

# Controllable Person Image Synthesis with Attribute-Decomposed GAN

Yifang Men<sup>1</sup>, Yiming Mao<sup>2</sup>, Yuning Jiang<sup>2</sup>, Wei-Ying Ma<sup>2</sup>, Zhouhui Lian<sup>1\*</sup>

<sup>1</sup>Wangxuan Institute of Computer Technology, Peking University, China

<sup>2</sup>Bytedance AI Lab

## Abstract

This paper introduces the *Attribute-Decomposed GAN*, a novel generative model for controllable person image synthesis, which can produce realistic person images with desired human attributes (e.g., pose, head, upper clothes and pants) provided in various source inputs. The core idea of the proposed model is to embed human attributes into the latent space as independent codes and thus achieve flexible and continuous control of attributes via mixing and interpolation operations in explicit style representations. Specifically, a new architecture consisting of two encoding pathways with style block connections is proposed to decompose the original hard mapping into multiple more accessible subtasks. In source pathway, we further extract component layouts with an off-the-shelf human parser and feed them into a shared global texture encoder for decomposed latent codes. This strategy allows for the synthesis of more realistic output images and automatic separation of un-annotated attributes. Experimental results demonstrate the proposed method's superiority over the state of the art in pose transfer and its effectiveness in the brand-new task of component attribute transfer.

## 1. Introduction

Person image synthesis (PIS), a challenging problem in areas of Computer Vision and Computer Graphics, has huge potential applications for image editing, movie making, person re-identification (Re-ID), virtual clothes try-on and so on. An essential task of this topic is pose-guided image generation [23, 24, 9, 33], rendering the photo-realistic images of people in arbitrary poses, which has become a new hot topic in the community. Actually, not only poses but also many other valuable human attributes can be used to guide the synthesis process.

In this paper, we propose a brand-new task that aims at synthesizing person images with controllable human attributes, including pose and component attributes such as head, upper clothes and pants. As depicted in Figure 1, users are allowed to input multiple source person images

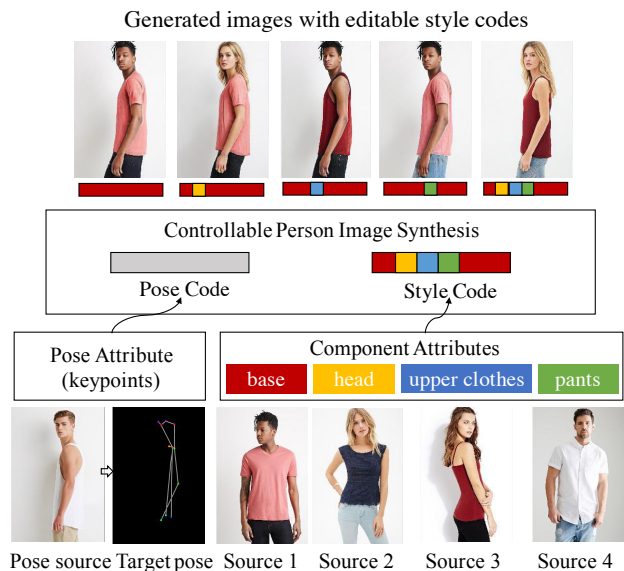


Figure 1: Controllable person image synthesis with desired human attributes provided by multiple source images. Human attributes including pose and component attributes are embedded into the latent space as the pose code and decomposed style code. Target person images can be generated in user control with the editable style code.

to provide desired human attributes respectively. The proposed model embeds component attributes into the latent space to construct the style code and encodes the keypoint-based 2D skeleton extracted from the person image as the pose code, which enables intuitive component-specific (pose) control of the synthesis by freely editing the style (pose) code. Thus, our method can automatically synthesize high-quality person images in desired component attributes under arbitrary poses and can be widely applied in not only pose transfer and Re-ID, but also garment transfer and attribute-specific data augmentation (e.g., clothes commodity retrieval and recognition).

Due to the insufficiency of annotation for human attributes, the simplicity of keypoint representation and the diversity of person appearances, it is challenging to achieve the goal mentioned above using existing methods. Pose

\*Corresponding author. E-mail: lianzhouhui@pku.edu.cn

transfer methods firstly proposed by [23] and later extended by [24, 9, 33, 46] mainly focus on pose-guided person image synthesis and they do not provide user control of human attributes such as head, pants and upper clothes. Moreover, because of the non-rigid nature of human body, it is difficult to directly transform the spatially misaligned body-parts via convolution neural networks and thus these methods are unable to produce satisfactory results. Appearance transfer methods [40, 38, 28] allow users to transfer clothes from one person to another by estimating a complicated 3D human mesh and warping the textures to fit for the body topology. Yet, these methods fail to model the intricate interplay of the inherent shape and appearance, and lead to unrealistic results with deformed textures. Another type of appearance transfer methods [30, 20, 45] try to model clothing textures by feeding the entire source person image into neural networks, but they cannot transfer human attributes from multiple source person images and lack the capability of component-level clothing editing.

