we propose estimating the attribute direction $\boldsymbol{d}^{ab}$ in a few-shot manner.

Attribute Direction Estimation. The set of attribute directions is denoted as $\mathbf{D} = \{(\mathbf{I}_k^a, \mathbf{I}_k^b)\}$, where $k \in [1, n]$ represents the k_{th} attribute image pair. I_k^a signifies the k_{th} image with attribute a, and $\mathbf{I}_k^b$ represents the image with attribute b modified from image $\mathbf{I}_k^a$. The attribute direction in the latent space $\mathcal{W}+$ is estimated as (1) Project the image pair $(\mathbf{I}_k^a, \mathbf{I}_k^b)$ into the latent space $\mathcal{W}+$ via the Encoder E_0. This is represented as $\mathbf{W}_{a_k}^+ = E_0(\mathbf{I}_k^a), \mathbf{W}_{b_k}^+ = E_0(\mathbf{I}_k^b)$, respectively. (2) The editing direction $\boldsymbol{d}_k^{ab}$ from attribute a to attribute b in the k_{th} image pairs is calculated as $\boldsymbol{d}_k^{ab} = \mathbf{W}_{ak}^+ – \mathbf{W}_{bk}^+$, $\boldsymbol{d}_k^{ab} \in \mathbb{R}^{18\times512}$. (3) $\boldsymbol{d}_k^{ab}$ is reshaped into a 1-dimensional vector $\boldsymbol{d}_k^{r,ab} \in \mathbb{R}^{1\times9216}$. (4) All direction editing vectors $\boldsymbol{d}_k^{r,ab}$ are combined into a matrix $\mathbf{D}^{ab} = [\boldsymbol{d}_1^{r,ab}, \boldsymbol{d}_2^{r,ab}, \ldots \boldsymbol{d}_k^{r,ab}, \ldots, \boldsymbol{d}_n^{r,ab}]$, $\mathbf{D}^{ab} \in \mathbb{R}^{n\times9216}$. (5) The column vector $\mathbf{v}^* \in \mathbb{R}^{(1\times9216)}$ of the V matrix is used as the final editing direction from attribute a to b, represented as $\boldsymbol{d}^{ab} = Reshape(\mathbf{v}^*)$, $\boldsymbol{d}^{ab} \in \mathbb{R}^{18\times512}$. where the $\mathbf{v}^* \in \mathbb{R}^{1\times9216}$ is associated with the highest singular value performed Singular Value Decomposition on $\mathbf{D}^{ab}$.

Attribute Image Pairs. To obtain the image pairs, we first collect images for each fashion attribute, e.g. 'V Neckline' which is one type of neckline. To create a counterpart pair with the 'V Neckline' images, we manually revise the corresponding part to another type of neckline using Photoshop, e.g., revise the 'V Neckline' to 'Round Neckline' while keeping the rest the same. For each editing group between two attributes ('V Neckline' and 'Round Neckline') within the same category (necklines), 20 image pairs are prepared. For categories related to length variation, such as sleeve length, only 20 short-to-long image pairs are needed. Consequently, the total number of manually created images for 78 fine-grained fashion attributes is 860. These were completed in two days with the assistance of five experts, resulting in low costs.

5 Experiments

This section is organized to demonstrate the effectiveness of the proposed approach by answering three questions:

- Is the introduced inversion method superior to state-of-the-art inversion methods?
- Is the proposed editing algorithm more advanced than existing editing methods?
- Can our inversion model and editing algorithm truly achieve the set goal, i.e., any fashion attribute editing?

Specifically, Subsection 5.1 introduces the experimental settings. Subsection 5.2 demonstrates the advantages of the proposed inversion method compared to GAN-based and Diffusion-based methods. Meanwhile, we present ablation study results to assess the effectiveness of the proposed technical components in Twin-Net. To further illustrate its generalization ability, we also present the inversion results on the general domain. Additionally, Subsection 5.3 focuses on showcasing the advancements of the proposed editing algorithm. We report the editing accuracy and conduct a disentanglement analysis to prove that our editing method indeed shows advancement. Finally, in Subsection 5.4, we present the results of our approach, i.e., using our inversion model to obtain the latent codes and then using our editing method to manipulate the fashion attribute, thus achieving any fashion attribute editing, including fine-grained attribute editing, independent attribute editing, and continuous attribute editing. We also present the editing results on different body shapes to demonstrate its robustness.

5.1 Experinment Settings

Dataset. We conducted experiments on three sub-datasets (AFED-S-Color, AFED-S-Print, AFED-H-Product) to verify the fashion attribute editing capabilities of the proposed dataset. For the AFED-S-Color and AFED-S-Print sub-datasets, 90% of the data is used for training and 10% for testing. The AFED-H-Product sub-dataset contains 210,000 images for training and 20,000 images for testing.

Additionally, to demonstrate the generalization ability of our proposed Twin-Net framework, we conducted experiments on a public general face dataset. We trained Twin-Net on the FFHQ [70] dataset and tested it on the Celeb-HQ [74] dataset.

Baseline Comparisons. We benchmarked the proposed Twin-Net with state-of-the-art GAN inversion methods such as pSp[24], e4e[25], ReStyle[29], HFGI[31], StyleT[75], FeatureStyle[76], and StyleRes[30]. In addition, we report the results with diffusion-based methods, including pix2pix-zero [77], DragDiffusion [6] and RegionDrag [78].

To demonstrate the effectiveness of the proposed editing algorithm, we further present comparative results with advanced image editing algorithms, including GANSpace [38], InterFaceGAN [15], ReSeFa [46] and DragGAN [19].

Evaluation Metrics. Following standard practice, we report results based on five metrics to evaluate the performance of inversion methods on two tasks: reconstruction ability and editability. When evaluating the reconstruction performance of methods, we focus on the similarity between the reconstruction result of the methods and the input image without performing any image editing operations. To this end, we use the following three evaluation metrics: mean squared error (MSE)[79], structural similarity index measure (SSIM)[80], and learned perceptual patch similarity (LPIPS)[81]. MSE measures the mean squared error between the reconstruction result and the input image. SSIM evaluates the visual similarity between the reconstruction result and the input image. LPIPS computes perceptual similarity. To standardize image size across these evaluation metrics, we used 1024x512 pixels for AFED-S-Color and AFED-S-Print, 512x256 pixels for AFED-H-Product.

In addition to these three metrics indicating reconstruction performance, we include the Fréchet Inception Distance (FID) [82] and Kernel Inception Distance [83], which evaluates the realism of the edited images. To calculate the Editing-FIDs and Editing-KIDs, we revise specific attributes and compute the FIDs and KIDs between the original and edited images. For a fair comparison, all Editing-FIDs and Eidting-KIDs are based on the results of our proposed editing method. Specifically, the 78 fine-grained attributes can be categorized into six groups: neck, sleeve shape, sleeve length, top length, pant length, and dress length. We select a 'start attribute' from each group: 'Round Neckline', 'Regular Sleeves', 'Short Sleeves', 'Short Top', 'Shorts', and 'Mini Dress'. Then, we selectively edit the starting attribute to other attributes within the same group, calculate the

TABLE 2: Comparison of the inversion methods on the AFED dataset. The abbreviation Ave. refers to Average.

Method	AFED-S-Color					AFED-S-Print					AFED-H-Product				
	Reconstruction			Editing		Reconstruction			Editing		Reconstruction			Editing	
	SSIM ↑	LPIPS ↓	MSE ↓	FID ↓	KID ↓	SSIM ↑	LPIPS ↓	MSE ↓	FID ↓	KID ↓	SSIM ↑	LPIPS ↓	MSE ↓	FID↓	KID↓
pSp[24]	0.9286	0.0669	0.0145	39.55	0.0100	0.9190	0.0661	0.0122	51.63	0.0213	0.8383	0.1170	0.0325	57.42	0.0196
e4e[25]	0.9181	0.0671	0.0182	40.15	0.0077	0.9191	0.0629	0.0160	49.30	0.0124	0.8139	0.1290	0.0385	59.46	0.0190
ReStyle[29]	0.9238	0.0763	0.0155	39.28	0.0099	0.9120	0.0654	0.0161	50.46	0.0151	0.8067	0.1386	0.0397	60.91	0.0194
HFGI[31]	0.9330	0.0608	0.0126	38.39	0.0084	0.9302	0.0590	0.0122	48.26	0.0118	0.8890	0.0988	0.0221	55.94	0.0169
StyleTransformer[75]	0.9351	0.0625	0.0121	37.20	0.0073	0.9317	0.0664	0.0113	46.24	0.0161	0.8413	0.1126	0.0321	52.24	0.0149
FeatureStyle[76]	0.9375	**0.0541**	0.0117	41.22	0.0124	0.9238	0.0564	0.0158	52.92	0.0137	0.9187	0.0687	0.0164	51.54	0.0152
StyleRes[30]	0.9098	0.0736	0.0203	41.37	0.0076	0.9119	0.0691	0.0183	54.92	0.0129	0.8033	0.1346	0.0414	60.31	0.0200
pixel2pixel-zero [77]	0.9322	0.0896	0.0118	145.30	0.0367	0.9324	0.0871	0.0147	168.56	0.0414	0.9419	0.0894	0.0090	67.52	0.0195
DragDiffusion [6]	**0.9502**	0.0804	**0.0062**	-	-	**0.9487**	**0.0509**	0.0103	-	-	**0.9606**	**0.0581**	**0.0058**	-	-
RegionDrag [78]	0.9336	0.0873	0.0106	-	-	0.9330	0.0807	0.0122	-	-	0.9403	0.0919	0.0097	-	-
Twin-Net(Ours)	0.9448	0.0549	0.0091	**36.53**	**0.0067**	0.9369	0.0573	**0.0099**	**45.20**	**0.0113**	0.9381	0.0726	0.0108	**50.85**	**0.0143**

FID and KID for each and report the average values. For the 'Round Neckline', we edit it to 'Shirt Collar', 'Square Neckline' and 'V Neckline'. For the 'Regular Sleeves', we edited it to 'Bell Sleeves', 'Puff Sleeves', and 'Shirt Sleeve'. For 'Short Sleeves', 'Short Top', 'Shorts', and 'Mini Dress', we edit them to their corresponding longer versions. All the test images for each 'start attribute' were randomly selected from the test sets of three sub-datasets, with 150 images for each attribute. Furthermore, to avoid the influence of the image format on the evaluation metric results [84], all evaluations use the PNG image calculation results.

