Attire Fit In

A

Project Report

Submitted for the partial fulfilment

of B.Tech. Degree

in

COMPUTER SCIENCE & ENGINEERING

by

Rishabh Balaiwar (1805210042)

Nishtha Shukla (1805232038)

Himani Srivastava (1805232026)

Under the supervision of

Dr. Upendra Kumar 04/06/2022

Ms. Deepa Verma Dverma

Department of Computer Science and Engineering

Institute of Engineering and Technology

Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh.

May 2022

CONTENTS

DECLARATION i
CERTIFICATE ii
ACKNOWLEDGEMENT iii
ABSTRACT iv
LIST OF FIGURES v

1. CHAPTER 1 - INTRODUCTION 1 – 2
2. CHAPTER 2 - LITERATURE REVIEW 3
3. CHAPTER 3
   - 3.1 METHODOLOGY
     - 3.1.1 PRE-PROCESSING 4 – 5
     - 3.1.2 SEGMENTATION GENERATOR 5 – 6
     - 3.1.3 CLOTHING IMAGE DEFORMATION 6 -7
     - 3.1.4 ATTIRE FITTING via ALIAS GENERATOR 7 – 8
4. CHAPTER 4 - EXPERIMENTAL RESULTS
   - 4.1 EXPERIMENTAL SETUP
     - 4.1.1 DATASET 9
     - 4.1.2 TRAINING INFERENCE 9 – 10
   - 4.2 IMPLEMENTATION
     - 4.2.1 PRE-PROCESSING 10
     - 4.2.2 TRAINING 11 – 16
5. CHAPTER 5
   - 5.1 CONCLUSION 17
   - 5.2 FUTURE WORKS 17

REFERENCES 18 - 19

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our belief and knowledge, it contains no material previously published or written by another person or material which to a substantial error has been accepted for the award of any degree or diploma of university or other institute of higher learning, except where the acknowledgement has been made in the text. The project has not been submitted by us at any other institute for requirement of any other degree.

Submitted by: - Date: 25-05-2022

(1) Name: Rishabh Balaiwar
Roll No: 1805210042
Branch: Computer Science & Engineering
Signature: RBalaiwar

(2) Name: Nishtha Shukla
Roll No: 1805232038
Branch: Computer Science & Engineering
Signature: AniShukla

(3) Name: Himani Srivastava
Roll No: 180232026
Branch: Computer Science & Engineering
Signature: Himani

CERTIFICATE

This is to certify that the project report entitled "Attire Fit In" presented by Rishabh Balaiwar, Nishtha Shukla and Himani Srivastava in the partial fulfillment for the award of Bachelor of Technology in Computer Science and Engineering, is a record of work carried out by them under my supervision and guidance at the Department of Computer Science and Engineering at Institute of Engineering and Technology, Lucknow.

It is also certified that this project has not been submitted at any other Institute for the award of any other degrees to the best of my knowledge.

Dr. Upendra Kumar

Ms. Deepa Verma

Department of Computer Science and Engineering

Institute of Engineering and Technology, Lucknow

ACKNOWLEDGEMENT

With immense pleasure, we final year undergraduates in Computer Science and Engineering at Institute of Engineering & Technology, Lucknow, present the Final Year Project report as a part of the curriculum of Bachelor of Technology.

We wish to express our deep gratitude towards our faculty advisor Dr. Upendra Kumar and Ms. Deepa Verma who provided us with the opportunity to work on this project and constantly backed us in every manner possible to give our best. We also wish to extend our profound thanks to our Head of Department, Dr. Divakar Singh Yadav for bestowing upon us an unending support during this project and giving time to make each concept and it's usage crystal clear.

We would like to take the opportunity to extend our regards to our parents for giving encouragement, enthusiasm, and invaluable assistance without which completion of this project would not have been possible.

Finally, we would like to apologize to all other unnamed people who helped us in numerous ways for the implementation of this project.