The notion of attribute editing is commonly used in the field of facial attribute manipulation [14, 41, 39], but to the best of our knowledge this work is the first to achieve attribute editing in the task of person image synthesis. Different from previous facial attribute editing methods which require strict attribute annotation (e.g., smiling, beard and eyeglasses exist or not in the training dataset), the proposed method does not need any annotation of component attributes and enables automatic and unsupervised attribute separation via delicately-designed modules. In another aspect, our model is trained with only a partial observation of the person and needs to infer the unobserved body parts to synthesize images in different poses and views. It is more challenging than motion imitation methods [6, 1, 35], which utilize all characters performing a series of same motions to disentangle the appearance and pose, or train one model for each character by learning a mapping from 2D pose to one specific domain.

To address the aforementioned challenges, we propose a novel controllable person image synthesis method via an Attribute-Decomposed GAN. In contrast to previous works [23, 3, 33] forcedly learn a mapping from concatenated conditions to the target image, we introduce a new architecture of generator with two independent pathways, one for pose encoding and the other for decomposed component encoding. For the latter, our model first separates component attributes automatically from the source person image via its semantic layouts which are extracted with a pretrained human parser. Component layouts are fed into a global texture encoder with multi-branch embeddings and their latent codes are recombined in a specific order to construct the style code. Then the cascaded style blocks, acting as a connection of two pathways, inject the component attributes represented by the style code into the pose code by

controlling the affine transform parameters of AdaIN layer. Eventually, the desired image can be reconstructed from target features. In summary, our contributions are threefold:

- We propose a brand-new task that synthesizes person images with controllable human attributes by directly providing different source person images, and solve it by modeling the intricate interplay of the inherent pose and component-level attributes.
- We introduce the Attribute-Decomposed GAN, a neat and effective model achieving not only flexible and continuous user control of human attributes, but also a significant quality boost for the original PIS task.
- We tackle the challenge of insufficient annotation for human attributes by utilizing an off-the-shelf human parser to extract component layouts, making an automatic separation of component attributes.

## 2. Related Work

### 2.1. Image Synthesis

Due to their remarkable results, Generative Adversarial Networks (GANs) [13] have become powerful generative models for image synthesis [16, 44, 4] in the last few years. The image-to-image translation task was solved with conditional GANs [26] in Pix2pix [16] and extended to high-resolution level in Pix2pixHD [36]. Zhu et al. [44] introduced an unsupervised method, CycleGAN, exploiting cycle consistency to generate the image from two domains with unlabeled images. Much of the work focused on improving the quality of GAN-synthesized images by stacked architectures [43, 27], more interpretable latent representations [7] or self-attention mechanism [42]. StyleGAN [18] synthesized impressive images by proposing a brand-new generator architecture which controls generator via the adaptive instance normalization (AdaIN) [15], the outcome of style transfer literature [10, 11, 17]. However, these techniques have limited scalability in handling attributed-guided person synthesis, due to complex appearances and simple poses with only several keypoints. Our method built on GANs overcomes these challenges by a novel generator architecture designed with attribute decomposition.

### 2.2. Person Image Synthesis

Up to now, many techniques have been proposed to synthesize person images in arbitrary poses using adversarial learning. PG<sup>2</sup> [23] firstly proposed a two-stage GAN architecture to generate person images, in which the person with the target pose is coarsely synthesized in the first stage, and then refined in the second stage. Esser et al. [9] leveraged a variational autoencoder combined with the conditional U-Net [31] to model the inherent shape and appearance. Siarohin et al. [33] used a U-Net based generator with

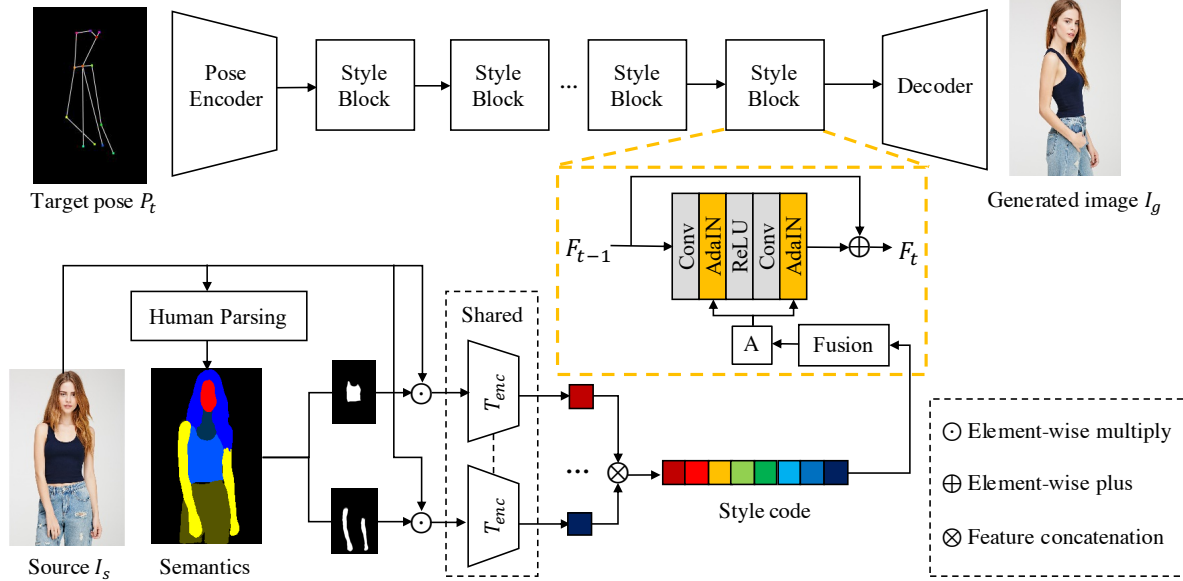


Figure 2: An overview of the network architecture of our generator. The target pose and source person are embedded into the latent space via two independent pathways, called pose encoding and decomposed component encoding, respectively. For the latter, we employ a human parser to separate component attributes and encode them via a global texture encoder. A series of style blocks equipped with a fusion module are introduced to inject the texture style of source person into the pose code by controlling the affine transform parameters in AdaIN layers. Finally, the desired image is reconstructed via a decoder.