Implementation Details of Inversion Methods. We use the pre-trained StyleGAN2 [69] on AFED as the pre-trained generator G_0 network for all inversion methods. All inversion baselines were trained on the three AFED sub-datasets, using either the training code released by the original authors or faithfully reproduced methods [30]. For Twin-Net, we use e4e [25] as the basic encoder E_0. The distortion alignment (DA) module and the consultation encoder E_c module are derived from HFGI [31]. Both G_0 and E_0 are fixed during training. We adopt the Adam [85] optimizer and set the learning rate to $1e^{-4}$. The batch size is set to 8, and the number of iterations is 100,000. The weight coefficients for the L_2 loss, perceptual loss, identity loss, L_{sobel}, and L_a are set to 1, 0.8, 0.1, 0.5, and 0.1, respectively.

For diffusion-based methods, we employ diffusion models as described in their original papers. Specifically, we utilize Stable Diffusion-1.4[86] for pix2pix-zero[77], and Stable Diffusion-1.5[41] for DragDiffusion[6] and RegionDrag [78] as the base model. For pix2pix-zero, we obtain reconstruction results using the image and its original text embedding. As for DragDiffusion, we fine-tune the LoRA [87] for the input image and obtain its reconstruction results without the need for additional masks and points. As for RegionDrag, we obtain the construction result by setting the same source and target region on the input image.

For testing the generalization ability, following the general practice, we trained Twin-Net on the FFHQ dataset [70] and tested it on the Celeb-HQ dataset [74].

Implementation Details of Editing Algorithms. The attribute editing directions for GANSpace [38], InterFaceGAN [15], and ReSeFa [46] are obtained via Principal Component Analysis, linear SVMs training, and factoring the target image region, respectively. For GANSpace, we sampled 10,000 random vectors in the $\mathcal{Z}$ space and computed their latent codes in the $\mathcal{W}$ space. Subsequently, we performed Principal Component Analysis on these 10,000 latent codes. We adopted the first ten principal components as editing directions and implemented layer-wise editing, that is, applying these directions to specific layers. Following the method in [15], we divided the 18 layers of StyleGAN into five groups: [0-1, 2-3, 4-5, 6-7, 8-17]. After removing the group [8-17] used for color manipulation, each attribute direction will have 40 choices (10×4). To identify the semantic boundaries of fashion attributes in InterfaceGAN [15], we first trained a ResNet-50 attribute prediction model, following the approach described in the paper. We then assign attribute scores to all 500K synthetic images, sorted them, and selected the top 10K and bottom 10K samples as candidate datasets. We used the latent codes of these samples in the $\mathcal{W}$ space to train an SVM to define the semantic boundaries. In ReSeFA [46], the calculation of attribute directions is based on the specific regions where the attributes are located. For each region, the top 8 attribute directions are selected for fashion attribute semantic search. For DragGAN [19], we mask the target region of the fashion image and provide the start and target points. The best-matching images are manually selected for comparison. For diffusion-based methods, we edit the input image following the original papers. Specifically, we calculate the edit direction based on the prompts provided for pix2pix-zero[77], specify the mask for the editing area and define the start and end points for DragDiffusion[6].

5.2 Image Inversion Evaluation

5.2.1 Part I Comparison with GAN-based baselines

Quantitative Results. In Tab. 2, across all three sub-datasets, our method outperforms all competing GAN-based inversion methods in terms of SSIM, MSE, Editing-FID, and Editing-KID (Ave.). Specifically, the improvements in performance of the proposed method on the SSIM and MSE metrics are not as significant as on the FID metric. This suggests that while our GAN inversion method may have relatively weaker advantages in detail preservation and image reconstruction accuracy, it demonstrates superior performance in the overall effect of image editing.

Simultaneously, it achieves LPIPS scores comparable to those of FeatureStyle [76], suggesting a similar level of perceptual consistency. Please refer to Tab I and II in the Supplementary Materials for FID and KID details.

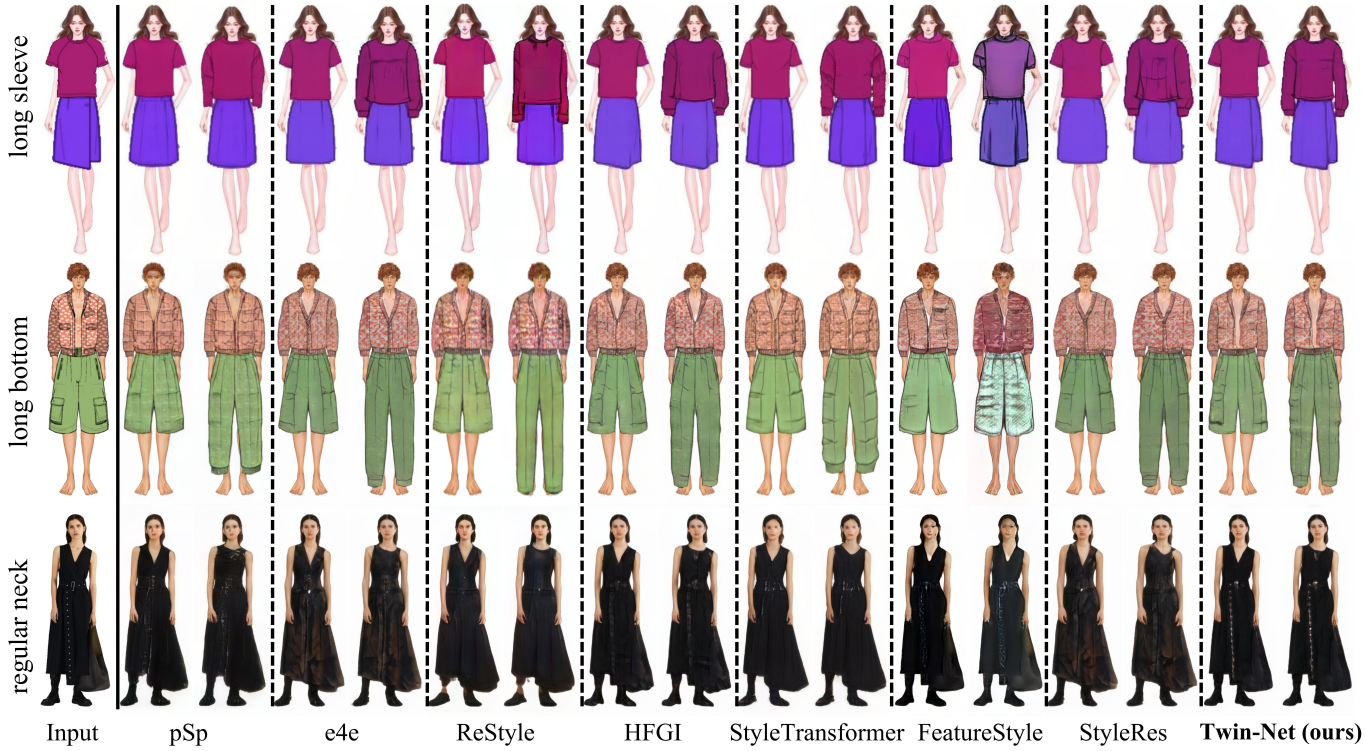


Fig. 7: Comparison results of GAN-based methods. For each pair, from left to right is the inverted and edited result.

