Weakly Supervised High-Fidelity Clothing Model Generation

Ruili Feng[1], Cheng Ma[2,3], Chengji Shen[2,3], Xin Gao[3], Zhenjiang Liu[3],
Xiaobo Li[3], Kairi Ou[3], Deli Zhao[3], Zheng-Jun Zha[1*]
[1]University of Science and Technology of China, Anhui, China
[2]Zhejiang University, Hangzhou, China
[3]Alibaba Group, Hangzhou, China
ruilifengustc@gmail.com, {cheng.ma,chengji.shen}@zju.edu.cn,
{zimu.gx,stan.lzj,xiaobo.lixb}@alibaba-inc.com,
{mailokr,zhaodeli}@gmail.com, zhazj@ustc.edu.cn

Abstract

The development of online economics arouses the demand of generating images of models on product clothes, to display new clothes and promote sales. However, the expensive proprietary model images challenge the existing image virtual try-on methods in this scenario, as most of them need to be trained on considerable amounts of model images accompanied with paired clothes images. In this paper, we propose a cheap yet scalable weakly-supervised method called Deep Generative Projection (DGP) to address this specific scenario. Lying in the heart of the proposed method is to imitate the process of human predicting the wearing effect, which is an unsupervised imagination based on life experience rather than computation rules learned from supervisions. Here a pretrained StyleGAN is used to capture the practical experience of wearing. Experiments show that projecting the rough alignment of clothing and body onto the StyleGAN space can yield photo-realistic wearing results. Experiments on real scene proprietary model images demonstrate the superiority of DGP over several state-of-the-art supervised methods when generating clothing model images.

Figure 1. Given a set of commercial models with underwear and different clothing images, the proposed method can generate realistic clothing model results with clear pattern reconstruction.

1. Introduction

Taking pictures of models on product clothes is an essential demand for online apparel retails to display clothes and promote sales. However, it is highly expensive to hire models and professional studios to actually take those pictures of wearing each clothing. Thus image virtual try-on [11, 12, 15, 22, 24, 30, 42, 47, 53, 56], the technique that generates wearing results from clothing and model images, has rapidly aroused broad academic and industrial interests.

A practical algorithm, however, must consider the cost to apply it in industrial scenarios. Existing image virtual try-on (VTO) methods are highly expensive to train. Most of them [22, 28, 30, 41, 53, 56] are trained on paired image data of clothing and a person on that clothing. Such paired data consume considerable labor cost thus is infeasible to collect at scale. Also, the testing data of clothing model generation should be proprietary model images as only they can be legally used in the final display on e-shop websites. Those images are highly expensive due to hiring commercial models and buying all necessary rights. Thus algorithms should avoid relying on those proprietary model images

*Corresponding author

during training, while managing to bear the non-negligible performance drop incurred by discrepancy in testing and training.

In response to those challenges, we propose Deep Generative Projection (DGP), a powerful weakly-supervised method to yield realistic try-on results while training on cheap unpaired data collected from the Web, which is motivated by the procedure of people predicting how they will look like while picking clothes. It is an imagination based on life experience rather than rules learned from paired annotations. People may pick up the clothes and align the clothes to their shoulders or necks, from which they then imagine the picture of wearing those clothes. Following this idea, the DGP method reproduces this process in the clothing model generation scenario. Given a clothing image and a proprietary model image, we first align the clothing to the model's body dominated by a simple perspective transformation [40] of four body key points. Then we project this rough alignment onto the synthesis space of a pretrained StyleGAN [32, 33]. The StyleGAN is trained on abundant unsupervised fashion images collected from the Web. Thus, it represents real-world knowledge of wearing. A couple of semantic and pattern searches will then yield the realistic clothing model image, as is shown in Fig. 1. The whole algorithm needs no paired data or proprietary model images during training, thus it is practical for the industrial scenario.

In conclusion, the contributions of this paper include:

- We propose the first framework to generate clothing model images for online clothing shops, which has not received enough attention in the virtual try-on community;
- The proposed method consumes only unpaired data, and no proprietary model images during the training, which is more practical for industrial applications than most existing methods;
- Our weakly-supervised method significantly outperforms some state-of-the-art supervised competitors in both numerical and visual quality, and demonstrates good robustness under preprocessing mistakes.

2. Related Works

Virtual Try-on Virtual try-on methods can be broadly divided into 3D-based methods [6, 19, 43, 43, 44, 46] and 2D image-based [8, 11, 21, 22, 30, 53, 56, 58] methods. As 3D methods often induce extra resources in collecting data or physical simulation, 2D methods are generally more popular. Many existing 2D methods [8, 12, 16, 21, 22, 30] split the try-on procedure into a warping procedure and a synthesis procedure. The warping procedure learns to deform the target garment to fit the figure of the model image, while the synthesis procedure tries to merge the warped garment image with the model image. Such methodology demands paired data [22] to supervise the training of warping modules. While recent advances in GAN image synthesis also inspire 2D methods based on pretrained GANs. VOGUE [37, 38] adopts interpolation to search a latent code that can generate the target clothing in the latent space of a pretrained StyleGAN. StylePoseGAN [48] and pose with style [3] explore the rich style space of pretrained StyleGANs to manipulate the poses of synthesis images. Those works often omit the discussion of encoder or inversion technique to send the original images to StyleGANs. They also often have trouble in precisely reconstructing the pattern of clothes, and lose certain semantic information of the person or target clothing images.

StyleGAN StyleGAN [31–33] is the dominating method in unconditional image synthesis. Since it is proposed, broad interests have been attracted to use StyleGANs in various domains of image manipulations [49, 51, 55, 56, 59]. Most previous works find that the style space [23, 49, 51, 59, 60], a feature layer after the first 8-layer MLP of the StyleGAN generator, reveals fascinating semantic controlling over synthesis images. Subsequent studies [45, 55] also confirm deeper layers of StyleGAN generator owning similar or even stronger ability. As the preprocessing of manipulating real images by StyleGAN, inverting real images to the style space of StyleGAN also receives special attention. Image2StyleGAN [1, 2] finds the inverted style code through solving an optimization problem based on distance metrics. While pSp [45] and e4e [52] train explicit encoders to obtain the style code, and claim that explicit encoders can acquire more meaningful semantics for subsequent manipulations.