deformable skip connections to alleviate the pixel-to-pixel misalignments caused by pose differences. A later work by Zhu et al. [46] introduced cascaded Pose-Attentional Transfer Blocks into generator to guide the deformable transfer process progressively. [29, 34] utilized a bidirectional strategy for synthesizing person images in an unsupervised manner. However, these methods only focused on transferring the pose of target image to the reference person and our method achieved a controllable person image synthesis with not only pose guided, but also component attributes (e.g., head, upper clothes and pants) controlled. Moreover, more realistic person images with textural coherence and identical consistency can be produced.

### 3. Method Description

Our goal is to synthesize high-quality person images with user-controlled human attributes, such as pose, head, upper clothes and pants. Different from previous attribute editing methods [14, 39, 41] requiring labeled data with binary annotation for each attribute, our model achieves automatic and unsupervised separation of component attributes by introducing a well-designed generator. Thus, we only need the dataset that contains person images  $\{I \in R^{3 \times H \times W}\}$  with each person in several poses. The corresponding keypoint-based pose  $P \in R^{18 \times H \times W}$  of  $I$ , 18 channel heat map that encodes the locations of 18 joints of a human body, can be automatically extracted via an existing pose estimation method [5]. During training, a target pose

$P_t$  and a source person image  $I_s$  are fed into the generator and a synthesized image  $I_g$  following the appearance of  $I_s$  but under the pose  $P_t$  will be challenged for realism by the discriminators. In the following, we will give a detailed description for each part of our model.

#### 3.1. Generator

Figure 2 shows the architecture of our generator, whose inputs are the target pose  $P_t$  and source person image  $I_s$ , and the output is the generated image  $I_g$  with source person  $I_s$  in the target pose  $P_t$ . Unlike the generator in [23] which directly concatenates the source image and target condition together as input to a U-Net architecture and forcedly learns a result under the supervision of the target image  $I_t$ , our generator embeds the target pose  $P_t$  and source person  $I_s$  into two latent codes via two independent pathways, called pose encoding and decomposed component encoding, respectively. These two pathways are connected by a series of style blocks, which inject the texture style of source person into the pose feature. Finally, the desired person image  $I_g$  is reconstructed from target features by a decoder.

##### 3.1.1 Pose encoding

In the pose pathway, the target pose  $P_t$  is embedded into the latent space as the pose code  $C_{pose}$  by a pose encoder, which consists of  $N$  down-sampling convolutional layers ( $N = 2$  in our case), following the regular configuration of encoder.

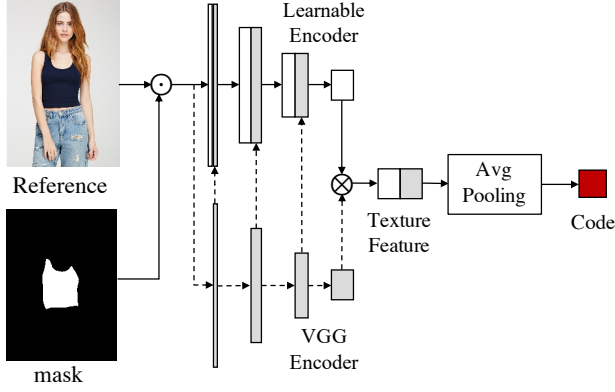


Figure 3: Details of the texture encoder in our generator. A global texture encoding is introduced by concatenating the output of learnable encoder and fixed VGG encoder.

### 3.1.2 Decomposed component encoding

In the source pathway, the source person image  $I_s$  is embedded into the latent space as the style code  $C_{sty}$  via a module called decomposed component encoding (DCE). As depicted in Figure 2, this module first extracts the semantic map  $S$  of source person  $I_s$  with an existing human parser [12] and converts  $S$  into a  $K$ -channel heat map  $M \in R^{K \times H \times W}$ . For each channel  $i$ , there is a binary mask  $M_i \in R^{H \times W}$  for the corresponding component (e.g., upper clothes). The decomposed person image with component  $i$  is computed by multiplying the source person image with the component mask  $M_i$

$$I_s^i = I_s \odot M_i, \quad (1)$$

where  $\odot$  denotes element-wise product.  $I_s^i$  is then fed into the texture encoder  $T_{enc}$  to acquire the corresponding style code  $C_{sty}^i$  in each branch by

$$C_{sty}^i = T_{enc}(I_s^i), \quad (2)$$

where the texture encode  $T_{enc}$  is shared for all branches and its detailed architecture will be described below. Then all  $C_{sty}^i, i = 1 \dots K$  will be concatenated together in a top-down manner to get the full style code  $C_{sty}$ .