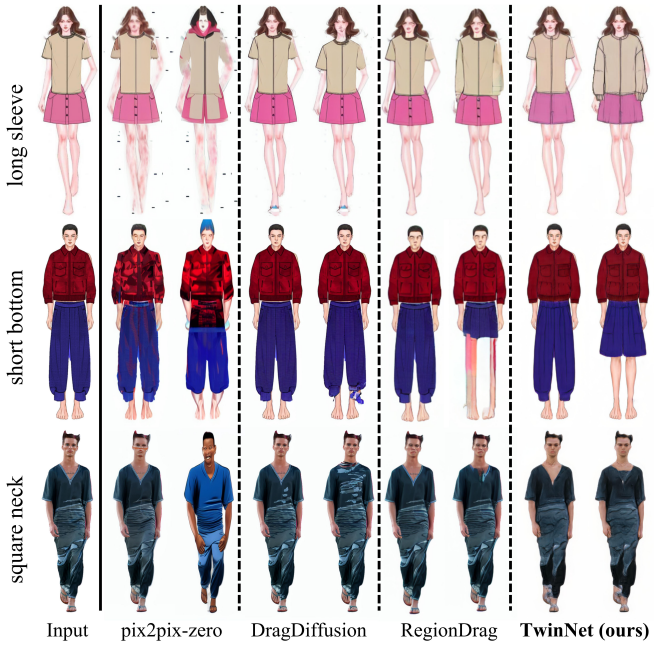


Fig. 8: Comparison results of diffusion-based methods. For each pair, from left to right is the inverted and edited result.

Qualitative Results. We present the qualitative results of image inversion and editing in Fig. 7 and draw the following insights: 1) Accurate Fine-Grained Attributes: The 2nd, 3rd, 5th, and 6th rows in Fig. 7 shows results with fine-grained fashion attributes. Our method outperforms all baselines in accurately reconstructing fashion attributes that are common in daily life but rarely seen in previous research. For instance, in the sixth row, our method reconstructs the fine-grained 'Square Neckline' attribute, a task both HFGI and StyleTransformer could not achieve. These results collectively demonstrate that our method excels in reconstructing fine-grained attributes. 2) Design detail consistency: The results also illustrate that our method demonstrates superior consistency in design details compared to the baselines, e.g., for the tops with an opening in rows 1 and 4, only our results accurately reflect the split in the middle of the top. Although HFGI, Restyle, and StyleT reconstructed the split for the top in the first row, they failed to reconstruct the arc shape of the lower part of the top. 3) Data dependencies: Besides its advanced performance in image reproduction, our method exhibits fewer dependencies on training data than other methods. This is evident in 6th row. Baselines are influenced by the training dataset and reconstruct the 'Square Neckline' to 'V Neckline' or 'Round Neckline'. 4) High-quality independent editing: Fig. 7 also shows attribute editing results on all inversion baselines using our proposed editing algorithm. It can be observed that, except for the FeatureStyle method, all other methods execute the editing of target attributes. This indicates that the attribute editing algorithm we proposed possesses a certain level of universality. Moreover, while facilitating target attribute editing, our approach ensures the invariance of the remaining attributes in all inversion methods (except FeatureStyle). This illustrates that our proposed editing algorithm can edit attributes independently. When combined with the GAN inversion method we proposed, we can thus achieve high-quality image attribute algorithm.

5.2.2 *Part II Comparison with Diffusion-based baselines*

Quantitative Results. We report the results of three recent diffusion-based methods in the last three rows of Table 2. Undeniably, the diffusion-based model excels in product image inversion, as indicated by results on SSIM, LPIPS, and MSE metrics. However, the poorer performance on Editing-FIDs suggests that this type of method may struggle with attribute editing. Meanwhile, it is noticeable that

TABLE 3: Quantitative results of ablation study.

Method	Reconstruction			Editing-FIDs	
	SSIM ↑	LPIPS ↓	MSE ↓	SL(+)	SL(-)
Pipeline1 w/o DA	0.9181	0.0671	0.0182	29.09	30.11
Pipeline1 w/ DA	0.9178	0.0686	0.0170	31.15	35.58
Pipeline1 + Pipeline2	**0.9326**	**0.0620**	**0.0122**	**30.34**	**28.37**

TABLE 4: Quantitative results of ablation study. 'w/o all' refers to Twin-Net w/o Sobel Loss and w/ random direction.

Method	Reconstruction			Editing-FIDs	
	SSIM ↑	LPIPS ↓	MSE ↓	SL(+)	SL(-)
w/o all	0.9326	0.0620	0.0122	30.34	28.37
w/o Sobel Loss	0.9435	0.0554	0.0094	28.20	27.47
w/ random direction	0.9344	0.0611	0.0118	29.84	27.93
w/all (**Ours**)	**0.9448**	**0.0549**	**0.0091**	**27.29**	**26.45**

the diffusion-based model has a significant drawback in computation time compared to GAN-based models. The editing inference times for diffusion-based methods are markedly higher: 35 seconds for pix2pix-zero, 78 seconds for DragDiffusion, and 1.5 seconds for RegionDrag, compared to GAN-based methods, with Twin-Net being 0.14 seconds and thus 10 to 500 times faster. The slow editing process not only prevents real-time editing but also degrades the user experience due to long wait times.

Qualitative Results. As shown in Fig. 8, our observations are as follows: 1) The diffusion-based method demonstrates a superior inversion capability in terms of the realism of the generated images. For instance, the fabric prints and textures produced by this type of method are more realistic than those of our approach. 2) The diffusion-based model, when not fine-tuned (e.g., pix2pix-zero, RegionDrag), performs poorly in the sketch domain. In contrast, DragDiffusion with LoRA fine-tuning performs better. However, it is time-consuming, as it requires fine-tuning for each input image. 3) Most importantly, the three diffusion-based methods exhibit a weak ability in attribute editing, even for the most basic manipulations such as length adjustment, e.g., the first two rows of Fig.8, these methods attempted revisions but ultimately failed. The qualitative results further support the conclusion drawn above, i.e., the current diffusion models although have a good performance on image reconstruction, brings difficulties in editing. This contrast highlights the research potential for improving diffusion-based methods in attribute editing in future work.

5.2.3 Part III Ablation Study

We conducted experiments to evaluate the efficacy of each proposed design choice, including the Twin-Net structure, those without Sobel Loss, with Sobel Loss, with random direction, and with semantic direction on the AFED-S-Color dataset. The quantitative results are detailed in Tab. 3 and Tab. 4, and qualitative results are depicted in Fig. 9. The structure of the Twin-Net, the Sobel Loss and semantic direction significantly enhance the inversion performance across all evaluation metrics. Further analysis of the visualized inversion results yielded the following insights:

1) Global shape and grained attribute accuracy: The proposed semantic attribute direction has notably improved the global shape and contributed to attribute consistency at

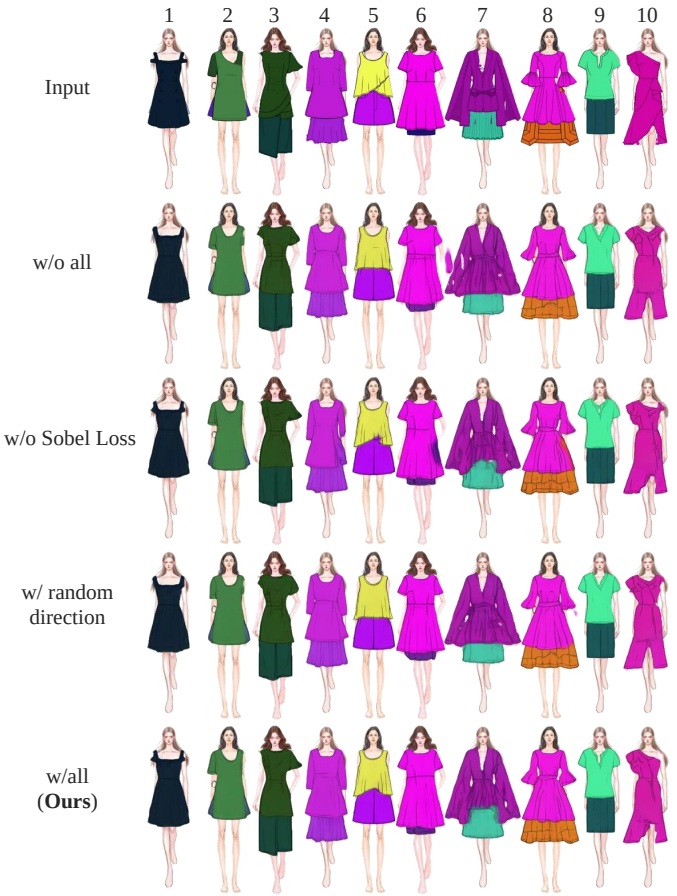


Fig. 9: Qualitative results of ablation study.

the semantic level. For instance, in the 1st, 2nd, 3rd, and 4th columns of Fig. 9, the semantic direction significantly aided the successful inversion of 'Cami Off-shoulder', 'One Shoulder', 'Asymmetric Skirt', and 'Square Neckline'.

2) Design details and local region: The proposed Sobel Loss contributes to local design details, as seen in the 3rd column of Fig. 9. It ensures consistent auxiliary lines depicting folds and layering, producing more realistic clothing representations. Despite its minor improvements at the image level, especially compared to the global shape and attribute accuracy from the semantic direction, Sobel Loss enhances the fidelity of inversion results in fashion design. It guides accurate local region reconstruction, complementing the semantic direction's global shape support. This is evident in columns 4 to 10 of Fig. 9. For instance, while the semantic direction identifies the 'Split Crew' neckline in the 9th column, it struggles to reconstruct this attribute without Sobel Loss accurately. This is also evident in the 10th column, where only our method successfully inverts the one-shoulder ruffle dress.