Rishabh Balaiwar	**Nishtha Shukla**	**Himani Srivastava**
1805210042	**1805232038**	**1805232026**

ABSTRACT

The COVID-19 pandemic triggered massive changes in the behaviour and lifestyle of people across the world. All the industries were impacted due to the restrictions imposed as a precaution to curb the spread of the virus. One such industry is the global fashion industry. There have been restrictions imposed on the trial of clothes in the shopping centres at regular intervals. There were numerous concerns. Can the Covid-19 infection be transferred by clothing? After a customer has tried on a piece of apparel, what happens next?

It is known that the virus can survive for a long time on a specific surface. This extended period can pose a risk to doorknobs, mirrors, and even chairs and hangers for dressing rooms. In addition, trying on the clothes themselves (touching the buttons and zippers) can pose a risk, no matter how small. A survey conducted by predictive analytics retailer First Insight in late April found that 65% of women and 54% of men find it unsafe to wear clothes in the dressing room. It is important to keep safe practices in mind when moving these next steps together into familiar but unfamiliar areas.

While reading about these concerns that people had regarding the spread of virus through clothes, we thought of an alternative that would help people visualize how the clothes would look on them without changing into them. This virtual trial system can be installed in the clothing area, where the people would have an opportunity to capture their pictures through a camera and select the clothing they wish to try from a screen. The system would generate a picture which would wrap the selected piece of clothing onto the body of the customer and provide them an idea of how the piece would look.

LIST OF FIGURES

FIGURE NO	REFERENCE
1.1	Qualitative comparison of the baseline
3.1	Overview of Attire Fit-In
3.2	Generation of the segmentation map of image
3.3	U-Net Architecture
3.4	Flowchart of ALIAS Generator
4.1	Cloth mask
4.2	Pose map
4.3	Segmentation map
4.4	Target cloth
4.5	Reference image
4.6	Pose map & Segmentation map generation
4.7	Segmentation generator architecture
4.8	Geometric Matching Module
4.9	ALIAS Generator
4.10	Result-1 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image
4.11	Result-2 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image
4.12	Result-3 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image
4.13	Result-4 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image
4.14	The results as displayed to the user using Tkinter (Left to Right) The target apparel, the reference image, and the synthetic image
4.15	Real time implementation – 1
4.16	Real time implementation – 2
4.17	Real time implementation – 3
4.18	Real time implementation – 4

CHAPTER 1

1.1 INTRODUCTION

The trend of online shopping is increasing. Trying on clothes virtually in these times will allow the customers to get an idea of how the cloth will look on them, which will substantially improve the customer experience. This is heading towards the idea of virtual dress fitting or virtual fitting. Image-based virtual fitting is basically an image generation task that changes a person's garment to another garment specified in another product image.

‘Virtual try-on’ is like image synthesis, but it brings with it certain aspects that need to be looked after in order to provide a satisfactory experience to the user. Here we need to make sure that provided the image of the client and the garment they need to try, the synthesized output image must preserve:

(1) The clients' pose, body shape, and identity.
(2) Natural deformation of clothing according to the region of clothing desired by the individual, reflecting posture and skeleton structure.
(3) Intricacies of the apparel material.
(4) The parts of the body that were originally covered with the reference individual’s costume in the primary picture should render correctly.

Many approaches for virtual try-on of clothing products have been proposed to date. These methods suggest distorting the clothes picture to position it to settle on the human body first, then combining the warped clothing image with the person's image and doing pixel-level refining. Certain approaches additionally use segmentation maps to figure out the client's conformation from the end image. Still and all the resolution of the synthesised pictures using these methodologies is low as the deviation in the alignment between the contorted apparels and a human figure because of warping cloth images to fit the target body results in the remnants in the disarrayed portions, that are evident as and when the image dimensions increase. The ‘thin-plate spline’ (TPS) transformation is used in the existing ways to alter garment images. ClothFlow anticipates the optical flow maps of the apparel and the target garment regions to appropriately deform them. Because this method needs pixel-level prediction of clothing movement, it has higher processing costs than previous methods. Second, present techniques that use a simple U-Net architecture are insufficient for synthesising originally obstructed human body regions in the resultant high-resolution images. According to research, using a simple U-Net-based architecture to synthesise high-resolution pictures results in unbalanced training as well as images of poor grade. Furthermore, simply improving the photos at the pixel level is insufficient to preserve the intricacies of high-resolution clothing images.