In contrast to the common solution that directly encodes the entire source person image, this intuitive DCE module decomposes the source person into multiple components and recombines their latent codes to construct the full style code. Such an intuitive strategy kills two birds with one stone: 1) It speeds up the convergence of model and achieves more realistic results in less time. Due to the complex structure of the manifold that is constituted of various person images with different clothes and poses, it is hard to encode the entire person with detailed textures, but much simpler to only learn the features of one component of the person. Also, different components can share the same network parameters for color encoding and thus DCE implicitly provides a data augmentation for texture learning. The



Figure 4: Visualization effects of the DCE and GTE. (a) A source person and (b) a target pose for inputs. (c) The result generated without either DCE or GTE. (d) The result generated without only DCE. (e) The result generated with both two modules.

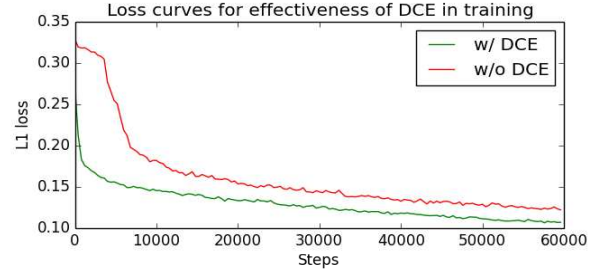


Figure 5: Loss curves for the effectiveness of our DCE module in the training process.

loss curves for the effects of our DEC module in training are shown in Figure 5 and the visualization effects are provided in Figure 4 (d)(e). 2) It achieves an automatic and unsupervised attribute separation without any annotation in the training dataset, which utilizes an existing human parser for spatial decomposition. Specific attributes are learned in the fixed positions of the style code. Thus we can easily control component attributes by mixing desired component codes extracted from different source persons.

For the texture encoder, inspired by a style transfer method [15] which directly extracts the image code via a pretrained VGG network to improve the generalization ability of texture encoding, we introduce an architecture of global texture encoding by concatenating the VGG features in corresponding layers to our original encoder, as shown in Figure 3. The values of parameters in the original encoder are learnable while those in the VGG encoder are fixed. Since the fixed VGG network is pretrained on the COCO dataset [21] and it has seen many images with various textures, it has a global property and strong generalization ability for in-the-wild textures. But unlike the typical style transfer task [15, 11] requiring only a roughly reasonable result without tight constraints, our model needs to output the explicitly specified result for a given source person in the target pose. It is difficult for the network with a fixed encoder to fit such a complex model and thus the learnable encoder is introduced, combined with the fixed one. The effects of the global texture encoding (GTE) are shown in Figure 4 (c)(d).





Figure 6: Auxiliary effects of the fusion module (FM) for DCE. (a) A source person and (b) a target pose for inputs. (c) The result generated without DCE. (d) The result generated with DCE introduced but no FM contained in style blocks. (e) The result generated with both DCE and FM.

### 3.1.3 Texture style transfer

Texture style transfer aims to inject the texture pattern of source person into the feature of target pose, acting as a connection of the pose code and style code in two pathways. This transfer network consists of several cascaded style blocks, each one of which is constructed by a fusion module and residual conv-blocks equipped with AdaIN. For the  $t^{th}$  style block, its inputs are deep features  $F_{t-1}$  at the output of the previous block and the style code  $C_{sty}$ . The output of this block can be computed by

$$F_t = \varphi_t(F_{t-1}, A) + F_{t-1}, \quad (3)$$

where  $F_{t-1}$  firstly goes through conv-blocks  $\varphi_t$ , whose output is added back to  $F_{t-1}$  to get the output  $F_t$ ,  $F_0 = C_{pose}$  in the first block and 8 style blocks are adopted totally.  $A$  denotes learned affine transform parameters (scale  $\mu$  and shift  $\sigma$ ) required in the AdaIN layer and can be used to normalize the features into the desired style [8, 15]. Those parameters are extracted from the style code  $C_{sty}$  via a fusion module (FM), which is an important auxiliary module for DEC. Because component codes are concatenated in a specified order to construct the style code, making a high correlation between the position and component features, this imposes much human ruled intervention and leads to a conflict with the learning tendency of the network itself. Thus we introduce FM consisting of 3 fully connected layers with the first two allowing the network to flexibly select the desired features via linear recombination and the last one providing parameters in the required dimensionality. FM can effectively disentangle features and avoid conflicts between forward operation and backward feedback. The effects of FM are shown in Figure 6. When DCE is applied to our model without FM, the result (see Figure 6 (d)) is even worse than that without DCE (see Figure 6 (c)). The fusion module makes our model more flexible and guarantees the proper performance of DCE.

### 3.1.4 Person image reconstruction

With the final target features  $F_{T-1}$  at the output of the last style block, the decoder generates the final image  $I_g$  from

$F_{T-1}$  via  $N$  deconvolutional layers, following the regular decoder configuration.

## 3.2. Discriminators

Following Zhu et al. [46], we adapt two discriminators  $D_p$  and  $D_t$ , where  $D_p$  is used to guarantee the alignment of the pose of generated image  $I_g$  with the target pose  $P_t$  and  $D_t$  is used to ensure the similarity of the appearance texture of  $I_g$  with the source person  $I_s$ . For  $D_p$ , the target pose  $P_t$  concatenated with the generated image  $I_g$  (real target image  $I_t$ ) is fed into  $D_p$  as a fake (real) pair. For  $D_t$ , the source person image  $I_s$  concatenated with  $I_g$  ( $I_t$ ) is fed into  $D_t$  as a fake (real) pair. Both  $D_p$  and  $D_t$  are implemented as PatchGAN and more details can be found in [16].