5.2.4 Part IV Comparison Results on General Domain

Quantitative Results. Following [30], we reported a comparison of the reconstruction and editable ability in Tab. 5. The comparison shows that our Twin-Net outperforms all competing methods regarding SSIM, MSE, and FIDs. The LPIPS of Twin-Net, at 0.10, is slightly lower than that of StyleRes, which stands at 0.09. Despite this minor setback in the LPIPS metric, the overall performance of Twin-Net is commendable.

TABLE 5: Quantitative comparison of the reconstruction and editing on the CelebA-HQ dataset. For reconstruction, we report SSIM, LPIPS and MSE scores. For editing, we report FID metrics for smile addition (+) and removal (-).

Method	Reconstruction			Editing-FIDs	
	SSIM ↑	LPIPS ↓	MSE ↓	Smile(+)	Smile(-)
pSp[24]	0.75	0.17	0.03	32.47	34.0
e4e[25]	0.71	0.21	0.03	38.58	39.68
ReStyle[29]	0.73	0.20	-	30.35	33.69
HFGI[31]	0.85	0.13	0.027	25.22	27.10
StyleTransformer[75]	0.75	0.17	0.036	34.32	34.61
FeatureStyle[76]	0.90	0.10	0.019	27.20	26.15
StyleRes[30]	0.90	**0.09**	-	23.52	21.80
Twin-Net(Ours)	**0.94**	0.10	**0.016**	**22.37**	**20.15**

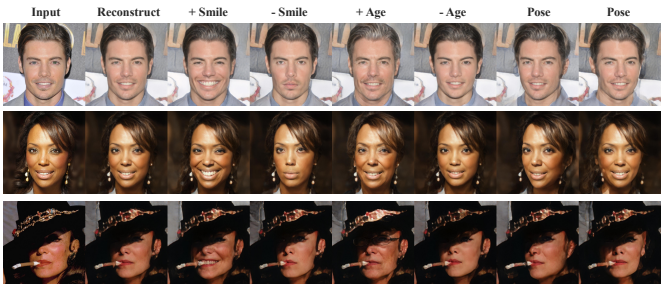


Fig. 10: Qualitative results of Twin-Net on the Celeb-A dataset. The semantic directions are from InterfaceGAN.

Qualitative Results. We present the results of face image inversion and editing in Fig. 10. Our approach excels in maintaining high fidelity to the input images, preserving identity and intricate details during editing. In the 1st row, the background remains unchanged. In the 2nd row, our method successfully reconstructs and preserves the earrings throughout the process. In the 3rd row, we edit target attributes without affecting complex occlusions like hats and cigarettes. These results demonstrate that Twin-Net exhibits robust image reconstruction and editing capabilities on a general dataset.

5.3 Image Editing Evaluation

After checking the advancement of the proposed inversion method, we start to examine the effectiveness of the proposed editing algorithm. For a fair comparison, all results in this section are based on the pretrained model which uses the proposed Twin-Net.

5.3.1 Part I Editing Accuracy

To evaluate the accuracy of an editing algorithm, it is necessary to first prepare the images that require editing. The corresponding attribute-editing algorithm is then applied to modify these images to the target attributes. Finally, the accuracy of the editing algorithm in achieving the desired changes in the target attributes is assessed. A detailed description of this process follows.

Preparation of Test Images for Editing Test images for editing were prepared across three sub-datasets. For the AFED-S-Color and AFED-H-Product subdatasets, test images with six attributes were selected: 'Short Top', 'Short Dress', 'Short Pants', 'Short Sleeves', 'Regular Sleeves' and 'Round Neckline'. The AFED-S-Print dataset, which contains only men's clothing, included test images with the remaining five attributes, excluding 'Short Dress'. For each attribute within each dataset, 150 test images were randomly selected from the corresponding attribute images of test set. In total, (150×6×2 + 150×5 = 2,550) images were selected for testing, and these test images will be made publicly available.

TABLE 6: Attribute recognition accuracy. The abbreviation 'S.' refers to 'Sleeves', and 'N.' refers to 'Neckline', respectively.

Data	Short Top	Short Dress	Short Pants	Short S.	Regular S.	Round N.
AFED-S-Color	98.00	97.33	100.00	97.33	92.67	94.00
AFED-S-Print	95.33	-	100.00	100.00	94.00	96.67
AFED-H-Product	95.33	93.33	99.33	99.33	91.33	92.67

The target attributes are defined as follows: 'Short Top' is edited to 'Long Top'; 'Short Dress' is edited to 'Long Dress'; 'Short Pants' is edited to 'Long Pants'; 'Short Sleeves' is edited to 'Long Sleeves'; 'Regular Sleeves' is edited to 'Bell Sleeves'; 'Round Neckline' is edited to 'Shirt Collar' and 'V Neckline', respectively.

Editing Accuracy Evaluation To evaluate the performance of attribute editing algorithms, we employed the recently open-sourced universal recognition model of Qwen2.5-VL[2]. This model takes an image and a query as input, and then interprets the user's query within the image using the model's capabilities. When using this model, we input the image containing the attribute to be identified, along with the corresponding prompt, and then obtain the model's explanation as output. Specifically, to determine the length of the top in the image, we input the prompt: "What is the top length in the image? Please select long top or short top. Judging criteria: the top length is short if it is above the navel, and long if it is below the navel." For all remaining attribute recognition prompts, please refer to Tab III in the Supplementary Materials.

The attribute recognition accuracy of Qwen2.5-VL on the test images of three subdatasets is shown in Table 6. It can be seen that the recognition accuracy of Qwen-2.5 in test images with known attributes reaches 90%. This shows that the model also has high accuracy in fashion attribute recognition.

Based on Qwen2.5-VL outstanding performance in attribute recognition tasks, we utilized this model to evaluate the ability of attribute editing algorithms, GANSpace [38], InterfaceGAN [15] and ReSeFA [46], in AFED-S-Color, AFED-S-Print, and AFED-H-Product. The results are illustrated in Fig. 11. The horizontal axis lists the target attributes for editing. The vertical axis represents the accuracy with which the Qwen2.5-VL model recognizes the edited attribute as the target attribute. Overall, the proposed Pair-PCA achieved the highest editing accuracy across all target attributes in the three subdatasets. Except for the "long top" and "shirt collar" in the AFED dataset, the editing accuracy for the remaining attributes reaches 80% or higher. Notably, the editing accuracy for the "long sleeve" and "long pants" attributes across the three subdatasets even achieves 90%

2. https://github.com/QwenLM/Qwen2.5-VL/blob/main/cookbooks/

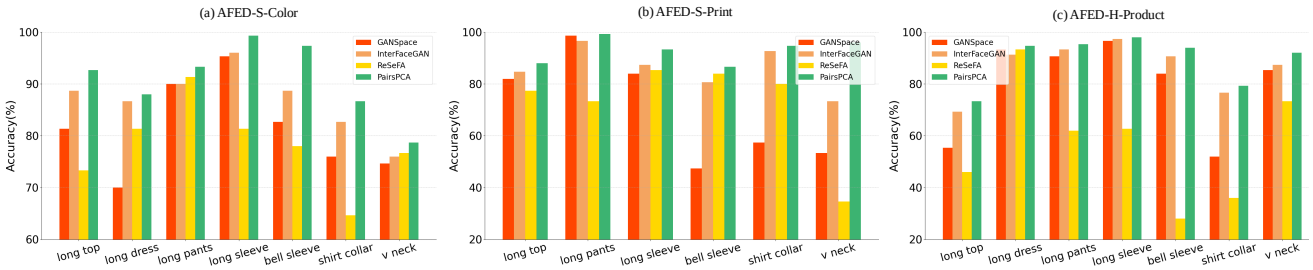


Fig. 11: Editing accuracy of image editing algorithms: GANSpace [38], InterFaceGAN [15] and ReSeFA [46]