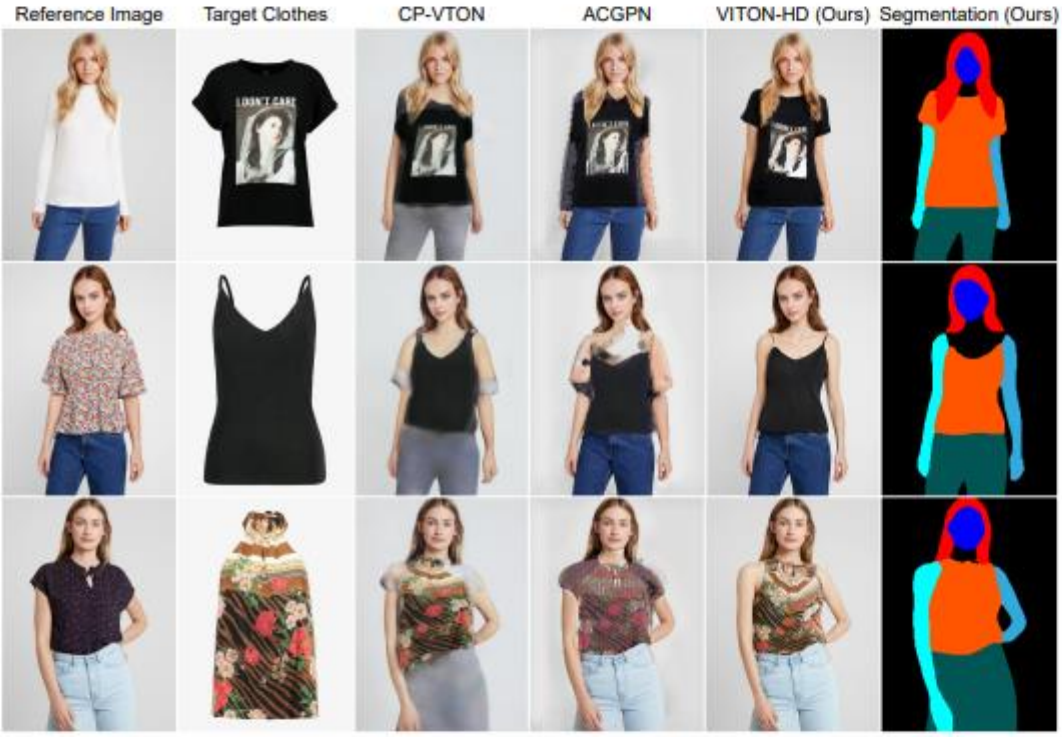


FIG 1.1 Qualitative comparison of the baseline

Several variants of these algorithms have been developed. Current research removes clothing information using a different clothing-agnostic person representation that utilizes posture information and the current segmentation map. Thereafter, the segmentation maps and the deformed clothing product are fed to the model for image generation. With additional information, the ALIignment Aware Segment (ALIAS) Normalization excludes information that is not relevant to the texture of the garment in the displaced area and propagates the semantic information throughout the network. Normalization uses the segmentation map individually to standardize and modulate standardized activations related to staggered regions and other regions. This ALIAS generator uses ALIAS normalization to synthesize images of customers wearing the target garment, and the misaligned areas are in the garment texture and garment details, maintaining the functional level of the details with multi-scale improvements at feature level.

Objectives:

- Utilize image-based virtual try-on methodology called Attire Fit-In to generate synthetic image of the client trying the apparel.
- Incorporate clothing-agnostic person portrayal in order to eliminate the reliance on the attire donned by the reference human figure in the first instance.
- ALIAS normalisation and ALIAS generator efficiently cater to the misalignment linking the distorted apparel and the target clothing portion, thereby maintaining the complexities of the target product.