3. Task Setting

In this paper, the new proposed clothing model generation task is different from typical VTO scenarios. Differences are elaborated as follows.

Gaps Between Training and Testing Environments The proprietary model images are too costly to construct a sufficiently large training set. Thus algorithms should avoid relying on those images during training. However, they need to test performance on those images, as only proprietary model images can be legally used in the product display. Generally, the solution to the clothing model generation task should be stable under such a dilemma.

Original Clothes of Models Models wearing thick long sleeve clothes can perturb the image generation process. While the task here is irrelevant to the original clothes of models, we only consider cases on simple sleeveless clothes like underwear or vest.

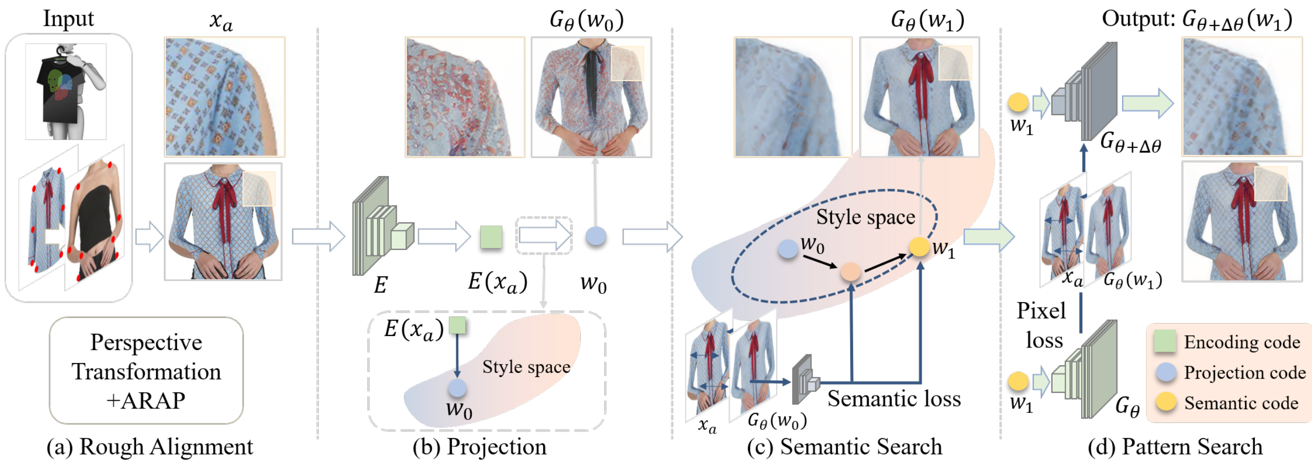

Figure 2. Framework of the proposed DGP method. A rough alignment $\boldsymbol{x}_a$ (a) of model and clothing images is fed into a novel projection operator (b), which truncates flaws of the aligned image, and transfers it into a projection code $\boldsymbol{w}_0$ that yields realistic synthesis and similar semantics on the StyleGAN synthesis space $\boldsymbol{G_\theta}$. This process is implemented by projecting the encoding code $\boldsymbol{E}(\boldsymbol{x}_a)$ of a pretrained encoder $\boldsymbol{E}$ onto the high-density region of style space. A semantic search (c) then solves a constraint optimization problem on the synthesis space of StyleGAN to find the semantic code $\boldsymbol{w}_1$ that recovers missing semantics. A pattern search (d) further adjusts parts of the StyleGAN parameters from $\boldsymbol{\theta}$ to $\boldsymbol{\theta} + \Delta\theta$. The new synthesis space $\boldsymbol{G}_{\boldsymbol{\theta}+\Delta\theta}$ then precisely reconstructs patterns of the original clothing in $\boldsymbol{G}_{\boldsymbol{\theta}+\Delta\theta}(\boldsymbol{w}_1)$. $\boldsymbol{G}_{\boldsymbol{\theta}+\Delta\theta}(\boldsymbol{w}_1)$ is the final output of the DGP method.

Benchmark The Commercial Model Image dataset (CMI) is collected to serve as a benchmark for real scene applications. The CMI dataset includes 2,348 images of models on underwear, including different genders, ages, body shapes, and poses. All model images are taken in professional studios and granted portrait rights. In addition, we also collect 1,881 clothing images with clean backgrounds from e-commerce platforms, evenly containing 16 categories, and corresponding category annotations. There are no paired relations between the clothing images and model images, and both those images are unavailable during the training phase. Please consult the supplementary materials for details of this dataset.

4. Deep Generative Projection

Overview Given a model image and a clothing image, we follow the procedure that people predict try-on results, and decompose it as a fast first impression, and a further mulling over the impression. To imitate the first impression, a novel projection operator is employed on the StyleGAN space. It projects a rough alignment of clothing and body onto the synthesis space of a pretrained StyleGAN. Different from typical literature in encoding GANs, here we do not pursue an exact reconstruction of the rough alignment, but a domain that preserves similar semantics yet maintains synthesis fidelity. The 'warping' of the proposed method is actually accomplished in this step, as realistic synthesis always yields realistic wearing. The further mulling of impression is conducted by two fine-grained information searches in the neighborhood of the encoder projection. One is the semantic search in the feature space of the StyleGAN. It recovers semantic information lost in the projection phase. The other is the pattern search in the parameter space of the StyleGAN to reconstruct the pattern of clothing. By strictly constraining these two steps in the neighborhood of the encoder projection, we can precisely reconstruct the target clothing while preserving the fidelity of the usual StyleGAN synthesis. A simple review of the proposed method is provided in Fig. 2.

