Dress-On online model base on Deep Learning and Computer Vision

by

Ngo Quoc Huy

Vo Minh Bao

THE FPT UNIVERSITY HO CHI MINH CITY

Dress-On online model base on Deep Learning and Computer Vision

by

Ngo Quoc Huy

Vo Minh Bao

Supervisor: Mr. Nguyen Quoc Trung
Mr. Ngo Dang Ha An

A final year capstone project submitted in partial fulfillment of the requirement for the Degree of Bachelor of Artificial Intelligent in Computer Science

DEPARTMENT OF ITS

THE FPT UNIVERSITY HO CHI MINH CITY

APRIL 2024

ACKNOWLEDGMENTS

Our sincere gratitude goes to our supervisors, Mr. Nguyen Quoc Trung and Mr. Ngo Dang Ha An, for their invaluable support, extensive knowledge, and unwavering patience throughout our thesis journey. His expert guidance and constructive feedback were instrumental in the successful completion of this project. We also express our appreciation to the Review Committee for their meticulous review, ensuring the adherence to high standards. Mr. Nguyen Quoc Trung and Mr. Ngo Dang Ha An's exceptional mentorship has been a cornerstone of our academic success, and we thank him for his dedication. Our heartfelt thanks extend to all who contributed, both in significant and subtle ways, as their support was crucial to the realization of this project.

AUTHOR CONTRIBUTIONS

The section is a short-written description about the students. The paragraph also provides their specifying individual contributions. The following statements should be used "Conceptualization, Ngo Quoc Huy; methodology, Ngo Quoc Huy; software, Vo Minh Bao; validation, Vo Minh Bao; formal analysis, Vo Minh Bao; investigation, Ngo Quoc Huy and Vo Minh Bao; resources, Vo Minh Bao; data curation, Vo Minh Bao; writing—original draft preparation, Ngo Quoc Huy; writing—review and editing, Vo Minh Bao; visualization, Vo Minh Bao; supervision, X.X.; project administration, Ngo Quoc Huy. All authors have read and agreed to the Final Capstone Project document."

ABSTRACT

Image-based virtual try-on seeks to fit target clothing into an image of a person. Especially, garment warping is a crucial step for aligning the desired garment with the body portions in the image. The previous model [(2)] typically applies a local appearance flow method for warping modules. The results of them are mostly harmful to difficult body poses and large misalignments between person and garment images. To minimize this limitation, we proposed a novel global appearance flow estimation model. This allows us to overcome the aforementioned difficulties by utilizing a global style vector to convey a whole-image context. Also, a flow refinement module is presented to add local context so that the StyleGAN flow generator can pay more attention to local garment deformation.

Keywords: Deep Learning (DL), Computer Vision (CV), optical flow

Contents

List of Figures

List of Tables

1 INTRODUCTION

The increasing importance of online shopping in the dynamic world of digital commerce has spurred technology advancements targeted at improving the client experience in its entirety. In the fashion industry, online customers sometimes miss out on the offline experience of trying on clothes in a changing room. Image-based virtual try-on, or VTON, has been the subject of extensive research lately to lower return costs for online retailers and provide customers with an offline experience when shopping online.

The work of VTON models is trying to fit an indicated garment into a person's image. Aligning the in-store clothing with the correct body components in the person's image is one of the VTON model's main goals. This is because the in-store clothing is typically not positioned about the person's image. To fuse the texture in the person and garment images, sophisticated detail-preserving images must first be applied to image translation models. This will result in an unnatural effect in the generated try-on image, especially in the occluded and misaligned regions.

Previous techniques use cloth warping to deal with this alignment problem. While some models define a single-stage approach for convenience, however, multi-stage models seem more flexible to handle this virtual try-on task which mainly contains: warping and generating modules.

In recent researches, for the warping module, many of them [(1), (8), (13), (16), (17), (18)] apply a Thin Plate Spline (TPS) [(3)], by using the correlation between the features derived from the person and the garment images. Previous works [(5), (7), (17)] show the limitation of handling complex warping by using TPS, e.g., when different areas of the garment require distinguished deformations. However, recently, many SOTA methods [(2), (7)] have manipulated the training network to predict the dense appearance flow [(4)] which represents the deformation required to align the body pose with the cloth.

Despite the appearance flow methods outperform the TPS methods, these existing methods still lack global context which is limited in accurate cloth warping. Current approaches for estimating optical flow [(6), (9)] rely on local feature correspondence, such as concatenation or correlation. They assume that the relevant regions from the person image and the in-store garment are located in the same local receptive field of the feature extractor to estimate the appearance flow. Current appearance flow-based approaches will significantly worsen and produce unacceptable outcomes when there is a significant misalignment between the garment and corresponding body components. When correspondences need to be explored outside of a local neighborhood, the flow-based VTON approaches that are currently in use are similarly susceptible to challenging poses or occlusions due to their lack of a global context.

This paper proposes a new methodology for estimating global appearance flows to address this issue. We offer the initial StyleGAN [(14), (15)] architecture for estimating dense appearance flows. This is substantially different from prior approaches [(2), (6), (7), (9)], which use a U-Net [(19)] design to maintain local spatial context. Our approach accurately captures global context by extracting a global style vector from reference and garment photos. Unfortunately, a single style vector looks to have lost its local spatial context which is required for local alignment. StyleGAN has been successfully applied to local face image manipulation tasks, where multiple style vectors can generate the same face at different views [(10)] and forms [(12), (24)]. This shows that a global style vector encodes local geographical context. The vanilla StyleGAN architecture [(14), (15)] is more resistant to large misalignment and challenging poses/occlusions than U-Net but has a lower performance in local deformation modeling. Therefore, we added a local flow refining module to the current StyleGAN generator to achieve the best of both worlds.

Our StyleGAN-based warping module (W in Fig 1) uses stacked warping blocks with inputs including a global style vector, clothing features, and person attributes. To simulate the global context modeling, the global style vector is created by combining the lowest-resolution feature maps of the individual image and the in-shop garment. The global style vector modulates feature channels in each warping block of the generator, estimating the appearance flow based on the associated garment feature map. To represent fine-grained local appearance flow, such as the

arm and hand regions (Fig 1), we add a refinement layer to each warping block above the style-based appearance flow calculation. The refinement layer warps the garment feature map, concatenates it with an individual feature map at the same resolution, and predicts the local detailed appearance flow.