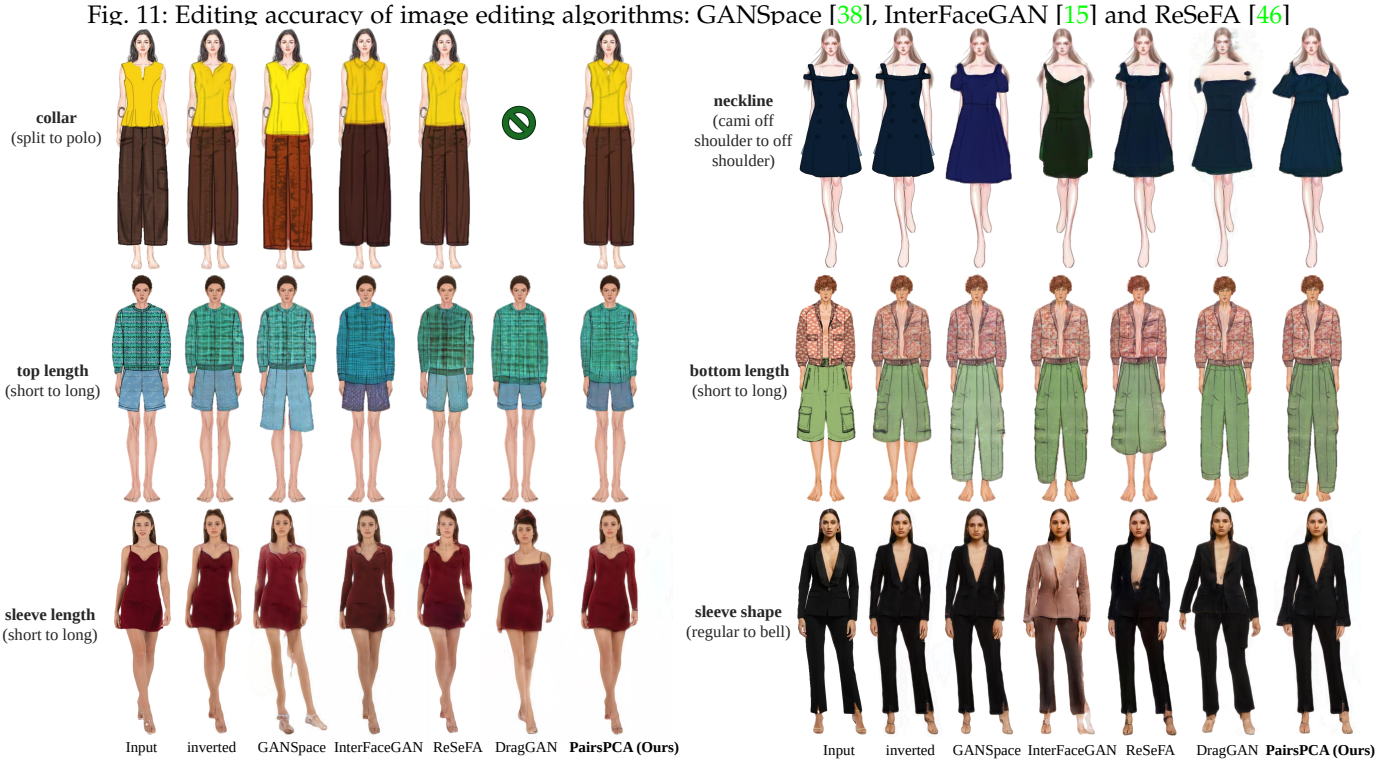


Fig. 12: Comparison image editing algorithms: GANSpace [38], InterFaceGAN [15], ReSeFA [46] and DragGAN [19].

or above. Following that, InterfaceGAN [15], which uses an auxiliary classifier, demonstrated satisfactory performance on multiple attributes by training SVM semantic boundaries on a large set of scored and filtered synthetic data. Next, GANSpace [38], whose editing accuracy is based on the semantic directions obtained from the principal component analysis. These semantic directions require manual verification, resulting in relatively larger fluctuations in identification accuracy across different attributes in the three subdatasets. The method with the lowest accuracy is ReSeFA [46]. ReSeFA depends on the accurate definition of attribute regions in the images. Using a uniform region definition places higher demands on the consistency of model poses across the dataset. Consequently, on the AFED-H-Product dataset, which exhibits significant pose variations, the method's attribute editing accuracy is notably reduced, especially for attributes related to sleeves and necks.

5.3.2 Part II Disentanglement Analysis

In addition to studying the accuracy of attribute editing, we further study the correlation between the semantic directions computed by the GANSpace, InterfaceGAN, ReSeFA editing algorithms, and our proposed PairPCA across three subdatasets. To be more precise, for any two semantic directions $\boldsymbol{n_1}$ and $\boldsymbol{n_2}$, we calculate the cosine similarity between them using the formula $cos(\boldsymbol{n_1}, \boldsymbol{n_2}) = \boldsymbol{n_1}^T \boldsymbol{n_2}$. Here, both $\boldsymbol{n_1}$ and $\boldsymbol{n_2}$ are unit vectors. The value of $cos(\boldsymbol{n_1}, \boldsymbol{n_2})$ is within the range [1,1]). The closer the similarity is to 1 or -1, the lower the degree of decoupling between the two attributes, meaning that when editing attribute $\boldsymbol{n_1}$, the impact on attribute $\boldsymbol{n_2}$ is greater; conversely, the closer the similarity is to 0, the higher the degree of decoupling between the two attributes, and when editing attribute $\boldsymbol{n_1}$, the influence on attribute $\boldsymbol{n_2}$ is less. The results are presented in Table 7. Our method is superior to the baselines on all attributes and datasets, which indicates the advantage of our method on attribute disentaglment.

Qualitative Results. Fig. 12 presents a comparison of the editing results of our method with other state-of-the-art techniques (GANSpace [38], InterFaceGAN [15], ReSeFA [46], and DragGAN [19]). Each example demonstrates that our method significantly outperforms baseline methods in terms of editing quality. ReSeFA and DragGAN have limitations in editing "uncommon" attributes. ReSeFA has minimal impact on the editing results, especially in terms of sleeve and collar types, while DragGAN causes severe deformations even with mask constraints (as shown in the example of sleeve length editing). GANSpace has achieved partial success in editing "uncommon" attributes but still

TABLE 7: Disentanglement analysis. The abbreviation 'L.' refers to 'Length', 'S.' refers to 'Sleeves', 'N.' refers to 'Neckline', and 'C.' refers to 'Collar' respectively.

AFED-S-Color (PairPCA)

	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	-0.02	0.00	0.08	0.06	-0.09	0.01
Dress L.		1.00	0.89	-0.03	0.01	-0.03	0.01
Pant L.			1.00	-0.05	0.00	-0.07	0.06
Sleeve L.				1.00	0.09	-0.08	0.09
Bell S.					1.00	0.07	0.01
Shirt C.						1.00	-0.09
V N.							1.00

AFED-S-Print (PairPCA)

	Top L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	0.04	-0.09	-0.04	-0.01	-0.09
Pant L.		1.00	-0.02	-0.09	0.08	0.00
Sleeve L.			1.00	0.14	-0.11	0.04
Bell S.				1.00	-0.04	-0.01
Shirt C.					1.00	0.00
V N.						1.00

AFED-H-Product (PairPCA)

	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	0.05	-0.03	0.09	-0.02	0.11	-0.08
Dress L.		1.00	0.73	0.08	0.04	0.00	0.03
Pant L.			1.00	0.05	0.04	-0.00	-0.01
Sleeve L.				1.00	0.09	-0.09	0.01
Bell S.					1.00	-0.06	-0.09
Shirt C.						1.00	0.08
V N.							1.00

AFED-S-Color (ReSeFa)

	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C	V N.
Top L.	1.00	0.00	0.01	0.04	0.06	0.02	-0.14
Dress L.		1.00	1.00	-0.01	-0.02	-0.15	-0.04
Pant L.			1.00	-0.01	-0.02	-0.15	-0.04
Sleeve L.				1.00	0.01	-0.04	0.00
Bell S.					1.00	-0.02	0.10
Shirt C.						1.00	0.07
V N.							1.00

AFED-S-Print (ReSeFa)

	Top L.	Pant L.	Sleeve L.	Bell S.	Shirt C	V N.
Top L.	1.00	0.00	0.08	0.01	-0.12	-0.12
Pant L.		1.00	-0.09	0.00	0.08	0.08
Sleeve L.			1.00	0.03	-0.02	-0.02
Bell S.				1.00	0.03	0.03
Shirt C.					1.00	1.00
V N.						1.00

AFED-H-Product (ReSeFa)

	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	-0.09	-0.09	-0.04	0.05	0.02	-0.20
Dress L.		1.00	1.00	-0.08	-0.06	-0.07	0.04
Pant L.			1.00	-0.08	-0.06	-0.07	0.04
Sleeve L.				1.00	0.00	0.00	-0.05
Bell S.					1.00	0.00	0.00
Shirt C.						1.00	-0.02
V N.							1.00

AFED-S-Color (InterFaceGAN)

	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C	V N.
Top L.	1.00	0.02	-0.01	0.00	-0.07	0.04	0.06
Dress L.		1.00	0.82	0.05	-0.03	-0.02	-0.01
Pant L.			1.00	-0.06	-0.01	0.03	0.01
Sleeve L.				1.00	0.07	-0.08	0.08
Bell S.					1.00	0.06	0.06
Shirt C.						1.00	0.00
V N.							1.00

AFED-S-Print (InterFaceGAN)

	Top L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	-0.02	-0.14	0.00	0.05	-0.16
Pant L.		1.00	0.05	0.02	0.07	0.05
Sleeve L.			1.00	0.09	-0.12	-0.06
Bell S.				1.00	-0.09	-0.01
Shirt C.					1.00	0.29
V N.						1.00

AFED-H-Product (InterFaceGAN)

	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	-0.15	0.01	0.01	0.02	0.17	0.03
Dress L.		1.00	0.22	0.01	-0.04	-0.01	-0.04
Pant L.			1.00	0.04	0.00	-0.02	-0.04
Sleeve L.				1.00	0.18	-0.07	0.03
Bell S.					1.00	0.01	-0.05
Shirt C.						1.00	-0.05
V N.							1.00

AFED-S-Color (GANSpace)

	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C	V N.
Top L.	1.00	0.00	0.00	0.00	1.00	0.00	0.00
Dress L.		1.00	0.00	0.00	0.00	0.00	0.00
Pant L.			1.00	0.00	0.00	0.00	0.00
Sleeve L.				1.00	0.00	1.00	0.00
Bell S.					1.00	0.00	0.00
Shirt C.						1.00	0.00
V N.							1.00

AFED-S-Print (GANSpace)