Rough Alignment The rough alignment aligns the clothing and model at key points of neck, hip, elbow, and wrist, as shown in Fig. 2 (a). The alignment of the neck and hip key points is implemented by a perspective transformation [40], while the alignment of elbow and writs is implemented by the As Rigid As Possible (ARAP) [4, 27, 29] algorithm. The ARAP is a classical non-parametric deformation algorithm, which is efficient in controlling key point alignment. For different types of clothes, the alignment rule admits slight differences. For example, sleeveless clothes do not involve alignment on the elbow and wrist. If hands or arms are in front of the body, they will further be cropped and stuck on top of the aligned image to maintain consistency. See the supplementary materials for more details.

Training of StyleGAN To train the StyleGAN, we collect an E-Shop Fashion (ESF) dataset of 180,000 clothing model images from the Internet. The images are all cropped to the region between jaw and thigh, and resized to the resolution of 512×512. The whole dataset is split into 170,000 training samples and 10,000 testing samples. The StyleGAN is trained on the training dataset and the training terminates at the FID score of 2.16. More details about the StyleGAN and ESF dataset can be found in the supplementary materials.

4.1. Projection

The projection is the key to the success of the DGP method, as it offers a compact and rich domain for the subsequent mulling of details. This section gives a theoretical perspective of good projection in our task.

High-density Region of Style Space Following previous studies on StyleGAN synthesis [1, 49, 51, 60], we focus on projecting images to the style space $\mathcal{W}+$ of the StyleGAN generator $\boldsymbol{G}$. The style space $\mathcal{W}+$ is the feature layer produced by the first 8-layer MLP of the StyleGAN generator. It reveals fascinating disentanglement of semantic features [33, 49]. Different from GAN inversion [1, 2, 45, 52] techniques focusing on the exact reconstruction of input, here we care more about synthesis fidelity instead of reconstruction accuracy. This is because the exact reconstruction of rough alignment is useless for our task. As a feature space, points in the style space $\mathcal{W}+$ are not uniformly distributed. Previous works [1, 33, 49] have demonstrated that regions of higher sampling probability density can yield much more plausible synthesis than those of lower density. To strike a good balance between fidelity and similarity, the projection should always land on the high-density region of style space $\mathcal{W}+$, and give reasonable strength to each semantic component.

Thus, instead of projecting the rough alignment onto $\mathcal{W}+$ directly, we propose to project it onto each of the principal components of $\mathcal{W}+$ space. The projection on each component is further truncated if it far exceeds the average strength of $\mathcal{W}+$ space on that component. We will prove later how this operation can help anchor the projection inside the high-density region.

The Projector Rigorously, we sample five million points on the $\mathcal{W}+$ space of the StyleGAN, and compute the PCA decomposition [54] of those points. We then get the mean value $\boldsymbol{\mu}$ of $\mathcal{W}+$, covariance matrix $\boldsymbol{\Sigma}$, and a set of principal components $\boldsymbol{Q} = (\boldsymbol{q}_1, ..., \boldsymbol{q}_n)$ together with their strengths stored in $\boldsymbol{\Lambda} = diag\{\sigma_1, ..., \sigma_n\}$, where n denotes the dimension of $\mathcal{W}+$, and $\boldsymbol{\Sigma} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$. Then, instead of training an encoder $\boldsymbol{E}$ to learn the style code directly, we propose to learn a series of principal strengths $\boldsymbol{s} = (s_1, s_2, ..., s_n)^T$ of a given image, and truncate those principal strengths to reproduce the style code in an appropriate region. Given a rough alignment image $\boldsymbol{x}_a$, the style code $\boldsymbol{w}_0$ then can be computed as

$$\boldsymbol{s} = \boldsymbol{E}(\boldsymbol{x}_a), \tag{1}$$

$$\boldsymbol{w}_0 = Tr(\boldsymbol{q_1} s_1 \sqrt{\sigma_1} + ... \boldsymbol{q}_n s_n \sqrt{\sigma_n}) + \boldsymbol{\mu} \tag{2}$$

$$= \boldsymbol{Q}\boldsymbol{\Lambda}^{\frac{1}{2}} Tr(\boldsymbol{s}) + \boldsymbol{\mu}, \tag{3}$$

where Tr is a truncation operator with cutoff coefficient $\psi > 0$, such that

$$Tr(\boldsymbol{v}) = \begin{cases} \boldsymbol{v}, \|\boldsymbol{v}\|_2 < \psi, \\ \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2}\psi, \|\boldsymbol{v}\|_2 \geq \psi. \end{cases} \tag{4}$$

For simplicity, we will call the operation

$$\boldsymbol{w} = \boldsymbol{P}(\boldsymbol{x}) = Tr(\boldsymbol{Q}\boldsymbol{E}(\boldsymbol{x}_a)) + \boldsymbol{\mu} \tag{5}$$

as projection, and $\boldsymbol{P}$ the projector. Given a rough alignment image $\boldsymbol{x}_a$, the projector $\boldsymbol{P}$ is used to project it onto the synthesis space of StyleGAN as

$$\boldsymbol{w} = \boldsymbol{P}(\boldsymbol{x}_a). \tag{6}$$

Property of the Projector The projector may lose certain information represented by the small principal components after truncation, but it will enforce the projection landing on the high-density region of the style space. Rigorously, we have the following theorem:

Theorem 1 *Assume that $\mathcal{W}+$ follows the multi-variable Gaussian distribution, then the output of the projector $\boldsymbol{P}$ will always fall in the high-density region of $\mathcal{W}+$, which is an n-dimensional ellipse $\mathcal{E}$ with axes $\boldsymbol{q}_1, ..., \boldsymbol{q}_n$, and axis lengths $\psi\sigma_1^{-\frac{1}{2}}, ..., \psi\sigma_n^{-\frac{1}{2}}$. Rigorously, let ω_{n-1} denote the volume of the $n-1$ dimensional unit ball, for a random sample $\mathbf{w}$ from $\mathcal{W}+$, the possibility of it outside $\mathcal{E}$ is*

$$\mathbb{P}(\boldsymbol{w} \notin \mathcal{E}) = \mathbb{P}(\chi_n^2 > \psi^2), \tag{7}$$

where χ_n^2 is the n-dimensional Chi-square distribution [36], and $\mathbb{P}(\chi_n^2 > \psi^2)$ drops to zero drastically as ψ grows larger; and for an arbitrary input $\boldsymbol{x}$, we have

$$\boldsymbol{P}(\boldsymbol{x}) \in \mathcal{E} = \{\boldsymbol{w} : (\boldsymbol{w} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{w} - \boldsymbol{\mu}) \leq \psi^2\}. \tag{8}$$