CHAPTER 2

LITERATURE REVIEW

U-Net: The work presented in this paper gave a general idea of using U-Net architecture to generate segmentation maps required for the second phase of virtual-try on. To make greater use of the existing annotated samples, this study developed a network and training technique that largely relied on data augmentation. The architecture consisted of a contracting path to capture context and a symmetric expanding path that enables precise localization. (Ronneberger, 2015)

Normalization Layers. Modern deep neural networks find wide application of Normalisation Layers (Ioffe, 2015) (Ulyanov, 2016). If external data is utilised for the estimation of affine parameters, such normalization layer is called the conditional normalization layer. (De Vries, 2017) (Huang, 2017) Style transfer task makes extensive use of conditional stack normalization and adaptive instance normalization. Spatially changing affine transformations are applied to SPADE (Park, 2019) and SEAN (Zhu, 2020) using segmentation maps. To compute the mean and variances of the misaligned areas the normalisation layer uses the misalignment mask as external data. After standardization, the standardized activation map is modulated with Affine specifications derived through the human analysis map to retain the particulars associated with the semantics.

Geometric Matching Module. As per the work mentioned over GMM, it comprises of two feature extractors and a regression network. The two extracted features are utilised to generate correlation matrix, and the TPS parameter θ is predicted by the regression network with the help of correlation matrix. A set of convolution layers form a feature extractor, and a regression network comprises of a series of convolution layers followed by a fully connected layer. This module has been used to warp the clothes to fit the human body while deforming the clothing image. (Wang, 2018)

Virtual Try-On Approaches. There are two main categories for virtual try-on approaches: 3D model-based (Guan, 2012) (Sekine, 2014) (Pons-Moll, 2017) (Patel, 2020) approaches and 2D image-based. (Han, 2018) (Wang, 2018) (Han X. H., 2019) (Yu, 2019) (Yang, 2020) (Dong, 2019) Although the 3D model-based approach could precisely reproduce clothing, it is generally not applicable because it relies on 3D computational data. The 2D image-based approach is commutatively effective and suitable for functional operation because it does not rely on 3D information. The first introduction to the task of exchanging fashion items through people's images was proposed by Jetchev and Bergmann (Jetchev, 2017) of CAGAN. The same problem is addressed by incorporating a synthetic scaffold from coarse to fine, including the TPS conversion of clothing to VITON (Han, 2018). To improve the accuracy of deformation CP VTON (Dong, 2019) adopted a geometric fitting module to learn the parameters of the TPS transformation. To guide image composition attempts, VTNFP (Yu, 2019) and ACGPN (Yang, 2020) predicted a human analysis map of the person wearing the subject's clothing. Though none of the approaches could generate realistic images at high resolution.

CHAPTER 3

3.1 METHODOLOGY

The purpose of Attire Fit-In is to build a synthetic image ^*I* of the same individual dressed in the target clothes *c*, while preserving the stance and body shape of *I* and the features of *c*, provided a reference pciture *I* of a person and a garment picture *c*.

We will create a garment-agnostic person representation and utilize it as input after removing the clothes information from *I*. To exclude the garment information, the model will use both the pose map and the segmentation map of the individual. To aid in the development of ˆ*I*, the simulation will construct a segmentation map from the garment agnostic person description. Then, to roughly align c with the human body, we will distort it. Finally, after deforming *c*, we recommend using the ALIgnment-Aware Segment (ALIAS) normalisation method to remove misleading data from the misaligned area. The ALIAS generator uses the garment texture to fill in the misalignment area while keeping the clothing features.