## 3.3. Training

Our full training loss is comprised of an adversarial term, a reconstruction term, a perceptual term and a contextual term

$$\mathcal{L}_{total} = \mathcal{L}_{adv} + \lambda_{rec}\mathcal{L}_{rec} + \lambda_{per}\mathcal{L}_{per} + \lambda_{CX}\mathcal{L}_{CX}, \quad (4)$$

where  $\lambda_{rec}$ ,  $\lambda_{per}$  and  $\lambda_{CX}$  denote the weights of corresponding losses, respectively.

**Adversarial loss.** We employ an adversarial loss  $\mathcal{L}_{adv}$  with discriminators  $D_p$  and  $D_t$  to help the generator  $G$  synthesize the target person image with visual textures similar to the reference one, as well as following the target pose. It penalizes for the distance between the distribution of real pairs ( $I_s(P_t), I_t$ ) and the distribution of fake pairs ( $I_s(P_t), I_g$ ) containing generated images

$$\begin{aligned} \mathcal{L}_{adv} = & \mathbb{E}_{I_s, P_t, I_t} [\log(D_t(I_s, I_t) \cdot D_p(P_t, I_t))] + \\ & \mathbb{E}_{I_s, P_t} [\log((1 - D_t(I_s, G(I_s, P_t))) \\ & \cdot (1 - D_p(P_t, G(I_s, P_t))))]. \end{aligned} \quad (5)$$

**Reconstruction loss.** The reconstruction loss is used to directly guide the visual appearance of the generated image similar to that of the target image  $I_t$ , which can avoid obvious color distortions and accelerate the convergence process to acquire satisfactory results.  $\mathcal{L}_{adv}$  is formulated as the L1 distance between the generated image and target image  $I_t$

$$\mathcal{L}_{rec} = ||G(I_s, P_t) - I_t||_1. \quad (6)$$

**Perceptual loss.** Except for low-level constraints in the RGB space, we also exploit deep features extracted from certain layers of the pretrained VGG network for texture matching, which has been proven to be effective in image synthesis [9, 33] tasks. Since visual style statistics can be well represented by feature correlations, we use the Gram matrix of features to measure the texture similarity. The Gram matrix  $\mathcal{G}(\mathcal{F}^l(I_t))$  for the target image  $I_t$  over its feature map  $\mathcal{F}^l(I_t)$  can be computed as

$$\mathcal{G}(\mathcal{F}^l(I_t)) = [\mathcal{F}^l(I_t)][\mathcal{F}^l(I_t)]^T, \quad (7)$$



Figure 7: Results of synthesizing person images in arbitrary poses.

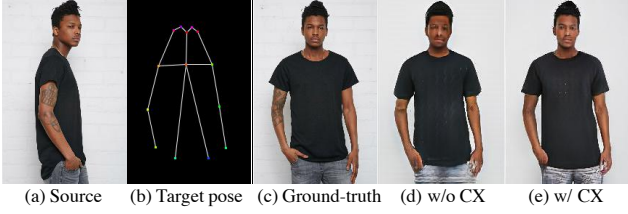


Figure 8: Effects of the contextual loss.

and the perceptual loss  $\mathcal{L}_{per}$  is defined by

$$\mathcal{L}_{per} = \| \mathcal{G}(\mathcal{F}^l(G(I_s, P_t))) - \mathcal{G}(\mathcal{F}^l(I_t)) \|^2. \quad (8)$$

**Contextual loss.** The contextual loss proposed in [25] is designed to measure the similarity between two non-aligned images for image transformation, which is also effective in our GAN-based person image synthesis task. Compared with the pixel-level loss requiring pixel-to-pixel alignment, the contextual loss allows spatial deformations with respect to the target, getting less texture distortion and more reasonable outputs. We compute the contextual loss  $\mathcal{L}_{CX}$  by

$$\mathcal{L}_{CX} = -\log(CX(\mathcal{F}^l(I_g), \mathcal{F}^l(I_t))), \quad (9)$$

where  $\mathcal{F}^l(I_g)$  and  $\mathcal{F}^l(I_t)$  denote the feature maps extracted from layer  $l = \{3, 2, 4, 2\}$  of the pretrained VGG19 network for images  $I_g$  and  $I_t$ , respectively, and  $CX$  denotes the similarity metric between matched features, considering both the semantic meaning of pixels and the context of the entire image. More details can be found in [25]. We show the effects of  $\mathcal{L}_{CX}$  in Figure 8, which enables the network to better preserve details with less distortion.

**Implementation details.** Our method is implemented in PyTorch using two NVIDIA Tesla-V100 GPUs with 16GB memory. With the human parser [2], we acquire the semantic map of person image and merge original labels defined in [12] into  $K$  ( $K = 8$ ) categories (i.e., background, hair, face, upper clothes, pants, skirt, arm and leg). The weights for the loss terms are set to  $\lambda_{rec} = 2$ ,  $\lambda_{per} = 2$ , and  $\lambda_{CX} = 0.02$ . We adopt Adam optimizer [19] with the momentum set to 0.5 to train our model for around 120k iterations. The initial learning rate is set to 0.001 and linearly decayed to 0 after 60k iterations. Following this configuration, we alternatively train the generator and two discriminators.