Training of the Projector We employ a simple ResNet50 [25] architecture for the encoder network $\boldsymbol{E}$, and train the projector $\boldsymbol{P}$ on the training data distribution p_{data} of the pretrained StyleGAN $\boldsymbol{G}$. The training loss consists of a pixel similarity $\mathcal{L}_p$, a perceptual similarity $\mathcal{L}_f$, an attribute similarity $\mathcal{L}_{attr}$, and an adversarial fidelity $\mathcal{L}_{adv}$:

$$\min_{\boldsymbol{P}} \lambda_p \mathcal{L}_p + \lambda_f \mathcal{L}_f + \lambda_{attr} \mathcal{L}_{attr} + \lambda_{adv} \mathcal{L}_{adv}, \tag{9}$$

where $\lambda_p, \lambda_f, \lambda_{attr}, \lambda_{adv}$ are hyperparameters. The pixel similarity is directly captured by l_2 distance in the pixel space:

$$\mathcal{L}_p = \mathbb{E}_{\boldsymbol{x} \sim p_{data}}[\|\boldsymbol{G}(\boldsymbol{P}(\boldsymbol{x})) - \boldsymbol{x}\|_2^2]. \tag{10}$$

The perceptual similarity is captured by a pretrained VGG16 network $\boldsymbol{V}$. The VGG16 [50] network $\boldsymbol{V}$ is trained on ImageNet [9], and the final convolution layer is taken as the feature space to compute the similarity:

$$\mathcal{L}_f = \mathbb{E}_{\boldsymbol{x}\sim p_{data}}[\|\boldsymbol{V}(\boldsymbol{G}(\boldsymbol{P}(\boldsymbol{x}))) - \boldsymbol{V}(\boldsymbol{x})\|_2^2]. \tag{11}$$

The attribute similarity is captured by a pretrained clothing attribute classifier $\boldsymbol{R}$, which is trained on the FashionAI [61] dataset. It is a simple ResNet50 [25] architecture that identifies seven different attributes of clothing, like types of the sleeve and neckline. The final convolution layer is taken as the feature space to compute the similarity: (see supplementary materials for the detail of the attribute classifier)

$$\mathcal{L}_{attr} = \mathbb{E}_{\boldsymbol{x}\sim p_{data}}[\|\boldsymbol{R}(\boldsymbol{G}(\boldsymbol{E}(\boldsymbol{x}))) - \boldsymbol{R}(\boldsymbol{x})\|_2^2]. \tag{12}$$

The adversarial fidelity loss is computed through carrying on the adversarial game of pretrained StyleGAN generator $\boldsymbol{G}$ and discriminator $\boldsymbol{D}$. The discriminator $\boldsymbol{D}$ is asked to distinguish the projected images $\boldsymbol{G}(\boldsymbol{P}(\boldsymbol{x}))$ from real images, and the projector to try to fool the discriminator by projecting images to regions of higher fidelity. The generator network $\boldsymbol{G}$ is frozen during training, while $\boldsymbol{P}$ and $\boldsymbol{D}$ are alternatively optimized as in training a typical GAN [18]:

$$\begin{aligned}\mathcal{L}_{adv} = \max_{\boldsymbol{D}} \mathbb{E}_{\boldsymbol{x}\sim p_{data}} &[\log(1 - \boldsymbol{D}(\boldsymbol{G}(\boldsymbol{P}(\boldsymbol{x}))))\\ &+ \log(\boldsymbol{D}(\boldsymbol{x}))]\,.\end{aligned} \tag{13}$$

Imagination Ability of the Projector The projector can serve as a fascinating feature extractor of arbitrary image input. Though it is only trained to reconstruct real images, it can also extract semantic features from unreal images produced by splicing, scrawling, or warping. The extracted features can then be sent to the generator to reproduce those semantics in a plausible way. Fig. 3 illustrates this ability of the projector. This ability is endowed by forcing the output of the encoder to stay inside the high-density domain of the generator during training. Thus whatever inputs will be projected to a suitable domain of the generator knowledge that only produces plausible images.

Projector vs. the SOTA Encoder As we have explained, the projector is designed to encourage fidelity rather than accuracy of reconstruction. Here we compare its reconstruction with the stat-of-the-art StyleGAN encoder pSp [45]. The results are reported in Fig. 4. When handling unrealistic rough alignment results, the projector produces far more plausible results than pSp. On the other hand, pSp is faithful to the rough alignment, thus can inherit the unrealistic effects and generate images of low fidelity.

Figure 3. The imagination ability of the projector. The first row shows the inputs, each of which has some unrealistic defects. The second row shows the results of the projection. Although the encoder fails to preserve all semantics and details of the original images, it always generates plausible outputs thanks to it staying inside the high-density region of StyleGAN.

4.2. Semantic Search

The style code $\boldsymbol{w}$ discovered by the projector can only reproduce some high-level semantics (such as the style and category of the garment, pose of the model). To obtain fine-grained semantics, we need an optimization-guided search inside the neighborhood of the projection. The optimization problem is

$$\min_{\boldsymbol{w}\in\mathcal{C}} \eta_p l_p + \eta_f l_f + \eta_{attr} l_{attr} + \eta_{adv} l_{adv}, \tag{14}$$

$$l_p = \|W * \boldsymbol{G}(\boldsymbol{w}) - W * \boldsymbol{x}_a\|_2^2, \tag{15}$$

$$l_f = \|\boldsymbol{V}(W * \boldsymbol{G}(\boldsymbol{w})) - \boldsymbol{V}(W * \boldsymbol{x}_a)\|_2^2, \tag{16}$$

$$l_{attr} = \|\boldsymbol{R}(W * \boldsymbol{G}(\boldsymbol{w})) - \boldsymbol{R}(W * \boldsymbol{x}_a)\|_2^2, \tag{17}$$