	Top L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	0.00	0.00	0.00	0.00	0.00
Pant L.		1.00	1.00	0.00	0.00	0.00
Sleeve L.			1.00	0.00	0.00	0.00
Bell S.				1.00	0.00	0.00
Shirt C.					1.00	0.00
V N.						1.00

AFED-H-Product (GANSpace)

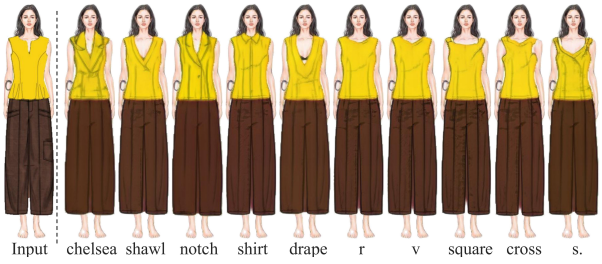
	Top L.	Dress L.	Pant L.	Sleeve L.	Bell S.	Shirt C.	V N.
Top L.	1.00	0.00	0.00	0.00	1.00	0.00	0.00
Dress L.		1.00	0.00	0.00	0.00	0.00	0.00
Pant L.			1.00	0.00	0.00	0.00	0.00
Sleeve L.				1.00	0.00	0.00	0.00
Bell S.					1.00	0.00	0.00
Shirt C.						1.00	0.00
V N.							1.00

suffers from attribute entanglement in some cases. For example, when editing sleeve length in the AFED-H-Product dataset, posture also changes. Moreover, GANSpace requires manual verification of editing directions, which increases the cost of use. InterFaceGAN has achieved good results in some attributes, but attribute entanglement remains significant. For example, when editing a top length in the AFED-S-Print dataset, the results show characteristics of a long sleeve. This further illustrates its insufficiency in disentanglement. Additionally, the process of InterFaceGAN is relatively complex, requiring an auxiliary classifier to categorize and score 500K synthetic images for training SVM semantic boundaries. This process is challenging to implement for unannotated datasets. In contrast, our method achieves comparable or even better results with only a small number of supervised data pairs, both in terms of attribute accuracy and disentanglement of editing directions.

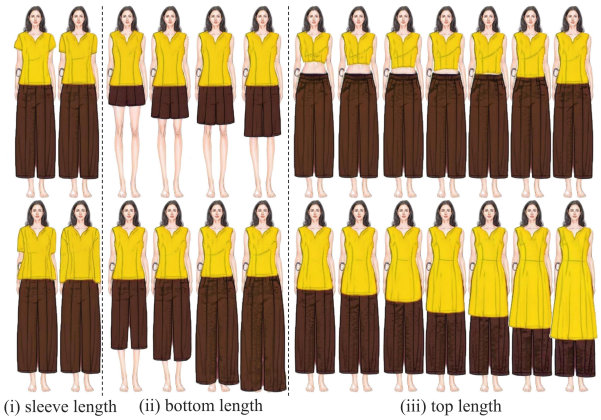
5.4 Any Fashion Attribute Editing

We present the results with fine-grained, independent, and continuous attribute editing to verify whether our approach can achieve any fashion attribute editing. Additionally, we showcase editing results on various body shapes.

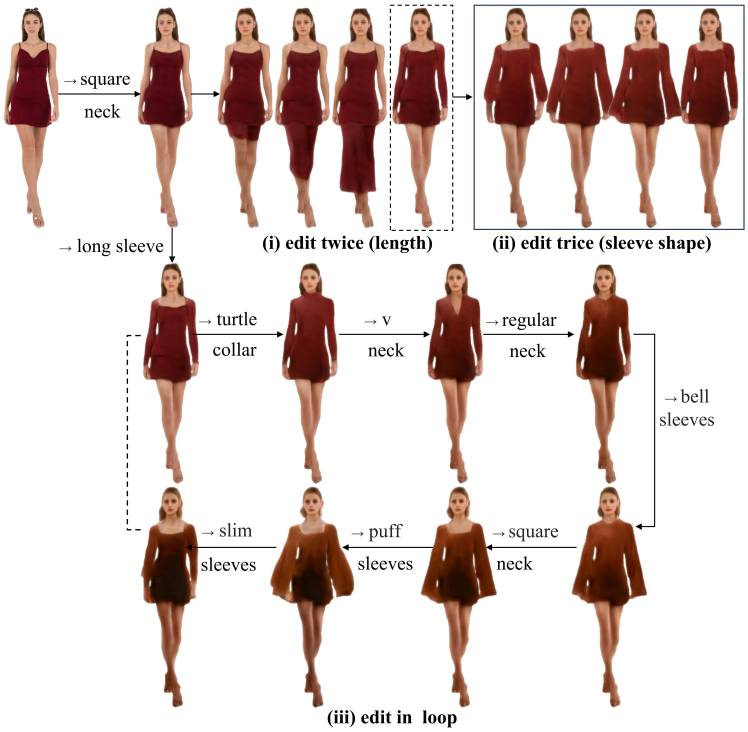
1) **Fine-grained Attribute Editing**: To demonstrate our method's superiority, additional fine-grained fashion attribute editing examples are presented in Fig. 13 (a). In the first example of Fig. 12, we successfully changed the 'Split Crew' to a 'Polo Collar'. We can also change it to other necklines such as 'Chelsea', which is a more oversized lapel than the 'Notch' (the 3rd example in Fig. 13 (a)). Another type of lapel is the 'Shawl' (the 2nd example in Fig. 13



(a) Fine-grained attributes editing on neckline.



(b) Independent attribute editing.



(c) Continuous attribute editing: edit on the edited result.

Fig. 13: More editing of our PairPCA including (a) Fine-grained attributes editing on neckline. The abbreviation s. refers to sweat-heart. (b) Independent attribute editing: change in the desired direction while keeping the rest non-deformaton. (c) Continuous attribute editing: edit on the edited result.

(a)), which has no split in the middle. In all examples in Fig. 13 (a), the bottom line of the tops and pants remains in the same horizontal line. This demonstrates the robustness of our method in changing the desired attributes while maintaining the rest unchanged.

2) **Independent Attribute Editing**: Besides fine-grained attributes, our methods also excel at changing the attribute independently. As shown in Fig. 13 (b), for sleeve length (originally 'sleeveless'), we present edited results with 'Cap', 'Short', 'Medium', and 'Long Sleeves'. Regarding bottom length (originally 'Mixi'), we provide examples of 'Mini', 'Short', 'Above-the-Knee', 'Knee', 'Cocktail', 'Midi', 'Long', and 'Floor Length'. Finally, for top length (originally 'Regular'), the 1st row shows examples of changing it to 'Short', while the 2nd row demonstrates the opposite change to 'Long'. It can be seen that as the clothing is shortened, the human body is successfully reconstructed. Notably, the pants, especially the part of the waistband, are also recovered. Conversely, as the top length increases, it naturally covers the pants without any unnatural deformation.

3) **Continuous Attribute Editing**: To show the flexibility of our method, we present continuous editing results on both sketch image and product image in Fig. 13 (c). In the left example, we begin with an initial fashion sketch of a shirt featuring a 'Split Crew Neckline' and 'Sleeveles Design'. We then sequentially edit the image to alter the collar to a shirt collar and adjust the length twice, showcasing our method's capacity for continuous multiple attribute edits. It's crucial to highlight that these edits are not merely a result of combining two latent directions in a single generation step. Instead, the process involves re-encoding the edited shirt collar image and adjusting its latent code to ensure a seamless and precise transformation. Similarly, in the second dotted box, we can further manipulate the image by changing the shape of the sleeves. This showcases the versatility of our method in performing multiple attribute edits consecutively. Furthermore, we demonstrate the power of our method by performing edits in a loop in Fig. 13 (c) (iii): iterative and progressive changes to the attributes, achieving a smooth and continuous transformation. We performed similar operations on product images, demonstrating our method's powerful ability to edit images continuously.

4) **Editing on Diverse Bodyshapes**: To show the robustness of our approach, we report the test results on different body shapes. Specifically, we categorized body shapes into four types based on the three-dimensional proportions of the human body: O (obese), X+ (slightly overweight), X- (slightly underweight), and H (slim). Then, we searched for images that meet these criteria from the AFED-H-Product dataset to showcase the editing results, as shown in Fig. 14. It is evident that, despite our attribute image pairs not specifically targeting different body shapes, our PairPCA method still achieves accurate and faithful editing effects across various body types. The conclusion remains the same for both 'common' attribute edits (such as bottom length, sleeve length, and 'V Neckline') and 'uncommon' attribute edits (such as 'Puff Sleeves', 'Irregular Neckline', and 'Square Neckline').

6 Limitation and Future Works

As shown in Fig. 12, our method significantly outperforms other editing baselines in single-edit scenarios, including

Fig. 14: Editing performance on different body shapes.

reconstructing high-frequency body attributes such as the model's face. However, it exhibits limitations during multiple consecutive edits. As Fig. 13(c) demonstrates, our method modified high-frequency body attributes when continuous editing (edit on the edited result). Future research should consider developing advanced model architectures and training strategies to mitigate such issues when continuous editing, optimizing facial areas with specialized techniques, and expanding the training dataset to include images rich in high-frequency attributes.