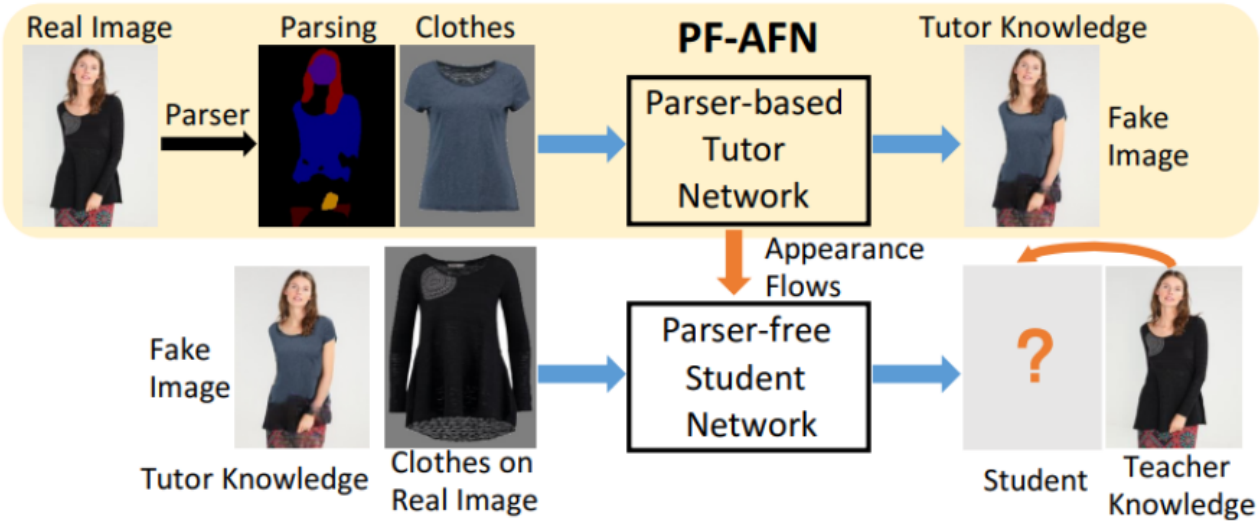

Figure 1: The PF-AFN handles the image of the fake person as "tutor knowledge" and feeds it into the parser-free "student" network which is supervised by the image of the real person (teacher expertise). The "tutor" network, which is parser-based, helps the "student" network generate high-quality images by further refining the appearance flows between the human and clothing images.

Our workflow is based on the Parser Free Appearance Flow Network (PF-AFN), which is known as the "teacher-tutor-student" knowledge distillation scheme for the try-on problem (described in Fig 1). This method produces lifelike results without relying on human segmentation or parsing. To put it simply, PF-AFN sees it as a "tutor" network, which may yield unrealistic results (e.g., tutor knowledge) and require improvement by a real teacher. The idea is to design where the teacher's knowledge originates from. PF-AFN uses the false person image (tutor knowledge) as input for the parser-free student model, which is then supervised by the original real person image (teacher knowledge), resulting in the student mimicking the original actual images. Similar to self-supervised learning, the student network is taught by transferring the garment from a real person image to a fake person image created by a parser-based model. In other words, the student is requested to change the garments on the false person image to those on the real person image, allowing them to be self-supervised by the genuine person image, which is free of artifacts.

The contributions of this work are the following: (1) We present a style-based appearance flow method to warp garments during virtual try-ons. Our VTON model is more resilient to substantial misalignments between human and clothing photos thanks to the global flow estimate approach. (2) Extensive experiments verify our strategy as superior to current state-of-the-art alternatives.

2 RELATED WORK

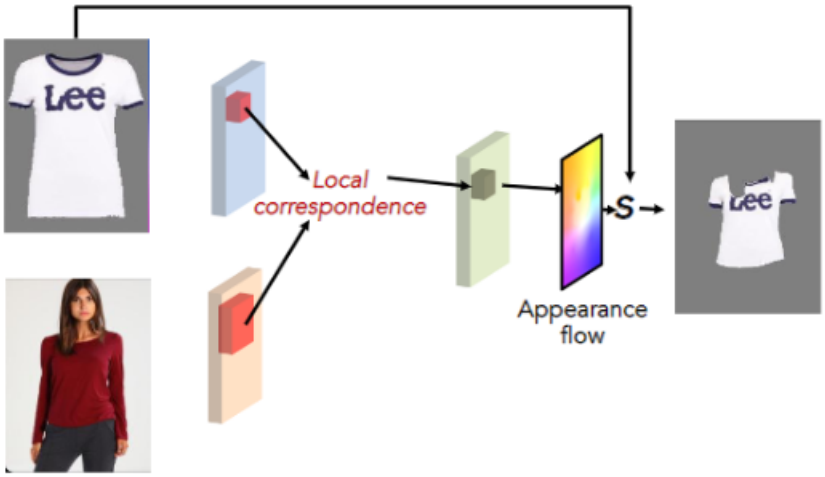

Figure 2: Appearance flow is used in garment warping in existing methods [(2)].

Virtual try-on There are two types of image-based (2D) VTON: parser-based and parser-free approaches. Their primary distinction is whether or not the inference stage calls for an off-the-shelf human parser.

To estimate the warping parameter, parser-based approaches mask the garment region in the input person image using a human segmentation map. The warped garment and the masked person image are concatenated, and the resulting data is sent into a generator to create the target try-on image. Additionally, [(17)] modifies the segmentation map to match the target garment for improved try-on image production. The final try-on image is generated using the altered parsing result, the masked human image, and the twisted garment. Because these techniques rely on a parser, they are susceptible to poor human parsing performance [(2), (13)], which invariably results in erroneous warping and try-on results.

Parser-free approaches [(2), (13)], on the other hand, only accept the human picture and the garment image as inputs during the inference stage. They are made expressly to remove the detrimental impacts caused by incorrect parsing outcomes. These techniques typically train

a parser-based teacher model first, and a parser-free student model second. [(13)] suggested a pipeline that uses paired triplets to condense the try-on generation network and garment warping module. Through the introduction of cycle consistency for improved distillation, [(2)] further enhanced [(13)].

We also introduce a parser-free technique. However, the main focus of our approach is the garment warping part's design, where we suggest a brand-new global appearance flow-based garment warping module.

3D virtual try-on Deep learning algorithms for virtual try-on can be classed as 3D model-based [(11), (23)] and 2D image-based [(7), (8), (13), (16), (17)]. While 3D VTON is more difficult than image-based VTON, it offers a better try-on experience (enabling being viewed with arbitrary viewpoints and positions, for example). The cost and effort involved in gathering large-scale 3D information limit the scalability of a 3D VTON model. In addition, 2D image-based approaches, which require less computational power and fewer measurements, are more widely applicable than 3D approaches. In comparison to 2D techniques, current 3D VTON still produces less detailed texture information.

Previous methods [(7), (8), (16), (17), (18)] mostly mask the clothing region of the person image and reconstruct the person image with the corresponding clothes image, which calls for accurate human parsing. This is because available datasets [(8)] for 2D image try-ons only contain unpaired data (clothes and a person wearing the clothes).

Manipulating image by using StyleGAN The study of image manipulation [(22), (24)] has recently undergone a revolution thanks to StyleGAN [(14), (15)]. Its suitability for learning a highly disentangled latent space is typically attributed to its effective application on picture editing tasks. Unsupervised latent semantics discovery has been the subject of recent work [(10), (21)]. [(25)] used virtual try-on with pose-conditioned StyleGAN. Nevertheless, their model is slow during inference and is unable to keep clothing details.