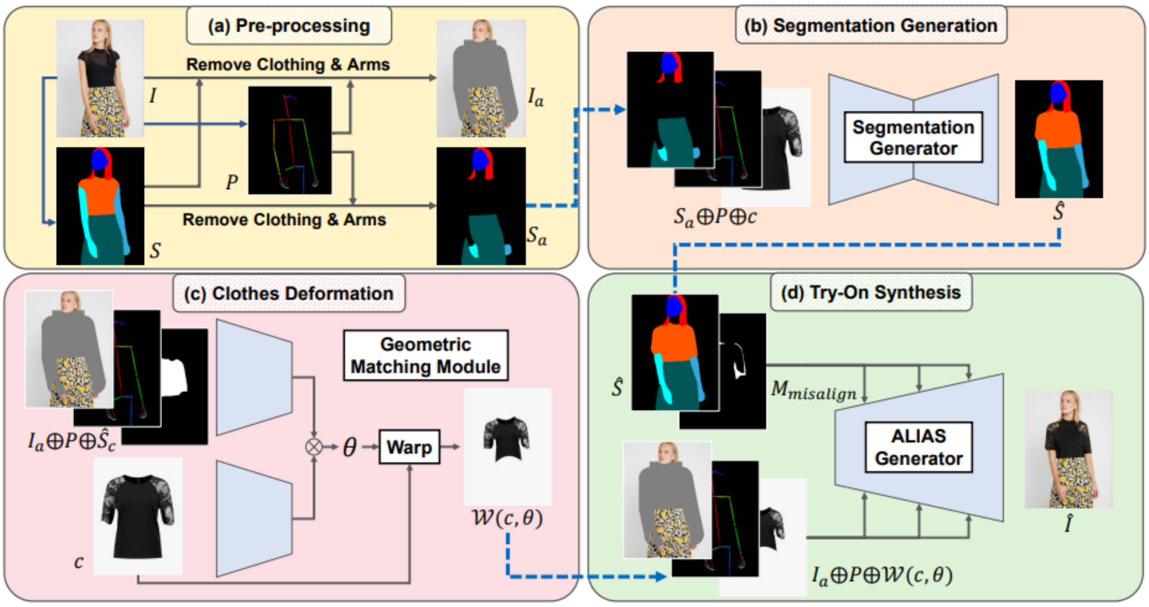


FIG 3.1 Overview of Attire Fit-In

3.1.1 PRE-PROCESSING

A person interpretation sans clothing data in *I* will be utilised to prepare the model using pairs of *c* and *I* previously dressed in *c* in the virtual try-on task. Such representations must meet the given requirements:

- remove the original item of clothing to be replaced,
- adequate knowledge to anticipate the person's stance and body shape should be preserved, and
- it is necessary to conserve the area to be spared (face, hands, etc.) while preserving the identity of the individual.

To tackle the issue of reproducing the body parts elaborately, our approach provides a garment-agnostic image I_a and a garment-agnostic segmentation map S_a as stage keys, which effectively remove the outline of the apparel while preserving the body portions that must be replicated. Our approach starts by predicting a segmentation map S and the pose map P of the picture I using the pre-trained networks. The segmentation map S is utilised to eliminate the to-be-changed garment region while preserving the remainder of the image. Because the hands are difficult to replicate, the position map P is used to eliminate the arms but not the hands. Create a cloth-independent image I_a and a cloth-independent segmentation map S_a based on S and P. This allows the prototype to eliminate the initial cloth data and retain the remaining of the picture.

3.1.2 SEGMENTATION GENERATOR

The segmentation generator anticipates the segmentation map ^S of the individual in the reference picture dressed in clothing c using the garment-agnostic person representation (S_a, P) and the target garment item c. We plan to teach G_S how to map S to (S_a, P, c), eliminating all information about the original clothing item. We use U-Net as the GS architecture. (Ronneberger, 2015)