7 Conclusion

We focus on fashion attribute editing with the AFED dataset and Twin-Net framework. AFED comprises 830K high-quality fashion images, providing a large-scale dataset for editing tasks. Twin-Net enables high-fidelity image inversion for precise and diverse editing. Additionally, PairsPCA enhances editing by identifying semantic directions, ensuring accuracy without manual supervision. Extensive experiments confirm the effectiveness of our approach.

Acknowledgments

This work is partially supported by the Laboratory for Artificial Intelligence in Design (Project Code: RP3-1), Innovation and Technology Fund, Hong Kong, SAR. This work is also partially supported by a grant from the Research Grants Council of the Hong Kong, SAR (Project No. PolyU/RGC Project PolyU 25211424).

References

[1] R. R. Holger Harreis, Theodora Koullias and K. Te, "Generative ai: Unlocking the future of fashion," *McKinsey & Company*, 2023. 1

[2] Z. Epstein, A. Hertzmann, I. of Human Creativity, M. Akten, H. Farid, J. Fjeld, M. R. Frank, M. Groh, L. Herman, N. Leach *et al.*, "Art and the science of generative ai," *Science*, vol. 380, no. 6650, pp. 1110–1111, 2023. 1

[3] M. Jovanovic and M. Campbell, "Generative artificial intelligence: Trends and prospects," *Computer*, vol. 55, no. 10, pp. 107–112, 2022. 1

[4] M. Chui, E. Hazan, R. Roberts, A. Singla, and K. Smaje, "The economic potential of generative ai," 2023. 1

[5] M. Cao, X. Wang, Z. Qi, Y. Shan, X. Qie, and Y. Zheng, "Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22 560–22 570. 1

[6] Y. Shi, C. Xue, J. H. Liew, J. Pan, H. Yan, W. Zhang, V. Y. Tan, and S. Bai, "Dragdiffusion: Harnessing diffusion models for interactive point-based image editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8839–8849. 1, 7, 8

[7] C. Mou, X. Wang, J. Song, Y. Shan, and J. Zhang, "Dragondiffusion: Enabling drag-style manipulation on diffusion models," *arXiv preprint arXiv:2307.02421*, 2023. 1

[8] K. Zhang, M. Sun, J. Sun, B. Zhao, K. Zhang, Z. Sun, and T. Tan, "Humandiffusion: a coarse-to-fine alignment diffusion framework for controllable text-driven person image generation," *arXiv preprint arXiv:2211.06235*, 2022. 1

[9] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10 850–10 869, 2023. 1

[10] Y. Huang, J. Huang, Y. Liu, M. Yan, J. Lv, J. Liu, W. Xiong, H. Zhang, L. Cao, and S. Chen, "Diffusion model-based image editing: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 1

[11] D. Morelli, M. Fincato, M. Cornia, F. Landi, F. Cesari, and R. Cucchiara, "Dress code: High-resolution multi-category virtual try-on," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2231–2235. 1, 2, 3

[12] S. Choi, S. Park, M. Lee, and J. Choo, "Viton-hd: High-resolution virtual try-on via misalignment-aware normalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 131–14 140. 1, 2, 3

[13] N. Rostamzadeh, S. Hosseini, T. Boquet, W. Stokowiec, Y. Zhang, C. Jauvin, and C. Pal, "Fashion-gen: The generative fashion dataset and challenge," *arXiv preprint arXiv:1806.08317*, 2018. 1, 2, 3, 4

[14] J. Fu, S. Li, Y. Jiang, K.-Y. Lin, C. Qian, C. C. Loy, W. Wu, and Z. Liu, "Stylegan-human: A data-centric odyssey of human generation," in *Proceedings of the European Conference on Computer Vision*. Springer, 2022, pp. 1–19. 1, 2, 3, 4

[15] Y. Shen, C. Yang, X. Tang, and B. Zhou, "Interfacegan: Interpreting the disentangled face representation learned by gans," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 4, pp. 2004–2018, 2020. 1, 2, 3, 5, 7, 8, 11, 12

[16] Z. Wu, D. Lischinski, and E. Shechtman, "Stylespace analysis: Disentangled controls for stylegan image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 863–12 872. 1, 2, 3

[17] Y. Shen and B. Zhou, "Closed-form factorization of latent semantics in gans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1532–1540. 1, 2, 3

[18] Y. Shi, X. Yang, Y. Wan, and X. Shen, "Semanticstylegan: Learning compositional generative priors for controllable image synthesis and editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 254–11 264. 1

[19] X. Pan, A. Tewari, T. Leimkühler, L. Liu, A. Meka, and C. Theobalt, "Drag your gan: Interactive point-based manipulation on the generative image manifold," in *Proceedings of the ACM SIGGRAPH Conference*, 2023, pp. 1–11. 1, 2, 3, 7, 8, 12

[20] P. Ling, L. Chen, P. Zhang, H. Chen, Y. Jin, and J. Zheng, "Freedrag: Feature dragging for reliable point-based image editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 6860–6870. 1

[21] S. Guo, W. Huang, X. Zhang, P. Srikhanta, Y. Cui, Y. Li, H. Adam, M. R. Scott, and S. Belongie, "The imaterialist fashion attribute

dataset," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0. 1, 2

[22] X. Zou, X. Kong, W. Wong, C. Wang, Y. Liu, and Y. Cao, "Fashionai: A hierarchical dataset for fashion understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0. 1, 2, 4

[23] J. Dong, Z. Ma, X. Mao, X. Yang, Y. He, R. Hong, and S. Ji, "Fine-grained fashion similarity prediction by attribute-specific embedding learning," *IEEE Transactions on Image Processing*, vol. 30, pp. 8410–8425, 2021. 1

[24] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: a stylegan encoder for image-to-image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2287–2296. 1, 2, 3, 7, 8, 11

[25] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, "Designing an encoder for stylegan image manipulation," *ACM Transactions on Graphics*, vol. 40, no. 4, pp. 1–14, 2021. 1, 2, 3, 4, 7, 8, 11

[26] X. Yang, X. Xu, and Y. Chen, "Out-of-domain gan inversion via invertibility decomposition for photo-realistic human face manipulation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 7492–7501. 1

[27] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," *Proceedings of the Advances in Neural Information Processing Systems Conference*, vol. 33, pp. 12 104–12 114, 2020. 2

[28] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018. 2

[29] Y. Alaluf, O. Patashnik, and D. Cohen-Or, "Restyle: A residual-based stylegan encoder via iterative refinement," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6711–6720. 2, 3, 7, 8, 11

[30] H. Pehlivan, Y. Dalva, and A. Dundar, "Styleres: Transforming the residuals for real image editing with stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1828–1837. 2, 3, 4, 7, 8, 10, 11

[31] T. Wang, Y. Zhang, Y. Fan, J. Wang, and Q. Chen, "High-fidelity gan inversion for image attribute editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 379–11 388. 2, 3, 4, 7, 8, 11

[32] Y. Alaluf, O. Tov, R. Mokady, R. Gal, and A. Bermano, "Hyperstyle: Stylegan inversion with hypernetworks for real image editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 511–18 521. 2, 3

[33] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018. 2, 3

[34] Z. Chen, R. Jiang, B. Duke, H. Zhao, and P. Aarabi, "Exploring gradient-based multi-directional controls in gans," in *Proceedings of the European Conference on Computer Vision*. Springer, 2022, pp. 104–119. 2, 3

[35] Y. Dalva, S. F. Altındiş, and A. Dundar, "Vecgan: Image-to-image translation with interpretable latent directions," in *Proceedings of the European Conference on Computer Vision*. Springer, 2022, pp. 153–169. 2, 3

[36] R. Abdal, P. Zhu, N. J. Mitra, and P. Wonka, "Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows," *ACM Transactions on Graphics*, vol. 40, no. 3, pp. 1–21, 2021. 2, 3

[37] Y. Gao, F. Wei, J. Bao, S. Gu, D. Chen, F. Wen, and Z. Lian, "High-fidelity and arbitrary face editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 115–16 124. 2, 3

[38] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, "Ganspace: Discovering interpretable gan controls," *Proceedings of the Advances in Neural Information Processing Systems Conference*, vol. 33, pp. 9841–9850, 2020. 2, 3, 7, 8, 11, 12

[39] X. Hou, X. Zhang, H. Liang, L. Shen, Z. Lai, and J. Wan, "Guided-style: Attribute knowledge guided style manipulation for semantic face editing," *Neural Networks*, vol. 145, pp. 209–220, 2022. 2, 3

[40] X. Li, S. Zhang, J. Hu, L. Cao, X. Hong, X. Mao, F. Huang, Y. Wu, and R. Ji, "Image-to-image translation via hierarchical style disentanglement," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8639–8648. 2, 3

[41] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695. 2, 3, 8

[42] S. Khodadadeh, S. Ghadar, S. Motiian, W.-A. Lin, L. Bölöni, and R. Kalarot, "Latent to latent: A learned mapper for identity preserving editing of multiple face attributes in stylegan-generated images," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 3184–3192. 2, 3

[43] H. Liang, X. Hou, and L. Shen, "Ssflow: style-guided neural spline flows for face image manipulation," in *Proceedings of the ACM International Conference on Multimedia*, 2021, pp. 79–87. 2, 3