Our garment warping network's architecture draws inspiration from StyleGAN's exceptional shape deformation performance in image modification [(10), (24)]. We employ style modulation

to predict the implicit appearance flow, which is then used to warp the garment via sampling, as opposed to using style modulation to generate the warped garment. Compared to [(25)], this style is far better suitable for maintaining the details of the garment.

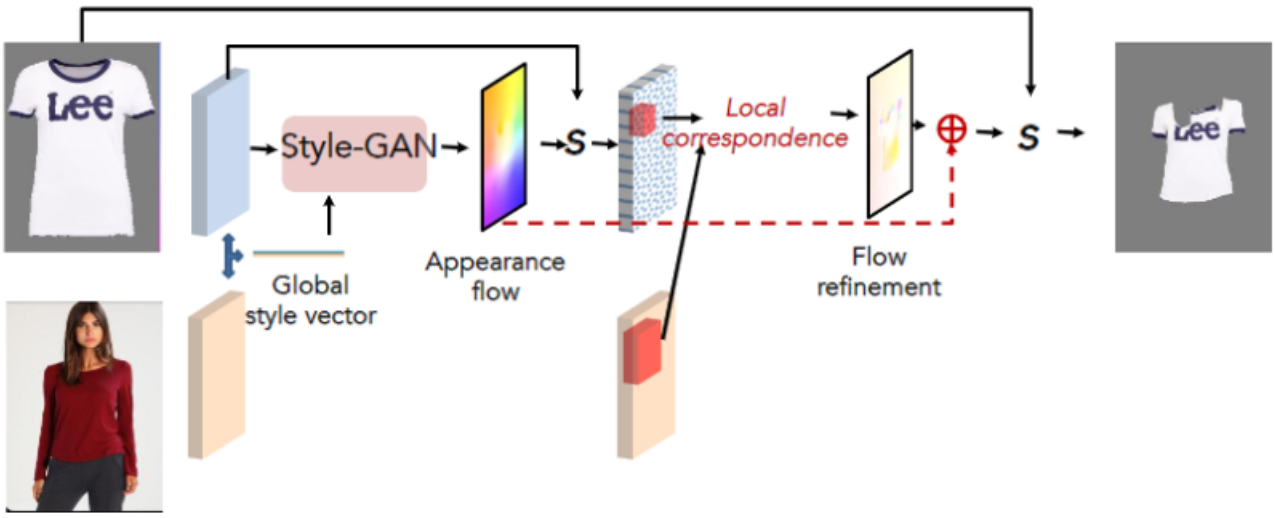

Figure 3: Style-based appearance flow estimation method is used in this work.

Appearance flow [(7)] revealed appearance flow for the first time in the context of VTON. Since then, it has drawn further interest and been included in more modern, cutting-edge VTON models [(2), (5)]. Appearance flow, which is superior in keeping detail and functions as a sample grid for garment warping, is essentially how it works. In addition to VTON, appearance flow is widely used in other jobs. [(4)] used it for the synthesis of a novel viewpoint. The concept of appearance flow was also used by [(26), (27)] to warp the feature map for human pose transfer. Unlike all the other appearance flow estimating techniques currently in use, our approach uses style modulation to estimate the appearance flow by applying a global style vector (see Fig 3). Therefore, our approach is inherently better at handling significant misalignments.

3 PROJECT MANAGEMENT PLAN

Table 1: *Project Management Plan.*

Week	**Task name**	**Owner**	**Note**
1	- Start researching the topic. - Collecting papers and projects related to the topic - Researching which try-on clothes models have the best accuracy.	All 2 members	
2	- Choose the best models. - Learning the overall architecture of each model. - Try running models with the checkpoint pretrained provided.	All 2 members	
3	- Continue testing models - Find the limitations of the models. - List the 3 most potential virtual fitting models. - Compare the accuracy of 3 models based on the available dataset.	All 2 members	
4	- Choose the PF-AFN model as the original model to improve. - Learn the overall architecture of the PF-AFN model. - Rerun all PF-AFN data and evaluate it compared to the results stated in the paper.	All 2 members	

5	- Learn other methods to improve of the PF-AFN model. - Proceed to train the PF-AFN model.	All 2 members	
6	- Research methods for preprocessing data of the PF-AFN model. - Research frameworks and libraries the author used for the PF-AFN model.	All 2 members	
7	- Find the limitations of the PF-AFN model. - Researching and finding solutions for each of those limitations. - Determine the limits of the PF-AFN model.	All 2 members	
8	- Research and understand the mathematical formulas the author uses to process the PF-AFN model. - Research and learn the metrics used to evaluate accuracy.	Huy Bao	
9	- Research and implement methods for optimizing the PF-AFN model. - Identify and collect additional necessary datasets for model training. - Meeting with the SE team to plan for deploying AI to the web.	All 2 members	
10	- Retrain the model after applying different optimization methods. - Test the model based on a trained weight set. - Write an API to deploy the AI model on the web.	Huy Bao Bao	

11	- Train again on a full steps model after optimization. - Test the model based on the set of weights after training. - Start writing reports.	Huy Bao Huy	
12	- Continue writing reports.	All 2 members	
13	- Edit and complete web demo. - Edit and complete report.	All 2 members	
14	- Review knowledge and complete defense presentation slides. - Present the topic to the council.	All 2 members	