$$l_{adv} = \log[1 - \boldsymbol{D}(\boldsymbol{G}(\boldsymbol{w}))], \tag{18}$$

where $\mathcal{C}$ is the neighborhood of $\boldsymbol{w}$, $\boldsymbol{R}$ and $\boldsymbol{V}$ are pretrained clothing attribute classifier and VGG16 network introduced in the Sec. 4.1 respectively, $\boldsymbol{x}_a$ is the rough alignment of clothing and model, $\eta_p, \eta_f, \eta_{attr}, \eta_{adv}$ are hyperparameters, and W is a dynamic spatial weight matrix to adjust the strength of optimization among different regions. Central regions of the body and clothing have higher weights, while marginal regions have tiny weights. This design allows the generator to adjust the marginal contents of synthesis according to central regions. We enforce the limitation of $\boldsymbol{w}$ staying in $\mathcal{C}$ to maintain the whole optimization inside the StyleGAN domain of high synthesis fidelity.

Dynamic Spatial Weight W The dynamic spatial weight matrix is an exponential function over the intersection of the model body and the aligned clothing. Let I denote the identity function for intersection of body and clothing, ∂I denote its boundary, and $d((i,j),\partial I)$ denote the distance of pixel position (i,j) to the boundary of I, then W is computed as

$$W_{ij} = \begin{cases} 1 - \exp(-d((i,j),\partial I)^2), & I(ij) = 1,\\ 0, & I(ij) = 0.\end{cases} \tag{19}$$

Figure 4. Projector vs. pSp [45]. pSp will generate a less plausible but more similar reconstruction to the input image. While the projector always generates plausible images regardless of the inputs.

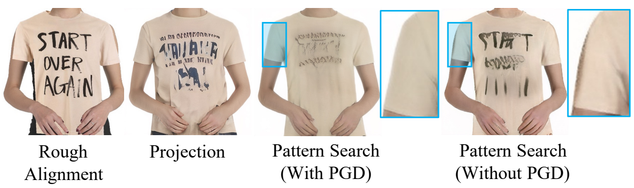

Figure 5. Influence of the Projected Gradient Descent (PGD) [39]. Without PGD, the result of pattern search will tend to overfit the original rough alignment image, which is what we do not want in this phase, as the warping of rough alignment is usually wrong.

More detailed instructions on computing W can be found in the supplementary materials.

Solve the Constrain $\mathcal{C}$ To make sure that the semantic search stays in the high-density region of StyleGAN knowledge space, we use a constraint optimization strategy that has been widely used in the field of adversarial attacks [13, 20, 39]. Given a convex optimization problem [5]

$$\min_{\boldsymbol{w}} f(\boldsymbol{w}) \ s.t. \ \boldsymbol{w} \in \mathcal{C}, \tag{20}$$

where $\mathcal{C}$ is a convex set, we can solve it by projecting the updated parameter onto $\mathcal{C}$ after each iteration of gradient descent, as shown in Algorithm 1 [39].

In the experiments, we find that the spherical neighborhood is good enough to constrain the optimization inside the high-density region of StyleGAN knowledge space. Thus we set $\mathcal{C}$ as a ball $B(\boldsymbol{w}_0, 4)$ centered at the projector output $\boldsymbol{w}_0$ with radius 4. Note that problem (22) then has a closed-form solution:

$$\boldsymbol{w}_{k+1} = \arg\min_{\boldsymbol{w}\in\mathcal{C}} \|\boldsymbol{w}_{k+1} - \boldsymbol{w}\| \tag{23}$$

$$= \begin{cases} \boldsymbol{w}_0 + 4\frac{\boldsymbol{w}_{k+1}-\boldsymbol{w}}{\|\boldsymbol{w}_{k+1}-\boldsymbol{w}\|_2}, & \|\boldsymbol{w}_{k+1} - \boldsymbol{w}_m\|_2 > 4, \\ \boldsymbol{w}_{k+1}, & \|\boldsymbol{w}_{k+1} - \boldsymbol{w}\|_2 \leq 4. \end{cases} \tag{24}$$

Problem (14) then can be solved efficiently by Algorithm 1.

Algorithm 1 Projected Gradient Descent.

Input: Hyperparameter γ, objective $f(\boldsymbol{w})$, convex constraint region $\mathcal{C}$, initial point $\boldsymbol{w}_0 \in \mathcal{C}$, counter $k = 0$.
repeat
Compute the gradient of f at $\boldsymbol{w}_k$ as $\nabla f(\boldsymbol{w}_k)$.
Update $\boldsymbol{w}_k$ by

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \gamma\nabla f(\boldsymbol{w}_k). \tag{21}$$

Project $\boldsymbol{w}_{k+1}$ back to $\mathcal{C}$ by

$$\boldsymbol{w}_{k+1} = \arg\min_{\boldsymbol{w}\in\mathcal{C}} \|\boldsymbol{w}_{k+1} - \boldsymbol{w}\|. \tag{22}$$

Update counter as $k = k + 1$.
until Convergence.
Output: The numerical solution $\boldsymbol{w}_k$ of problem (20).

Necessity of Neighborhood Constraint $\boldsymbol{w} \in \mathcal{C}$ The neighborhood constraint $\boldsymbol{w} \in \mathcal{C}$ is very important in the semantic search. Fig. 5 reports the results with and without it. Without this constraint and the Projected Gradient Descent (PGD), the optimization quickly runs out of the high-density region of the generator and produces implausible details.