[44] P. Zhu, R. Abdal, Y. Qin, J. Femiani, and P. Wonka, "Improved stylegan embedding: Where are the good latents?" *arXiv preprint arXiv:2012.09036*, 2020. 2, 3

[45] L. Goetschalckx, A. Andonian, A. Oliva, and P. Isola, "Ganalyze: Toward visual definitions of cognitive image properties," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 5744–5753. 2, 3

[46] J. Zhu, Y. Shen, Y. Xu, D. Zhao, and Q. Chen, "Region-based semantic factorization in gans," in *International Conference on Machine Learning*. PMLR, 2022, pp. 27 612–27 632. 2, 3, 4, 7, 8, 11, 12

[47] X. Liang, S. Liu, X. Shen, J. Yang, L. Liu, J. Dong, L. Lin, and S. Yan, "Deep human parsing with active template regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 12, pp. 2402–2414, 2015. 2

[48] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1116–1124. 2

[49] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1096–1104. 2, 4

[50] K. Gong, X. Liang, D. Zhang, X. Shen, and L. Lin, "Look into person: Self-supervised structure-sensitive learning and a new benchmark for human parsing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 932–940. 2

[51] S. Zhu, R. Urtasun, S. Fidler, D. Lin, and C. Change Loy, "Be your own prada: Fashion synthesis with structural coherence," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1680–1688. 2

[52] G. Yildirim, C. Seward, and U. Bergmann, "Disentangling multiple conditional inputs in gans," *arXiv preprint arXiv:1806.07819*, 2018. 2, 3

[53] X. Han, Z. Wu, Z. Wu, R. Yu, and L. S. Davis, "Viton: An image-based virtual try-on network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7543–7552. 2, 3

[54] W. Chen, P. Huang, J. Xu, X. Guo, C. Guo, F. Sun, C. Li, A. Pfadler, H. Zhao, and B. Zhao, "Pog: Personalized outfit generation for fashion recommendation at alibaba ifashion," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2662–2670. 2, 3, 4

[55] H. Dong, X. Liang, X. Shen, B. Wang, H. Lai, J. Zhu, Z. Hu, and J. Yin, "Towards multi-pose guided virtual try-on network," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9026–9035. 2, 3, 4

[56] Y. Jiang, S. Yang, H. Qiu, W. Wu, C. C. Loy, and Z. Liu, "Text2human: Text-driven controllable human image generation," *ACM Transactions on Graphics*, vol. 41, no. 4, pp. 1–11, 2022. 2, 3, 4

[57] J. Zhu, Y. Shen, D. Zhao, and B. Zhou, "In-domain gan inversion for real image editing," in *Proceedings of the European Conference on Computer Vision*. Springer, 2020, pp. 592–608. 3

[58] Y. Xu, Y. Shen, J. Zhu, C. Yang, and B. Zhou, "Generative hierarchical features from synthesizing images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4432–4442. 3

[59] A. Creswell and A. A. Bharath, "Inverting the generator of a generative adversarial network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 7, pp. 1967–1974, 2018. 3

[60] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 172–189. 3

[61] W. Shen and R. Liu, "Learning residual images for face attribute manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4030–4038. 3

[62] T. Xiao, J. Hong, and J. Ma, "Elegant: Exchanging latent encodings with gan for transferring multiple face attributes," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 168–184. 3

[63] Y. Alaluf, O. Patashnik, and D. Cohen-Or, "Only a matter of style: Age transformation using a style-based regression model," *ACM Transactions on Graphics*, vol. 40, no. 4, pp. 1–12, 2021. 3

[64] A. Jahanian, L. Chai, and P. Isola, "On the" steerability" of generative adversarial networks," *arXiv preprint arXiv:1907.07171*, 2019. 3

[65] N. Spingarn-Eliezer, R. Banner, and T. Michaeli, "Gan" steerability" without optimization," *arXiv preprint arXiv:2012.05328*, 2020. 3

[66] A. Voynov and A. Babenko, "Unsupervised discovery of interpretable directions in the gan latent space," in *International Conference on Machine Learning*. PMLR, 2020, pp. 9786–9796. 3

[67] O. K. Yüksel, E. Simsar, E. G. Er, and P. Yanardag, "Latentclr: A contrastive learning approach for unsupervised discovery of interpretable directions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 263–14 272. 3

[68] M. Contributors, "Mmsegmentation: Openmmlab semantic segmentation toolbox and benchmark," 2020. 4

[69] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119. 4, 8

[70] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410. 4, 7, 8

[71] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proceedings of the European Conference on Computer Vision*. Springer, 2016, pp. 694–711. 6

[72] Y. Lu, Y.-W. Tai, and C.-K. Tang, "Attribute-guided face generation using conditional cyclegan," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 282–297. 6

[73] I. E. Sobel, *Camera models and machine perception*. stanford university, 1970. 6

[74] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017. 7, 8

[75] X. Hu, Q. Huang, Z. Shi, S. Li, C. Gao, L. Sun, and Q. Li, "Style transformer for image inversion and editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 337–11 346. 7, 8, 11

[76] X. Yao, A. Newson, Y. Gousseau, and P. Hellier, "A style-based gan encoder for high fidelity reconstruction of images and videos," in *Proceedings of the European Conference on Computer Vision*. Springer, 2022, pp. 581–597. 7, 8, 11

[77] G. Parmar, K. Kumar Singh, R. Zhang, Y. Li, J. Lu, and J.-Y. Zhu, "Zero-shot image-to-image translation," in *Proceedings of the ACM SIGGRAPH Conference Proceedings*, 2023, pp. 1–11. 7, 8

[78] J. Lu, X. Li, and K. Han, "Regiondrag: Fast region-based image editing with diffusion models," in *Proceedings of the European Conference on Computer Vision*, 2024. 7, 8

[79] E. Bauer and R. Kohavi, "An empirical comparison of voting classification algorithms: Bagging, boosting, and variants," *Machine Learning*, vol. 36, pp. 105–139, 1999. 7

[80] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004. 7

[81] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 586–595. 7

[82] 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," *Proceedings of the Advances in Neural Information Processing Systems Conference*, vol. 30, 2017. 7

[83] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton, "Demystifying mmd gans," *arXiv preprint arXiv:1801.01401*, 2018. 7

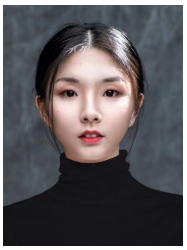
[84] G. Parmar, R. Zhang, and J.-Y. Zhu, "On aliased resizing and surprising subtleties in gan evaluation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 410–11 420. 8

[85] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. 8

[86] D. Roich, R. Mokady, A. H. Bermano, and D. Cohen-Or, "Pivotal tuning for latent-based editing of real images," *ACM Transactions on Graphics*, vol. 42, no. 1, pp. 1–13, 2022. 8

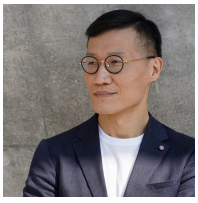
[87] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *International Conference on Learning Representations*, vol. 1, no. 2, p. 3, 2022. 8

Shumin Zhu received the B.S. degree in computer science and technology from Nanjing Agricultural University, Nanjing, China, in 2018, and the M.S. degree in pattern recognition and intelligence system from the Nanjing University of Science and Technology, Nanjing, in 2021. She is currently working toward a PhD degree at the School of Fashion and Textiles, The Hong Kong Polytechnic University, Hong Kong. Her research focuses on the application of artificial intelligence to digital fashion.



Xingxing Zou received the Ph.D. degree from The Hong Kong Polytechnic University, Kowloon, Hong Kong, in 2020. She is currently an assistant professor at The Hong Kong Polytechnic University, Hong Kong. Her major research focuses on AI Art.

Wenhan Yang (Member, IEEE) received the B.S degree and Ph.D. degree (Hons.) in computer science from Peking University, Beijing, China, in 2012 and 2018. He is currently an associate researcher with PengCheng Laboratory, Shenzhen, Guangdong, China. His current research interests include image/video processing/restoration, bad weather restoration, human-machine collaborative coding. He has authored over 50 technical articles in refereed journals and proceedings, and holds 9 granted patents. He received the 2023 IEEE Multimedia Rising Star Runner-Up Award, the IEEE ICME-2020 Best Paper Award, the IFTC 2017 Best Paper Award, the IEEE CVPR-2018 UG2 Challenge First Runner-up Award, and the MSA-TC Best Paper Award of ISCAS 2022. He was the Candidate of CSIG Best Doctoral Dissertation Award in 2019. He served as the Area Chair of IEEE ICME-2021/2022/2023/2024, the Session Chair of IEEE ICME-2021, and the Organizer of IEEE CVPR-2019/2020/2021 UG2+ Challenge and Workshop.



Wai Keung Wong received the Ph.D. degree a the Hong Kong Polytechnic University, Hong Kong SAR, in 2002. He is a full professor at The Hong Kong Polytechnic University and currently serving as the CEO & Centre Director of the Laboratory for Artificial Intelligence in Design (AiDLab). He has published over 150 scientific articles in refereed journals, including the IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Image Processing, IEEE Transactions on Cybernetics, Pattern Recognition, etc. His recent research interests include pattern recognition, feature extraction, and machine learning.