4 MATERIALS AND METHODS

4.1 Methodology

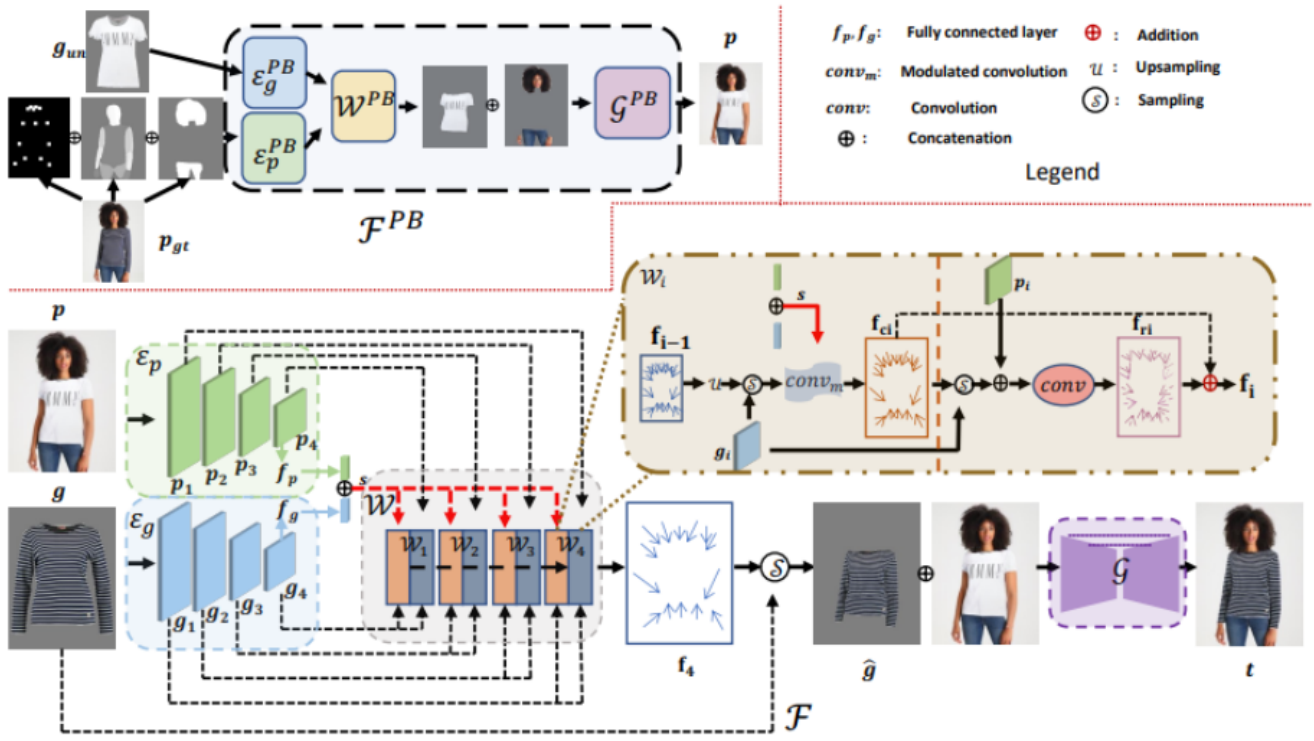

Figure 4: An illustration of our framework. As the input for the parser-free model $\mathcal{F}$, the pre-trained parser-based model $\mathcal{F}^{PB}$ produces an output image. The features of the human and clothing images are extracted by the two feature extractors in $\mathcal{F}$, respectively. A style vector is created by extracting the lowest-resolution feature maps from the human and garment images. The appearance flow map is produced by the warping module using the style vector and feature maps from the person and clothing images. The garment is then warped using the appearance flow. Ultimately, the person image and the twisted garment are concatenated and put into the generator to produce the target try-on image. Keep in mind that $\mathcal{F}^{PB}$ is only utilized in training.

4.1.1 PF AFN

The training flow of our model which is based on [(2)] includes two methods: a parser-based network PB-AFN and a parser-free network PF-AFN. In each method, there are two stages: a warping stage and a generating stage. First, we train PB-AFN with requirement inputs: segmentation map, keypoint pose and dense pose, real person image, and unpaired garment.

Going through this PB-AFN, the output of this is the image p where the person is wearing an unpaired garment gun. Next, p with g will be used as inputs for a parser-free model PF-AFN which produces output t. According to [(2)], this architecture gains from the fact that the parser-free model F may now be judged by personal image p_{gt}

$$\mathcal{F}^* = arg\ \min_{\mathcal{F}} ||t - p_{gt}|| \tag{1}$$

where $t = \mathcal{F}(p, g)$ is the output of $\mathcal{F}$ (as mentioned above)

Convolutional Encoders $\mathcal{E}$

For more details of the network, to extract the features of p and g, we use two convolutional encoders, $\mathcal{E}_p$ and $\mathcal{E}_g$. The architecture, which consists of stacked residual blocks, is the same for $\mathcal{E}_p$ and $\mathcal{E}_g$. $\mathcal{W}$ will utilize the extracted feature maps to predict the appearance flow.

Warping module $\mathcal{W}$

A style-based global appearance flow estimation module is the primary innovative feature of the suggested approach. Our method, which is based on a global style vector, first estimates a coarse appearance flow via style modulation and then refines the predicted coarse appearance flow based on local feature correspondence. This is different from previous methods that estimate appearance flow based on local feature correspondence [(2), (7)], originally proposed in optical flow estimation [(6), (9)].

Our warping module $\mathcal{W}$ contains N stacked warping blocks ($\{\mathcal{W}_i\}_1^N$). There is a global style-based appearance flow layer (orange rectangle) and a local appearance flow refinement layer (blue rectangle) in each warping block. In addition, a global style vector N^{th} ($s \in \mathbb{R}^c$) is extracted by using the output from the Nth block of convolutional encoders $\mathcal{E}_p$ and $\mathcal{E}_g$ respectively. s can be denoted as p_N and g_N, as:

$$s = [f_p(p_N), f_g(g_N)] \tag{2}$$

where the above formula ($[\cdot, \cdot]$) indicates concatenation and f_p and f_g are fully connected layers.

Also, a coarse flow predicted by style modulation can be represented as:

$$\mathbf{f_{ci}} = conv_m(\mathcal{S}(g_{N+1-i}, U(\mathbf{f_{i-1}})), s) \tag{3}$$

where U is the upsampling operator, $\mathbf{f_{i-1}}$ is the expected flow from the previous warping block, $\mathcal{S}(\cdot,\cdot)$ is the sampling operator, and $conv_m$ stands for modulated convolution [(14)]. It should be noted that the first block $\mathcal{W}_1$ in $\mathcal{W}$ only accepts the style vector and the lowest resolution garment feature map, or $\mathbf{f_{c1}} = conv_m(g_N, s)$. The global style vector and the garment feature map determine the predicted $\mathbf{f_{ci}}$. As a trade-off, the style vector s can only estimate the local fine-grained appearance flow with limited accuracy because it is a global representation (Fig. 5). Therefore, a local correspondence-based appearance flow refinement layer is illustrated that aims to estimate a local fine-grained appearance flow:

$$\mathbf{f_{ri}} = conv(\mathcal{S}(g_{N+1-i}, \mathbf{f_{ci}}), p_{N+1-i}) \tag{4}$$

where *conv* stands for convolution and $\mathbf{f_{ri}}$ for the anticipated refining flow. At its core, the refinement layer calculates the refinement flow using the local correspondence, which is the correspondence between the clothing feature and the features of the warped person within the same receptive field.

Lastly, we combine the local fine-grained appearance flow with the coarse flow as each warping block's output:

$$\mathbf{f_i} = \mathbf{f_{ci}} + \mathbf{f_{ri}} \tag{5}$$

The clothing is warped using the estimated appearance flow $\mathbf{f_N}$ from the final block in W:

$$\hat{g} = \mathcal{S}(g, \mathbf{f_N}) \tag{6}$$

The person image and the warped garment $\hat{g}$ are then concatenated and sent into a generator to create the target try-on image:

$$t = \mathcal{G}([\hat{g}, p]) \tag{7}$$

While the parser-free generating module concatenates the warped clothes and the tutor pictures use as inputs, the parser-based generative module concatenates the warped clothes, human posture estimate, and the conserved region on the human body. Both modules use residual connections in conjunction with the Res-UNet architecture, which is based on the UNet [(14)] framework and can maintain the features of the twisted clothing while producing realistic try-on outcomes.