4.3. Pattern Search

The pretrained generator contains rich semantic information. However, for a specific pattern like characters, we may not be able to reconstruct it accurately by semantic search. Our strategy here turns to optimizing some key parameters $\boldsymbol{\theta}$ of the generator to 'overfit' such pattern. We find that optimizing the side-way noise injection parameters [1, 2, 14, 32] in the StyleGAN Network is a good choice. Those parameters are proven to decide the local details and stochastic variations of generated images, as is carefully studied in the original paper of StyleGAN [32, 33]. As our purpose is to 'overfit' the pattern, here we do not need a feature or attribute loss to close the semantics. The optimization only involves pixel loss and adversarial loss to preserve fidelity:

$$\begin{aligned} \min_{\boldsymbol{\theta}\in B(\boldsymbol{\theta}_0,4)} \ & \eta_p \|W * \boldsymbol{G}_{\boldsymbol{\theta}}(\boldsymbol{w}) - W * \boldsymbol{x}_a\|_2 \\ & + \log(1 - \boldsymbol{D}(\boldsymbol{G}_{\boldsymbol{\theta}}(\boldsymbol{w}))), \end{aligned} \tag{25}$$

where $\boldsymbol{\theta}_0$ is the initial value of those parameters in the pretrained StyleGAN, and $B(\boldsymbol{\theta}_0, 4)$ is a ball centered at it with radius 4. We again solve the problem with Algorithm 1. The result is the final output of the DGP pipeline.

5. Experiments

In this section we evaluate the proposed *weakly-supervised* DGP method from four different aspects, and

Figure 6. Optimizing $\boldsymbol{\theta}$ alone recovers better texture details. Optimizing $\boldsymbol{w}$ alone recovers better semantic information such as the overall color and shape of clothing characteristics. Optimizing $\boldsymbol{\theta}$ and $\boldsymbol{w}$ at the same time gets the optimal results.

compare it with several *supervised* state-of-the-art competitors. Sec. 5.1 justifies the effect of the optimization components of the DGP method. Sec. 5.2 evaluates the performance on clothing model generation of the DGP method against some supervised competitors trained on paired image data. Here we focus on three state-of-the-art supervised methods, VITON-HD [8], PF-AFN [16], and ACGPN [56]. For experiments on all of those supervised methods, we use the pretrained models provided in their official repositories [7, 17, 57]. Sec. 5.3 further evaluates the performance on the MPV [11] dataset which is very similar to the training data [22] of competitor methods. It is worthwhile to mention that their original training set VITON [22] is no longer legal for academic use, thus MPV [11] may be the most pleasant dataset we can find for the competitor methods. Sec. 5.4 evaluates the robustness of the proposed method against mistakes in the preprocessing. Throughout this paper, the optimizations of both semantic and pattern searches are terminated after 1,000 steps of projected gradient descent. Supplementary materials include table recording hyper-parameter selection of training and optimization objectives.

5.1. Ablation Study

In this section we conduct extensive experiments to verify the two optimization components of the DGP method. As shown in Fig. 6, optimizing θ alone recovers better texture details, but is less effective in reconstructing semantic information. Optimizing w alone, on the contrary, recovers better semantic information such as the overall color and shape, but poorer details of patterns and letters. Optimizing θ and w at the same time yields the optimal results.

5.2. Clothing Model Generation

The real scene clothing model generation task demands algorithms to handle unseen model and clothing images from unknown distributions. To evaluate algorithms under this scenario, we conduct experiments on the CMI benchmark dataset introduced in Sec. 3, which is unavailable for all methods during training. For each model image of the CMI dataset, we randomly pick up a garment image from the

Table 1. Numerical metrics of DGP, ACGPN, PF-AFN, and VITON-HD on CMI and MPV datasets. ↓ indicates lower is better.

Methods	CMI		MPV	
	FID↓	**SWD↓**	**FID↓**	**SWD↓**
ACGPN [56]	137.9	121.3	81.1	90.4
PF-AFN [16]	97.3	76.7	67.8	67.1
VITON-HD [8]	87.5	56.1	**40.6**	52.7
DGP (Ours)	**51.6**	**22.4**	48.4	**36.7**

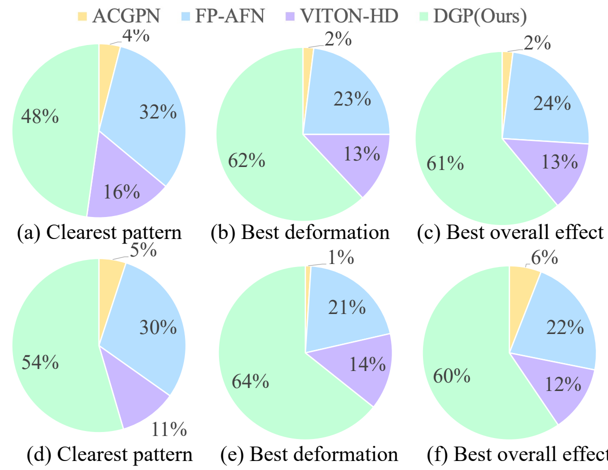

Figure 7. User study on CMI ((a), (b), (c)) and MPV ((d), (e), (f)) datasets. The proposed weakly-supervised method outperforms all supervised competitors significantly over all three aspects.

1,881 clothing images of CMI. It yields a testing set of 2,348 model and clothing pairs. All qualitative and quantitative evaluations are conducted on the 2,348 image pairs. The results are reported in Fig. 7, 8, and Tab. 1.

Qualitative Comparison Fig. 8 reports the qualitative comparison on the CMI dataset. The results reflect the advantages of the proposed method from three aspects. First, while not trained on the CMI dataset, the proposed method still works properly, with realistic synthesis. The competitors are overall less satisfactory, and work poorly in cases of complicated unseen clothes. Second, the proposed method synthesizes much clear patterns, while competitor methods often blur the patterns. Third, the proposed method can handle complicated clothing like coats, but competitor methods often fail in these cases.

Quantitative Comparison To quantitatively compare DGP with its competitors, we also measure the Fréchect Inception Distance (FID) [26] and Sliced Wasserstein Distance (SWD) [10, 34, 35] of result images. While the CMI dataset does not contain ground-truth data, here the testing set of the previously mentioned E-Shop Fashion dataset is used as the reference images. All images are cropped to

Figure 8. Comparison on the CMI and MPV datasets. The supervised competitor methods are basically less appealing, and perform especially poorly on complicated clothing like coats.

the same region and then resized to 512×512 resolution for a fair comparison. The results are reported in Tab. 1. A user study is further conducted on the visual quality of the wearing results from three aspects: 1) which method generates the clearest pattern; 2) which method generates the most realistic wearing; 3) and which method generates the best overall effect. The results are reported in Fig. 7. Both numerical metrics and user study confirm the superiority of the proposed method. Details of user study are presented in supplementary materials.