## 4. Experimental Results

In this section, we verify the effectiveness of the proposed network for attributes-guided person image synthesis tasks (pose transfer and component attribute transfer), and illustrate its superiority over other state-of-the-art methods. Detailed results are shown in the following subsections and more are available in the supplemental materials (Supp).

**Dataset.** We conduct experiments on the In-shop Clothes Retrieval Benchmark DeepFashion [22], which contains a large number of person images with various appearances and poses. There are totally 52,712 images with the resolution of  $256 \times 256$ . Following the same data configuration in pose transfer [46], we randomly picked 101,966 pairs of images for training and 8,750 pairs for testing.

**Evaluation Metrics.** Inception Score (IS) [32] and Structural Similarity (SSIM) [37] are two most commonly-used evaluation metrics in the person image synthesis task, which were firstly used in PG<sup>2</sup> [23]. Later, Siarohin et al. [33] introduced Detection Score (DS) to measure whether the person can be detected in the image. However, IS and DS only rely on an output image to judge the quality in itself and ignore its consistency with conditional images. Here, we introduce a new metric called contextual (CX) score, which is proposed for image transformation [25] and uses the cosine distance between deep features to measure the similarity of two non-aligned images, ignoring the spatial position of the features. CX is able to explicitly assess the texture coherence between two images and it is suitable for our task to measure the appearance consistency between the generated image and source image (target image), recording as CX-GS (CX-GT). Except for these computed metrics, we also perform the user study to assess the realness of synthesized images by human.

### 4.1. Pose transfer

#### 4.1.1 Person image synthesis in arbitrary poses

Pose is one of the most essential human attributes and our experiments verify the effectiveness of our model in pose-controlled person image synthesis. Given the same source person image and several poses extracted from person images in the test set, our model can generate natural and real-

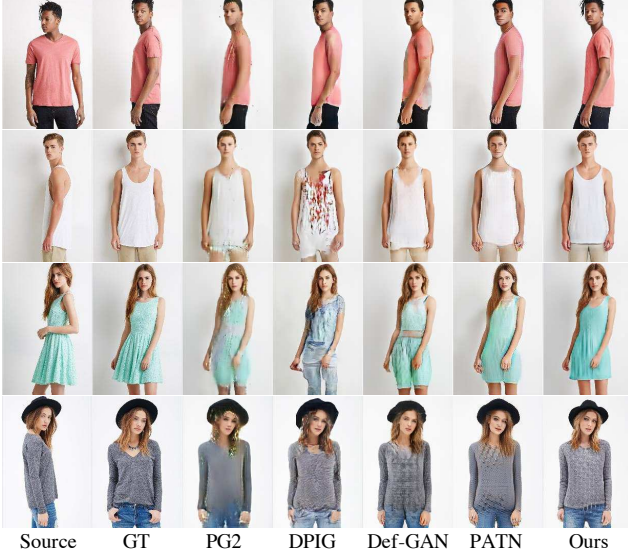


Figure 9: Qualitative comparison with state-of-the-art methods.

istic results even when the target poses are drastically different from the source in scale, viewpoints, etc. We show some results of our method in Figure 7 and more are available in Supp.

#### 4.1.2 Comparison with state-of-the-art methods

For pose transfer, we evaluate our proposed method with both qualitative and quantitative comparisons.

**Qualitative comparison.** In Figure 9, we compare the synthesis results of our method with four state-of-the-art pose transfer methods: PG<sup>2</sup> [23], DPIG [24], Def-GAN [33] and PATN [46]. All the results of these methods are obtained by directly using the source codes and trained models released by authors. As we can see, our method produced more realistic results in both global structures and detailed textures. The facial identity is better preserved and even detailed muscles and clothing wrinkles are successfully synthesized. More results can be found in Supp.

**Quantitative comparison.** In Table 1, we show the quantitative comparison with abundant metrics described before. Since the data split information in experiments of [23, 24, 33] is not given, we download their pre-trained models and evaluate their performance on our test set. Although it is inevitable that testing images may be contained in their training samples, our method still outperforms them in most metrics. The results show that our method generates not only more realistic details with the highest IS value, but also more similar and natural textures with respect to the source image and target image, respectively (lowest CX-GS and CX-GT values). Furthermore, our method has the highest confidence for person detection with the best DS value. For SSIM, we observe that when the value of IS increases,

Model	IS $\uparrow$	SSIM $\uparrow$	DS $\uparrow$	CX-GS $\downarrow$	CX-GT $\downarrow$
PG <sup>2</sup>	3.202	0.773	0.943	2.854	2.795
DPIG	3.323	0.745	0.969	2.761	2.753
Def-GAN	2.265	0.770	0.973	2.751	2.713
PATN	3.209	<b>0.774</b>	0.976	2.628	2.604
Ours	<b>3.364</b>	0.772	<b>0.984</b>	<b>2.474</b>	<b>2.474</b>

Table 1: Quantitative comparison with state-of-the-art methods on DeepFashion.