4.1.2 PB AFN

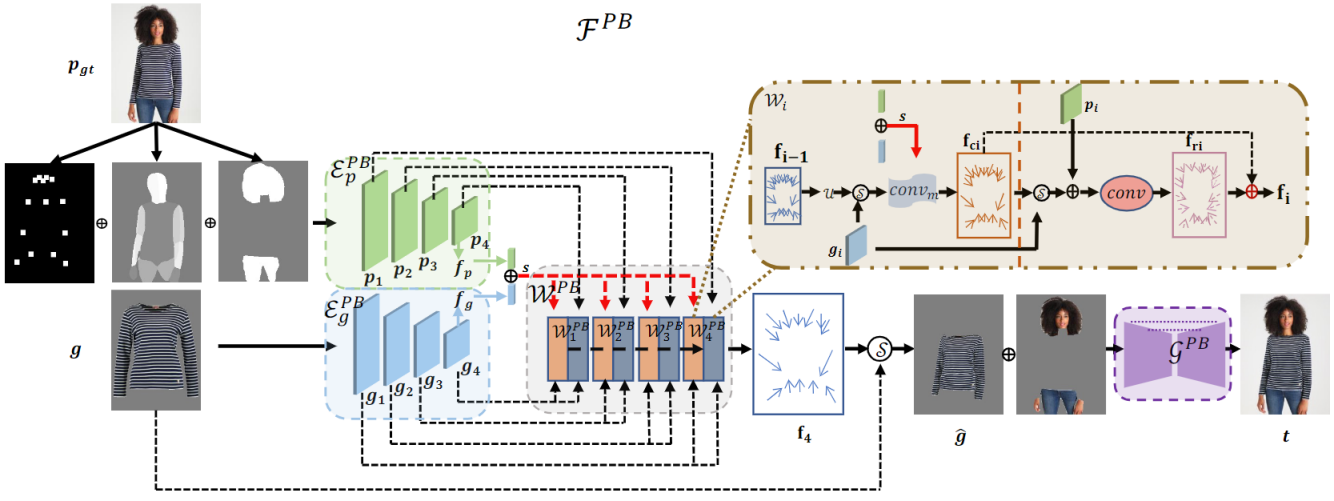

Figure 5: A schematic of parser-based model FPB.

In Fig. 2, the architecture of $\mathcal{F}^{PB}$ is shown. It and F have identical internal architecture. The sole distinction is that the generator of the person encoder in $\mathbf{F}^{PB}$ receives the masked person picture and the warped garment as inputs, while the person encoder itself receives the de-clothed

person representation (position, dense pose, and human segmentation map).

Images of paired people and clothing are used to train $\mathcal{F}^{PB}$. More precisely, we extract the posture, dense pose, and human segmentation map for the person image using the human parser, the off-the-shelf pose detection model, and the dense pose model. The retrieved representations are subsequently input into the person encoder by concatenating them. The information flows in all other cases are identical to those in the parser-free model $\mathcal{F}$.

4.2 Training hyperparameters

The parser-based network method is trained first which contains a warping module and generator module. In the warping module, we train within 100 epochs with 50 epochs iterate at a starting learning rate of 0.00005 and 50 epochs left iter to linearly decay learning rate to zero. In the generator module of parser-based, we continue to use the last warping module checkpoint to produce the warp garment for the generator to fit it into the human pose with the same hyperparameters in the warping stage. Note that the warping module is continuously trained while the generator module stage is trained.

Next, the second network is parser-free which also contains a warping and generator module but this time, this network uses the above approach - parser-based network method - to criteria itself. We used the same learning rate of 0.00005 for both warping and generator modules compared to the parser-based method. In this parser-free warping stage, we trained with 200 epochs which the last 100 epochs used to decay the learning rate. For the generator in parser-free, 100 epochs are used while the last 50 epochs decay the learning rate.

4.3 Loss

Perceptual loss (VGG loss):

To criteria image super-resolution and style transfer, VGG Loss is commonly used for this problem. By including additional features into the original image, super-resolution seeks to produce high-resolution images from low-resolution ones. Style transfer applies the content of one image

to the style of another.

VGG Loss is used in both cases to quantify the perceptual similarity of two images. By reducing the VGG Loss, the generated images become more natural and lifelike.

VGG Loss is a content loss technique that evaluates the similarity between two images by utilizing features taken from the VGG network. Widely employed in style transfer and image super-resolution, it has been demonstrated to yield more perceptually similar outcomes than pixel-wise losses.

L1 Loss:

The measurement of errors between paired observations that represent the same phenomenon is called Mean Absolute Error (MAE), or L1 Loss. It represents the average of the absolute errors. Knowing if the MAE units match the anticipated target helps determine whether or not the error's magnitude warrants concern. The mean average of these errors, or MAE, gives us an understanding of the model's performance across the entire dataset. One method of comparing predictions with actual results is to use the mean absolute error.

$$\text{Loss} = \frac{1}{N} \sum_{i=1}^{N} |I_{\text{real}_i} - I_{\text{fake}_i}| \tag{8}$$

L2 Norm:

A vector's magnitude in a multidimensional space is expressed in terms of the L2 norm, sometimes referred to as the Euclidean norm. The square root of the sum of the squares of the vector's component squares is used to calculate it.

The L2 norm is frequently employed in machine learning as a regularization strategy to stop models from overfitting. The model is punished for having big weights by adding the L2 norm of its weights to the loss function. This serves to prevent the model from better generalizing to unseen data and fitting the noise in the training data.

Furthermore, the similarity between data points in a multi-dimensional space is measured using

the L2 norm as a distance metric in a number of machine learning methods, including k-nearest neighbors (KNN) and support vector machines (SVM).

General formula of L2 Norm:

$$||x||_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} \tag{9}$$

Total Variance Loss (TV Loss):
It was found that optimization aimed just at minimizing style and content losses produced outputs that were extremely noisy and pixelated. Total variation loss was introduced to handle the same. To prevent noisy and unnecessarily pixelated outcomes, this is implemented to ensure spatial continuity and smoothness in the created image.

$$V_{\text{aniso}}(y) = \sum_{ij} \sqrt{|y_{i+1,j} - y_{i,j}|^2} + \sqrt{|y_{i,1+j} - y_{i,j}|^2} = \sum_{i,j} |y_{i+1,j} - y_{i,j}| + |y_{i,1+j} - y_{i,j}| \tag{10}$$

where $y_{i,j}$ denotes the pixel value at position (i, j) in the image, the $|| \cdots ||^2$ represents the L2 norm, or Euclidean norm, $i+1, j$ and $i, j+1$ correspond to neighboring positions in the horizontal and vertical directions.