5.3. Comparison on MPV Dataset

To challenge the proposed DGP method in an unfair setting, we further compare the results of all those methods on the MPV dataset. The MPV dataset is collected from the same source as the VITON dataset, which is the training set of the competitor methods and is no longer available due to legal issues. While DGP is not trained on MPV or VITON, the superiority in this scenario can be even more attractive. We pick 1,476 image pairs of person and clothing from the MPV dataset to construct the testing set. We report qualitative comparison in Fig. 8, numerical metrics of FID and SWD in Tab. 1, and user study results in Fig. 7. Here an independent sampling of 1,476 images from MPV is used as the reference images to compute FID and SWD. The proposed method still maintains advantages in most aspects, and yields the same appealing results as in the CMI dataset.

5.4. Robustness of DGP

The imagination ability in Sec. 4.1 of the projector is very appealing for the clothing model generation. So this section further investigates how this ability can help the DGP method overcome mistakes in the preprocessing period. We deliberately feed the DGP method with flawed rough alignment images, such as missing parts of clothing, wrong key point alignments, and zigzag clothing boundaries. We then observe how the DGP method will perform with those mistakes. The results are reported in supplementary material, which confirm that DGP can easily correct these tiny mistakes, and yield realistic synthesis in the final results.

6. Conclusion

This paper studies the clothing model generation problem for online clothing retails. We propose a weakly supervised method to ease the demands of paired training data in typical virtual try-on algorithms. The proposed method casts the problem of warping clothing images to models' bodies into projecting a rough alignment of them onto the knowledge space of a pretrained StyleGAN. Extensive experiments demonstrate the superiority of our unpaired methods over several SOTA competitors trained with paired data. Future studies will focus on easing time consumption and increasing generality on extremely complicated poses of models.

Acknowledgment

This work was supported by the National Key R&D Program of China under Grant 2020AAA0105702, National Natural Science Foundation of China (NSFC) under Grant U19B2038, the University Synergy Innovation Program of Anhui Province under Grants GXXT-2019-025, and the Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2StyleGAN: How to embed images into the StyleGAN latent space? In *Int. Conf. Comput. Vis.*, pages 4432–4441, 2019. 2, 4, 6

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2StyleGAN++: How to edit the embedded images? In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8296–8305, 2020. 2, 4, 6

[3] Badour AlBahar, Jingwan Lu, Jimei Yang, Zhixin Shu, Eli Shechtman, and Jia-Bin Huang. Pose with style: Detail-preserving pose-guided image synthesis with conditional StyleGAN. *arXiv preprint arXiv:2109.06166*, 2021. 2

[4] Marc Alexa, Daniel Cohen-Or, and David Levin. As-rigid-as-possible shape interpolation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pages 157–164, 2000. 3

[5] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 6

[6] Remi Brouet, Alla Sheffer, Laurence Boissieux, and Marie-Paule Cani. Design preserving garment transfer. *ACM Trans. Graph.*, 31(4):Article–No, 2012. 2

[7] Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. VITON-HD - official pytorch implementation. https://github.com/shadow2496/VITON-HD, 2021. 7

[8] Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. VITON-HD: High-resolution virtual try-on via misalignment-aware normalization. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 14131–14140, 2021. 2, 7

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255. Ieee, 2009. 5

[10] Ishan Deshpande, Ziyu Zhang, and Alexander G Schwing. Generative modeling using the sliced Wasserstein distance. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3483–3491, 2018. 7

[11] Haoye Dong, Xiaodan Liang, Xiaohui Shen, Bochao Wang, Hanjiang Lai, Jia Zhu, Zhiting Hu, and Jian Yin. Towards multi-pose guided virtual try-on network. In *Int. Conf. Comput. Vis.*, pages 9026–9035, 2019. 1, 2, 7

[12] Haoye Dong, Xiaodan Liang, Xiaohui Shen, Bowen Wu, Bing-Cheng Chen, and Jian Yin. FW-GAN: Flow-navigated warping GAN for video virtual try-on. In *Int. Conf. Comput. Vis.*, pages 1161–1170, 2019. 1, 2

[13] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9185–9193, 2018. 6

[14] Ruili Feng, Deli Zhao, and Zheng-Jun Zha. Understanding noise injection in GANs. In *Proc. Int. Conf. Machine Learning*, pages 3284–3293. PMLR, 2021. 6

[15] Xin Gao, Zhenjiang Liu, Zunlei Feng, Chengji Shen, Kairi Ou, Haihong Tang, and Mingli Song. Shape controllable virtual try-on for underwear models. In *ACM Int. Conf. Multimedia*, pages 563–572, 2021. 1

[16] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, and Ping Luo. Parser-free virtual try-on via distilling appearance flows. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8485–8493, 2021. 2, 7

[17] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, and Ping Luo. Parser-free virtual try-on via distilling appearance flows. https://github.com/geyuying/PF-AFN, 2021. 7

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Adv. Neural Inform. Process. Syst.*, 27, 2014. 5

[19] Peng Guan, Loretta Reiss, David A Hirshberg, Alexander Weiss, and Michael J Black. Drape: Dressing any person. *ACM Trans. Graph.*, 31(4):1–10, 2012. 2

[20] Chuan Guo, Jacob Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Weinberger. Simple black-box adversarial attacks. In *Proc. Int. Conf. Machine Learning*, pages 2484–2493. PMLR, 2019. 6

[21] Xintong Han, Xiaojun Hu, Weilin Huang, and Matthew R Scott. Clothflow: A flow-based model for clothed person generation. In *Int. Conf. Comput. Vis.*, pages 10471–10480, 2019. 2

[22] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. VITON: An image-based virtual try-on network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7543–7552, 2018. 1, 2, 7

[23] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. GANspace: Discovering interpretable GAN controls. *Adv. Neural Inform. Process. Syst.*, 33:9841–9850, 2020. 2