Indicator	PG <sup>2</sup>	DPIG	Def-GAN	PATN	Ours
R2G	9.2	-	12.42	19.14	<b>23.49</b>
G2R	14.9	-	24.61	31.78	<b>38.67</b>
Prefer	1.61	1.35	16.23	7.26	<b>73.55</b>

Table 2: Results of the user study (%). R2G means the percentage of real images rated as generated w.r.t. all real images. G2R means the percentage of generated images rated as real w.r.t. all generated images. The user preference of the most realistic images w.r.t. source persons is shown in the last row.

this metric slightly decreases, meaning the sharper images may have lower SSIM, which also has been observed in other methods [23, 24].

**User study.** We conduct a user study to assess the realness and faithfulness of the generated images and compare the performance of our method with four pose transfer techniques. For the realness, participants are asked to judge whether a given image is real or fake within a second. Following the protocol of [23, 33, 46], we randomly selected 55 real images and 55 generated images, first 10 of which are used for warming up and the remaining 100 images are used for evaluation. For the faithfulness, participants are shown a source image and 5 transferred outputs, and they are asked to select the most natural and reasonable image with respect to the source person image. We show 30 comparisons to each participant and finally 40 responses are collected per experiment. The results in Table 2 further validate that our generated images are more realistic, natural and faithful. It is worth noting that there is a significant quality boost of synthesis results obtained by our approach compared with other methods, where over 70% of our results are selected as the most realistic one.

#### 4.2. Component Attribute Transfer

Our method also achieves controllable person image synthesis with user-specific component attributes, which can be provided by multiple source person images. For example, given 3 source person images with different component attributes, we can automatically synthesize the target image with the basic appearance of person 1, the upper clothes of person 2 and the pants of person 3. This also provides a powerful tool for editing component-level human attributes,



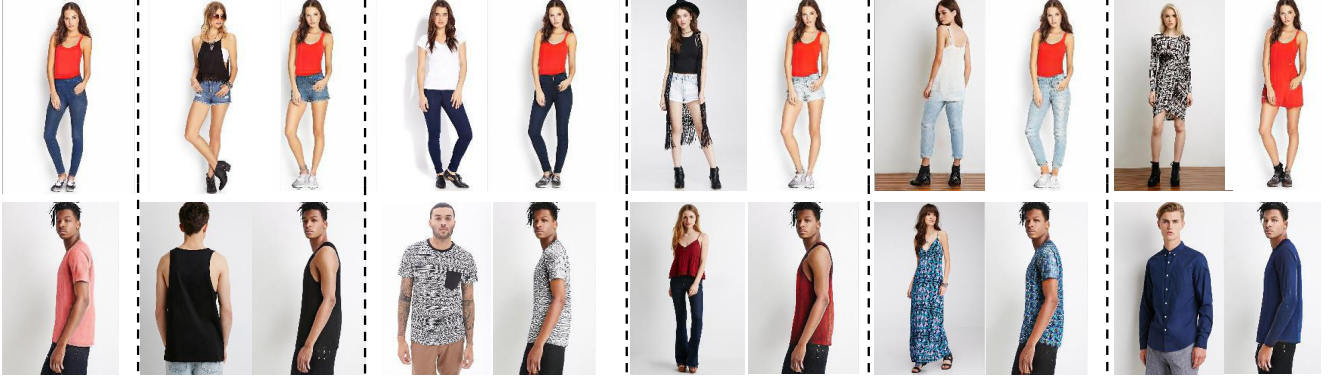


Figure 10: Results of synthesizing person images with controllable component attributes. We show original person images in the first column and the images in the right are synthesized results whose pants (the first row) or upper clothes (the second row) are changed with corresponding source images in the left.



Figure 11: Failure cases caused by component or pose attributes that extremely bias the manifold built upon training data.

such as pants to dress, T-shirt to waistcoat, and head of man to woman.

By encoding the source person images to decomposed component codes and recombining their codes to construct the full style code, our method can synthesize the target image with desired attributes. In Figure 10, we edit the upper clothes or pants of target images by using additional source person images to provide desired attributes. Our method generates natural images with new attributes introduced harmoniously while preserving the textures of remaining components.

**Style Interpolation.** Using our Attribute-Decomposed GAN, we can travel along the manifold of all component attributes of the person in a given image, thus synthesizing an animation from one attribute to another. Take for example the codes of upper clothes from person1 and person2 ( $C_{uc1}$  and  $C_{uc2}$ ), we define their mixing result as

$$C_{mix} = \beta C_{uc1} + (1 - \beta) C_{uc2}, \quad (10)$$

where  $\beta \in (0, 1)$  and  $\beta$  decreases from 1 to 0 in specific steps. Results of style interpolation are available in Supp.

### 4.3. Failure cases

Although impressive results can be obtained by our method in most cases, it fails to synthesize images with pose and component attributes that extremely bias the manifold

built upon the training data. The model constructs a complex manifold that is constituted of various pose and component attributes of person images, and we can travel along the manifold from one attribute to another. Thus, valid synthesis results are actually the mixtures of seen ones via the interpolation operation. As shown in Figure 11, the specific cartoon pattern in T-shirt of a woman fails to be interpolated with seen ones and the person in a rare pose cannot be synthesized seamlessly.