4.3.1 PF Loss

The overall learning objective is:

$$L = \lambda_p L_p + \lambda_g L_g + \lambda_R L_R + \lambda_D L_D \tag{11}$$

where $\lambda_p, \lambda_g, \lambda_R$ and λ_D denote the hyperparameters for balancing the four objectives, L_p is a perceptual loss between the output of F and the ground truth person picture p_{gt} to train our

model:

$$L_p = \sum_i ||\phi(t) - \phi_i(p_{gt})|| \tag{12}$$

where ϕ_i is the i^{th} block of the pre-trained VGG network.

L_g is a L1 loss on the warped garment to supervise the training of the warping model W:

$$L_g = ||\hat{g} - m_g \cdot p_{gt}|| \tag{13}$$

where m_g is the p_{gt} garment mask that an off-the-shelf human parsing model predicts.

Our clothing flow estimate network matches pixel-to-pixel between source and target clothing regions, resulting in more accurate geometric alterations and photorealistic outcomes. However, employing dense flows frequently displays unpleasant artifacts without sufficient regularization, thus we further incorporate a total variation loss that regularizes the estimated flow field to ensure smoothness.

$$L_R = \sum_i || \bigtriangledown \mathbf{f_i}|| \tag{14}$$

where $||\bigtriangledown \mathbf{f_i}||$ is the generalized charbonnier loss function on the predicted flow from each block in W which is similar in spirit to the regularization term in the TV-L1 method [(28)] for estimating optical flow.

L_D is a distillation loss to direct the learning of person encoder $\mathcal{E}_p$ in $\mathcal{F}$ since the inputs (segmentation map, keypoint pose, and dense pose) to the parser-based person encoder ($\mathcal{E}^{PB}$) include more semantic information than those of the parser free model $\mathcal{F}$ (person image):

$$L_D = \sum_i ||p_i^{PB} - p_i|| \tag{15}$$

where p_i^{PB} is the output feature map from i_{th} block in the person encoder $\mathcal{P}_i^{PB}$ in the pre-trained parser-based model $\mathcal{F}^{PB}$.

4.3.2 PB Loss

$\mathcal{F}^{PB}$ is trained with three losses, similar to $\mathcal{F}$. First, we apply a perceptual loss [(7)] between the ground truth person picture p_{gt} and the output of $\mathcal{F}^{PB}$.

$$L_p = \sum_i ||\phi_i(t) - \phi_i(p_{gt})|| \tag{16}$$

where is the ϕ_i block of the pre-trained VGG network.

We apply a loss on the warped garment in order to oversee the training of the warping model $\mathcal{W}^{PB}$:

$$L_g = ||\hat{g} - m_g \cdot p_{gt}|| \tag{17}$$

where m_g is the p_{gt} clothing mask that the off-the-shelf human parsing model predicts [(4)]. In keeping with the convention of earlier appearance flow algorithms [(3), (6)], we additionally regularize the smoothness of the predicted flow from every block in $\mathcal{W}$:

$$L_R = \sum_i || \bigtriangledown \mathbf{f_i}|| \tag{18}$$

$|| \bigtriangledown \mathbf{f_i}||$ is the generalized charbonnier loss function[].

The overall learning objective is:

$$L = \lambda_p L_p + \lambda_g L_g + \lambda_R L_R \tag{19}$$

where λ_p, λ_g and λ_R denote the hyperparameters for balancing the three objectives.

4.4 Dataset

We use the VITON dataset to evaluate our model. It is the most widely utilized dataset from earlier VTON studies. VITON has a testing dataset with 2,032 pairs and a training set with 14, 221 picture pairs. The resolution of the photographs of the person and clothing is 256 $\times$ 192.

We also create a testing dataset, denoted by augmented VITON, to evaluate the model's robustness to the randomly positioned person image (see example in Fig. 4) with larger misalignments with the garment images in the original dataset. As most testing person images in VITON are well positioned such that the person image and the garment are well pre-aligned (e.g., most corresponding regions in the person image and garment image are roughly located in the same receptive field), it is not suited for this evaluation.

Person and garment Different human poses and different styles of clothes are contained in this dataset. Also, cloth masks are prepared for model implementation which is produced by OpenCV.

Figure 6: Random sample of dataset of human poses, garments, and garment masks.

Human parsing This is the task of segmenting a human image into different fine-grained semantic parts such as the head, torso, arms, and legs. The VITON dataset has included human parsing. Also, we can use Self Correction Human Parsing for custom datasets.

Figure 7: Random sample of dataset of human parsing []

Openpose Carnegie Mellon University created the real-time human pose estimation technology known as OpenPose. It is a computer vision tool that can precisely calculate the position of the human body in three dimensions and detect and track the human body in real-time. In our approach, we use Openpose to detect the 2D positions of body joints (key points) in images and save the output in JSON files for implementation purposes.

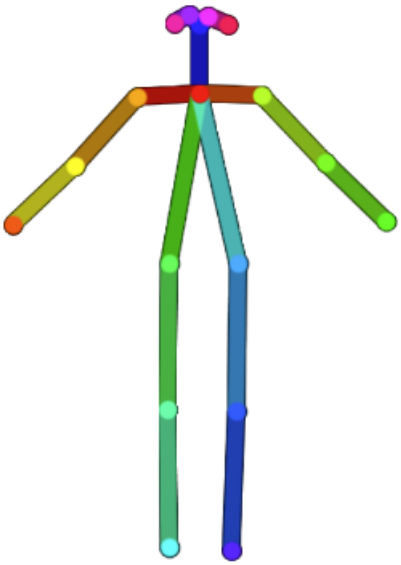

Figure 8: The visualization of Openpose images which are converted to JSON files.

Dense pose In order to understand the pose and surface geometry of the human body in images or videos, Dense pose which was first introduced by Facebook AI Research (FAIR) appears to solve this problem. This work aims at mapping all human pixels of an RGB image to the 3D surface of the human body. For preparation of the dataset, we convert densepose output into Numpy extension files.

Figure 9: The visualization of densepose images which are converted to numpy files.

5 Result

5.1 Experiments

As mentioned before, we tried running the model again based on the data that was processed by our method. There have been a few cases with better results when running on data provided by the author. The figures below show the difference between the original PF-AFN model and ours.

Figure 10:

Figure 11:

Figure 12:

We also re-ran the evaluation metrics and got results approximately the same as the author's.

Table 2: *Metrics Comparison*

	PF-AFN	**OURS**
FID	9.882	8.9054
SSIM	0.8773	0.8805

Evaluation metrics

FID: A metric called the Frechet Inception Distance score (FID) determines the separation between feature vectors computed for generated and real pictures.

To generate the try-on results during the test, a target clothing item and a reference human image are provided. Because we do not have ground-truth images (reference person wearing the target clothes), we use the Frechet Inception Distance (FID) [(29)] as an evaluation metric, which captures the similarity of generated images to real images* (i.e., reference person images). A lower FID score suggests high-quality outcomes. According to Rosca et al. [(31)], using the Inception Score (IS) [(30)] to models trained on datasets other than ImageNet can result in misleading findings. Therefore, we do not use the IS.