[24] Stefan Hauswiesner, Matthias Straka, and Gerhard Reitmayr. Virtual try-on through image-based rendering. *IEEE Trans. Vis. Comput. Graph.*, 19(9):1552–1565, 2013. 1

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016. 4, 5

[26] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. *Adv. Neural Inform. Process. Syst.* 7

[27] Takeo Igarashi, Tomer Moscovich, and John F Hughes. As-rigid-as-possible shape manipulation. *ACM Trans. Graph.*, 24(3):1134–1141, 2005. 3

[28] Thibaut Issenhuth, Jérémie Mary, and Clément Calauzenes. Do not mask what you do not need to mask: a parser-free virtual try-on. In *Eur. Conf. Comput. Vis.*, pages 619–635. Springer, 2020. 1

[29] Alec Jacobson, Ilya Baran, Jovan Popovic, and Olga Sorkine. Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph.*, 30(4):78, 2011. 3

[30] Hyug Jae Lee, Rokkyu Lee, Minseok Kang, Myounghoon Cho, and Gunhan Park. LA-VITON: a network for looking-attractive virtual try-on. In *Int. Conf. Comput. Vis.*, pages 0–0, 2019. 1, 2

[31] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *Int. Conf. Learn. Represent.*, 2018. 2

[32] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4401–4410, 2019. 2, 6

[33] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8110–8119, 2020. 2, 4, 6

[34] Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo K Rohde. Generalized sliced Wasserstein distances. *arXiv preprint arXiv:1902.00434*, 2019. 7

[35] Soheil Kolouri, Gustavo K Rohde, and Heiko Hoffmann. Sliced Wasserstein distance for learning Gaussian mixture models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3427–3436, 2018. 7

[36] Henry Oliver Lancaster and Eugene Seneta. Chi-square distribution. *Encyclopedia of Biostatistics*, 2, 2005. 4

[37] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. TryOnGAN: body-aware try-on via layered interpolation. *ACM Trans. Graph.*, 40(4):1–10, 2021. 2

[38] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. VOGUE: Try-on by StyleGAN interpolation optimization. *arXiv preprint arXiv:2101.02285*, 2021. 2

[39] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 6

[40] Jack Mezirow. Perspective transformation. *Adult education*, 28(2):100–110, 1978. 2, 3

[41] Matiur Rahman Minar, Thai Thanh Tuan, Heejune Ahn, Paul Rosin, and Yu-Kun Lai. CP-VTON+: Clothing shape and texture preserving image-based virtual try-on. In *IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, 2020. 1

[42] Assaf Neuberger, Eran Borenstein, Bar Hilleli, Eduard Oks, and Sharon Alpert. Image based virtual try-on network from unpaired data. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5184–5193, 2020. 1

[43] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. Tailornet: Predicting clothing in 3D as a function of human pose, shape and garment style. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7365–7375, 2020. 2

[44] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J Black. Clothcap: Seamless 4D clothing capture and retargeting. *ACM Trans. Graph.*, 36(4):1–15, 2017. 2

[45] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a StyleGAN encoder for image-to-image translation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2287–2296, 2021. 2, 4, 5, 6

[46] Damien Rohmer, Tiberiu Popa, Marie-Paule Cani, Stefanie Hahmann, and Alla Sheffer. Animation wrinkling: augmenting coarse cloth simulations with realistic-looking wrinkles. *ACM Trans. Graph.*, 29(6):1–8, 2010. 2

[47] Igor Santesteban, Miguel A Otaduy, and Dan Casas. Learning-based animation of clothing for virtual try-on. In *Computer Graphics Forum*, volume 38, pages 355–366. Wiley Online Library, 2019. 1

[48] Kripasindhu Sarkar, Vladislav Golyanik, Lingjie Liu, and Christian Theobalt. Style and pose control for image synthesis of humans from a single monocular view. *arXiv preprint arXiv:2102.11263*, 2021. 2

[49] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of GANs for semantic face editing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9243–9252, 2020. 2, 4

[50] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[51] Ayush Tewari, Mohamed Elgharib, Gaurav Bharaj, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt. Stylerig: Rigging StyleGAN for 3D control over portrait images. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6142–6151, 2020. 2, 4

[52] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for StyleGAN image manipulation. *ACM Trans. Graph.*, 40(4):1–14, 2021. 2, 4

[53] Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, Liang Lin, and Meng Yang. Toward characteristic-preserving image-based virtual try-on network. In *Eur. Conf. Comput. Vis.*, pages 589–604, 2018. 1, 2

[54] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987. 4

[55] Zongze Wu, Dani Lischinski, and Eli Shechtman. Stylespace analysis: Disentangled controls for StyleGAN image generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12863–12872, 2021. 2

[56] Han Yang, Ruimao Zhang, Xiaobao Guo, Wei Liu, Wangmeng Zuo, and Ping Luo. Towards photo-realistic virtual try-on by adaptively generating-preserving image content. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7850–7859, 2020. 1, 2, 7

[57] Han Yang, Ruimao Zhang, Xiaobao Guo, Wei Liu, Wangmeng Zuo, and Ping Luo. Towards photo-realistic virtual try-on by adaptively generating preserving image content. `https://github.com/minar09/ACGPN`, 2020. 7

[58] Ruiyun Yu, Xiaoqi Wang, and Xiaohui Xie. VTNFP: An image-based virtual try-on network with body and clothing feature preservation. In *Int. Conf. Comput. Vis.*, pages 10511–10520, 2019. 2

[59] Heliang Zheng, Jianlong Fu, Yanhong Zeng, Jiebo Luo, and Zheng-Jun Zha. Learning semantic-aware normalization for generative adversarial networks. *Adv. Neural Inform. Process. Syst.*, 33:21853–21864, 2020. 2

[60] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain GAN inversion for real image editing. In *Eur. Conf. Comput. Vis.*, pages 592–608. Springer, 2020. 2, 4

[61] Xingxing Zou, Xiangheng Kong, Waikeung Wong, Congde Wang, Yuguang Liu, and Yang Cao. FashionAI: A hierarchical dataset for fashion understanding. In *IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, pages 0–0, 2019. 5