## 5. Conclusion

In this paper, we presented a novel Attribute-Decomposed GAN for controllable person image synthesis, which allows flexible and continuous control of human attributes. Our method introduces a new generator architecture which embeds the source person image into the latent space as a series of decomposed component codes and recombines these codes in a specific order to construct the full style code. Experimental results demonstrated that this decomposition strategy enables not only more realistic images for output but also flexible user control of component attributes. We also believed that our solution using the off-the-shelf human parser to automatically separate component attributes from the entire person image could inspire future researches with insufficient data annotation. Furthermore, our method is not only well suited to generate person images but also can be potentially adapted to other image synthesis tasks.

## Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant No.: 61672043 and 61672056), Beijing Nova Program of Science and Technology (Grant No.: Z191100001119077), Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology).



## References

- [1] Kfir Aberman, Rundi Wu, Dani Lischinski, Baoquan Chen, and Daniel Cohen-Or. Learning character-agnostic motion for motion retargeting in 2d. *arXiv preprint arXiv:1905.01680*, 2019. 2
- [2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 6
- [3] Guha Balakrishnan, Amy Zhao, Adrian V Dalca, Fredo Durand, and John Guttag. Synthesizing images of humans in unseen poses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8340–8348, 2018. 2
- [4] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 2
- [5] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7291–7299, 2017. 3
- [6] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5933–5942, 2019. 2
- [7] Xi Chen, Yan Duan, Rein Houthoofd, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016. 2
- [8] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016. 5
- [9] Patrick Esser, Ekaterina Sutter, and Björn Ommer. A variational u-net for conditional appearance and shape generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8857–8866, 2018. 1, 2, 5
- [10] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 262–270, 2015. 2
- [11] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 2414–2423. IEEE, 2016. 2, 4
- [12] Ke Gong, Xiaodan Liang, Dongyu Zhang, Xiaohui Shen, and Liang 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*, pages 932–940, 2017. 4, 6
- [13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2
- [14] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *IEEE Transactions on Image Processing*, 2019. 2, 3
- [15] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 2, 4, 5
- [16] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2, 5
- [17] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016. 2
- [18] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 2
- [19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6
- [20] Christoph Lassner, Gerard Pons-Moll, and Peter V Gehler. A generative model of people in clothing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 853–862, 2017. 2
- [21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 4
- [22] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1096–1104, 2016. 6
- [23] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. In *Advances in Neural Information Processing Systems*, pages 406–416, 2017. 1, 2, 3, 6, 7
- [24] Liqian Ma, Qianru Sun, Stamatis Georgioulis, Luc Van Gool, Bernt Schiele, and Mario Fritz. Disentangled person image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 99–108, 2018. 1, 2, 7
- [25] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 768–783, 2018. 6
- [26] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2
- [27] Gonçalo Mordido, Haojin Yang, and Christoph Meinel. Dropout-gan: Learning from a dynamic ensemble of discriminators. *arXiv preprint arXiv:1807.11346*, 2018. 2
- [28] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J Black. Clothcap: Seamless 4d clothing capture and retargeting. *ACM Transactions on Graphics (TOG)*, 36(4):73, 2017. 2

- [29] Albert Pumarola, Antonio Agudo, Alberto Sanfeliu, and Francesc Moreno-Noguer. Unsupervised person image synthesis in arbitrary poses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8620–8628, 2018. 3
- [30] Amit Raj, Patsorn Sangkloy, Huiwen Chang, James Hays, Duygu Ceylan, and Jingwan Lu. Swapnet: Image based garment transfer. In *European Conference on Computer Vision*, pages 679–695. Springer, 2018. 2
- [31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2
- [32] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016. 6
- [33] Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuilière, and Nicu Sebe. Deformable gans for pose-based human image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3408–3416, 2018. 1, 2, 5, 6, 7
- [34] Sijie Song, Wei Zhang, Jiaying Liu, and Tao Mei. Unsupervised person image generation with semantic parsing transformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2357–2366, 2019. 3
- [35] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018. 2
- [36] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018. 2
- [37] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6
- [38] Shan Yang, Tanya Ambert, Zherong Pan, Ke Wang, Licheng Yu, Tamara Berg, and Ming C Lin. Detailed garment recovery from a single-view image. *arXiv preprint arXiv:1608.01250*, 2016. 2
- [39] Weidong Yin, Yanwei Fu, Leonid Sigal, and Xiangyang Xue. Semi-latent gan: Learning to generate and modify facial images from attributes. *arXiv preprint arXiv:1704.02166*, 2017. 2, 3
- [40] Mihai Zanfir, Alin-Ionut Popa, Andrei Zanfir, and Cristian Sminchisescu. Human appearance transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5391–5399, 2018. 2
- [41] Gang Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Generative adversarial network with spatial attention for face attribute editing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 417–432, 2018. 2, 3
- [42] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018. 2
- [43] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xianggang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5907–5915, 2017. 2
- [44] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2
- [45] Shizhan Zhu, Raquel Urtasun, Sanja Fidler, Dahua Lin, and Chen Change Loy. Be your own prada: Fashion synthesis with structural coherence. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1680–1688, 2017. 2
- [46] Zhen Zhu, Tengting Huang, Baoguang Shi, Miao Yu, Bofei Wang, and Xiang Bai. Progressive pose attention transfer for person image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2347–2356, 2019. 2, 3, 5, 6, 7