SSIM: The structural similarity index measure (SSIM) [(32)] performs the comparison between the two images based on Luminance, Contrast, and Structure. The higher the value of SSIM, the more similarity between the two images.

The majority of picture quality assessment algorithms rely on quantifying errors between a reference and a sample image. One popular metric is to quantify the difference in the values of each relevant pixel between the sample and reference images (for example, Mean Squared Error). To distinguish the variations between the information retrieved from a reference and a sample scene, the human visual perception system is highly capable of recognizing structural information from a scene. As a result, the SSIM metric that replicates this tendency will perform better on tasks

that require distinguishing between a sample and a reference image.

Loss results obtained from two-stage training of PBAFN and PFAFN

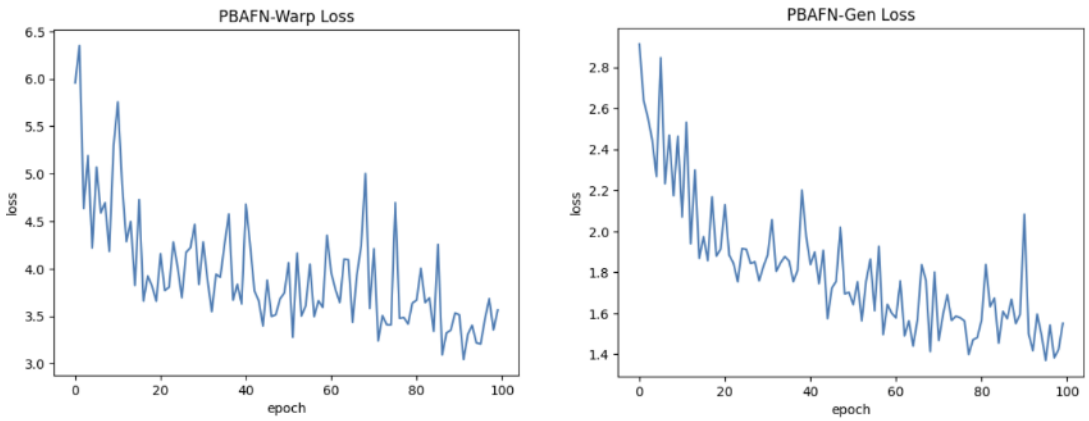

Figure 13: PBAFN Loss

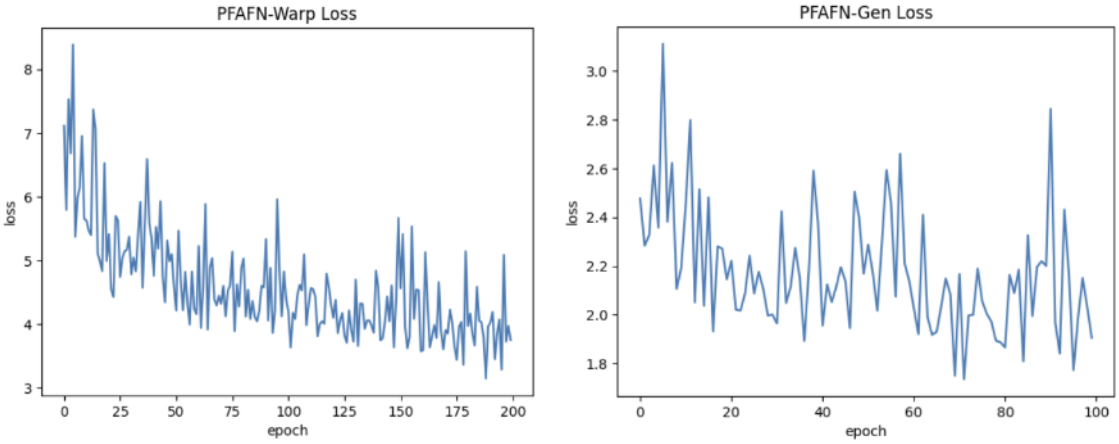

Figure 14: PFAFN Loss

Ablation study: This experiment validates the design of our appearance flow estimating blocks ($\mathcal{W}_i$). We first tested our method with solely global style modulation (SM)-based appearance flow estimation, utilizing $\mathbf{f_{ci}}$ in Equation 3 for each $\mathcal{W}_i$. We then test our technique with only refinement flow (RF) estimation, i.e., utilizing $\mathbf{f_{ri}}$ in Equation 4 in every $\mathcal{W}_i$. Finally, we experiment with our combined method (SM + RF), which estimates the appearance flow worldwide using style modulation and then refines it locally using local correspondence.

The suggested global style modulation (SM) based appearance flow approach surpasses the lo-

Figure 15: Comparing results with only $\mathbf{f_{ci}}$ used in $\mathcal{W}_i$ and $\mathbf{f_{ci}} + \mathbf{f_{ri}}$ used in $\mathcal{W}_i$.

cal correspondence method. Combining them improves overall performance. Fig.15 shows that our global style modulation method may not effectively forecast fine-grained appearance flow, such as sleeve areas, resulting in unsatisfactory try-on images. However, relying solely on local correspondence for appearance flow estimate, such as employing $\mathbf{f_{ri}}$ in $\mathcal{W}_i$, has limitations when regions are not in the same receptive field. $\mathbf{f_{ri}}$ is unable to effectively assess appearance flow when there is a significant mismatch between input photos of the person and garment (Fig. 16). After using $\mathbf{f_{ci}}$ to eliminate misalignment, our model successfully overcomes the challenge.

Figure 16: Comparing results with only $\mathbf{f_{ri}}$ used in $\mathcal{W}_i$ and $\mathbf{f_{ci}} + \mathbf{f_{ri}}$ used in $\mathcal{W}_i$ in the case of large misalignment between the input person image and garment image.

5.2 Fail case analysis

During the process of training PF-AFN model to generate images of clothing item on the person's body, some specific problems appeared, causing unexpected results during the image generation process. Below is a detailed descriptions of these issues:

- **Incomplete rendering of both arms due to poor output:** A significant challenge faced by the model is its inability to generate high-quality images displaying both arms of the model. This could be due to inaccuracies in the image generation process, possibly stemming from the model's inability to learn the complex relationships between different body parts. When encountering an individual wearing long-sleeved clothing and then trying on short-sleeves, the model fails to generate images with the arms accurately displayed in short sleeves.

- **Unable to process the length of the fitting garment:** Another problem is that the model is not able to adapt to changes in the shape and length of the fitting garment. This is because the model has not established a relationship between the body shape of the model and the required length of the fitting garment, so the results generated do not accurately reflect the fitting process.

- **The arms are still partially visible with the sleeves:** When trying on a short-sleeved shirt over a long-sleeved shirt, there is a glitch where part of the long-sleeved shirt remains visible on the arms. This could be due to the model's inability to align the arm length during the try-on process, resulting in an error.

- In addition, there are a few other cases that can lead to try-on failures. Low-quality input images, such as those with noise or blur, pose significant challenges to virtual try-on systems. These images hinder the model's ability to accurately extract and process crucial information, leading to several detrimental consequences.

Figure 17: Bad cases: (1) Incomplete rendering of both arms, (2) Person's shirt too long, (3) Arms partially showing sleeves.

6 DISCUSSIONS

6.1 Experimental Results

Over the past 14 weeks, we have researched the best models for virtual try-on models and selected the most suitable one for our project. We conducted a comprehensive review of the code, dataset, and input processing steps proposed by the original authors. We then performed experiments based on the authors' input and achieved promising results. We also attempted to optimize the model under controlled experimental conditions and obtained encouraging results when testing on images generated after applying the processing steps. Our input processing results showed some improvements compared to the original images. However, due to the time constraint of 14 weeks, our optimization process was not fully completed. We collaborated with another project team on building a website for virtual clothing try-ons and integrated our AI technology into the website. However, the overall running time is still relatively slow because the images need to go through multiple processing steps to generate the final results.

6.2 User Interaction and Adaptability

Our virtual try-on system uses advanced AI technology to create realistic images, helping users experience trying on clothes naturally and flexibly. Users have the flexibility to seamlessly try on various clothing items, from formal wear to sportswear, without changing their primary image, enhancing their online shopping experience such as Trying on real clothes, saving time and effort of shopping in person.

6.3 Challenges and Future Enhancements

While our virtual try-on system has demonstrated promising results, there are still challenges to address and improvements to be made for future enhancements. Certain challenges, such as when parts of the hand are obscured, can hinder the image processing steps. This lack of necessary information provided to the main model can result in lower-quality generated images

and slower processing speeds. We will continue to conduct research and experiments with a focus on optimizing the model architecture and training process to achieve higher accuracy and faster inference times.

7 CONCLUSIONS

Our research and development efforts over the past 14 weeks have yielded a promising virtual try-on system. Initial experiments demonstrate the system's ability to create realistic images and provide a natural user experience. However, there are areas for improvement, particularly in model optimization and processing speed. By overcoming these challenges and implementing the proposed enhancements, we believe our virtual try-on system has the potential to revolutionize the online shopping experience by offering a more efficient, personalized, and engaging way to try on clothes.

8 REFERENCES

[1] Chongjian Ge, Yibing Song, Yuying Ge, Han Yang, Wei Liu, and Ping Luo. *Disentangled cycle consistency for highly realistic virtual try-on.* In CVPR, 2021.

[2] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, and Ping Luo. *Parser-free virtual try-on via distilling appearance flows.* In CVPR, 2021.

[3] Jean Duchon. *Splines minimizing rotation-invariant seminorms in Sobolev spaces.* In Constructive theory of functions of several variables, pages 85–100. Springer, 1977.

[4] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. *View synthesis by appearance flow.* In ECCV, 2016.

[5] Ayush Chopra, Rishabh Jain, Mayur Hemani, and Balaji Krishnamurthy. *Zflow: Gated appearance flow-based virtual try-on with 3d priors.* In ICCV, 2021.

[6] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. *Flownet: Learning optical flow with convolutional networks.* In CVPR, 2015.

[7] Xintong Han, Xiaojun Hu, Weilin Huang, and Matthew R Scott. *Clothflow: A flow-based model for clothed person generation.* In ICCV, 2019.

[8] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. *Viton: An image-based virtual try-on network.* In CVPR, 2018.

[9] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. *Flownet 2.0: Evolution of optical flow estimation with deep networks.* In CVPR, 2017.

[10] Yujun Shen and Bolei Zhou. *Closed-form factorization of latent semantics in gans.* In CVPR, 2021.

[11] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J Black. *Clothcap: Seamless 4D clothing capture and retargeting.* TOG, 36, 2017.

[12] Sen He, Wentong Liao, Michael Ying Yang, Yi-Zhe Song, Bodo Rosenhahn, and Tao Xiang. *Disentangled lifespan face synthesis.* In ICCV, 2021.

[13] Thibaut Issenhuth, Jer´e'mie Mary, and Cle'ment Calauzenes. *Do not mask what you do not need to mask: a parser-free virtual try-on.* In ECCV, 2020.

[14] Tero Karras, Samuli Laine, and Timo Aila. *A style-based generator architecture for generative adversarial networks.* In CVPR, 2019.

[15] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. *Analyzing and improving the image quality of Stylegan.* In CVPR, 2020.

[16] Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, Liang Lin, and Meng Yang. *Toward characteristic preserving image-based virtual try-on network.* In ECCV, 2018.

[17] 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 CVPR, 2020.

[18] Ruiyun Yu, Xiaoqi Wang, and Xiaohui Xie. *Vtnfp: An image-based virtual try-on network with body and clothing feature preservation.* In ICCV, 2019.

[19] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-net: Convolutional networks for biomedical image segmentation.* In MICCAI, 2015.

[20] Roy Or-El, Soumyadip Sengupta, Ohad Fried, Eli Shechtman, and Ira Kemelmacher-Shlizerman. *Lifespan age transformation synthesis.* In ECCV, pages 739–755, 2020.

[21] Anton Cherepkov, Andrey Voynov, and Artem Babenko. *Navigating the gan parameter space for semantic image editing.* In CVPR, 2021.

[22] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. *Interfacegan: Interpreting the disentangled face representation learned by Gans.* TPAMI, 2020.

[23] Igor Santesteban, Miguel A Otaduy, and Dan Casas. *Learning-based animation of clothing for virtual try-on.* In Computer Graphics Forum, volume 38, 2019.

[24] Roy Or-El, Soumyadip Sengupta, Ohad Fried, Eli Shechtman, and Ira Kemelmacher-Shlizerman. *Lifespan age transformation synthesis.* In ECCV, pages 739–755, 2020.

[25] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. *Tryongan: body-aware try-on via layered interpolation.* TOG, 40(4):1–10, 2021

[26] 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.* In SIGGRAPH Asia, 2021.

[27] Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H Li, and Ge Li. *Deep image spatial transformation for person image generation.* In CVPR, 2020.

[28] Lijie Fan, Wenbing Huang, Chuang Gan, Stefano Ermon, Boqing Gong, and Junzhou Huang. *End-to-end learning of motion representation for video understanding.* In CVPR, 2018.

[29] 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.* In NeurIPS, 2017.

[30] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. *Improved techniques for training gans.* In NeurIPS, 2016.

[31] Mihaela Rosca, Balaji Lakshminarayanan, David WardeFarley, and Shakir Mohamed. *Variational approaches for auto-encoding generative adversarial networks.* arXiv preprint arXiv:1706.04987, 2017.

[32] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. *Image quality assessment: from error visibility to structural similarity.* TIP, 13(4):600–612, 2004.