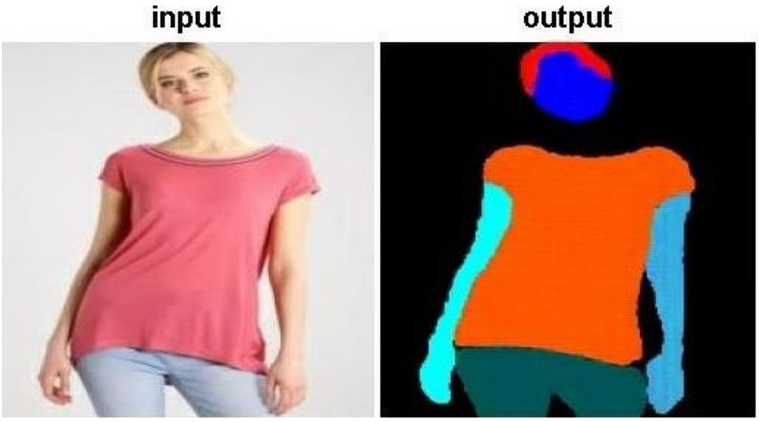


FIG 3.2 Generation of the segmentation map of image

The U-Net is a refined architecture solving most problems using the concept of a fully convolutional network. The central idea of the implementation is to use a continuous shrink layer and directly afterwards use an ‘up-sampling operator’ to achieve higher resolution output on the input image.

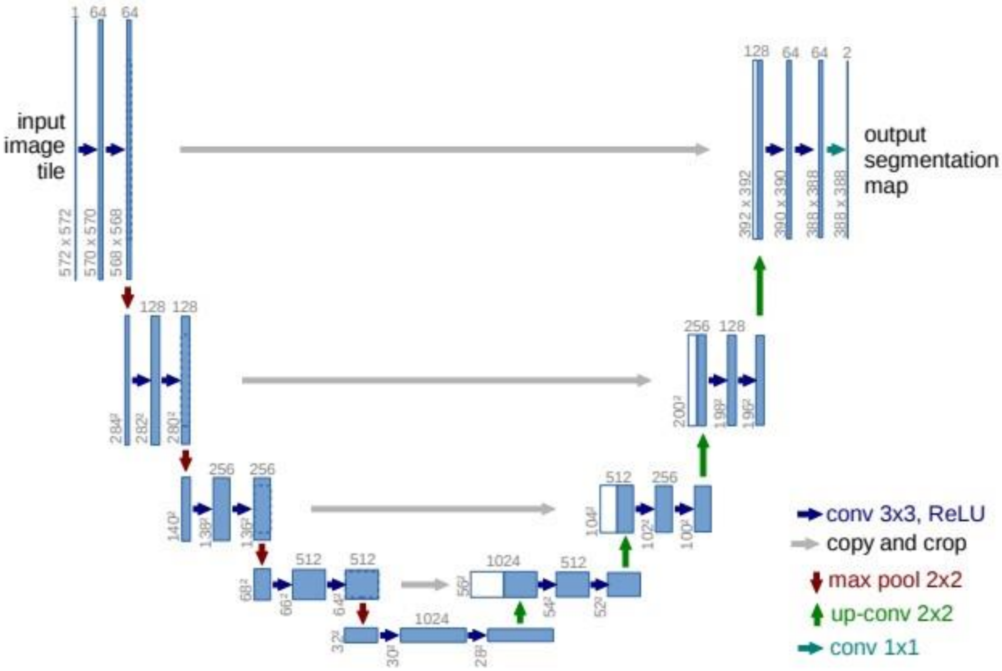


FIG 3.3 U-Net architecture (Ronneberger, 2015)

The architecture shows that an input image is passed through the model and then it is followed by a couple of convolutional layers with the ReLU activation function. There is a reduction in the image size because of the use of unpadded convolutions. Also, apart from the convolution block, there is an encoder block on the left and a decoder block towards the right.

The encoder block using the max-pooling layers reduces the image size. The encoder architecture also has a layer of convolution that repeats as the number of filters increases. Then, when we reach the decoder block, there is a gradual decrease in the convolutional layer, increasing the up-sampling in the layers following it.

There is also a presence of the skip connection which connects the previous output to the layer of the decoder block. These connections retain the loss from the preceding layers so that they can signify a more substantial global value. It helps produce better results.

3.1.3 CLOTHING IMAGE DEFORMATION

At this juncture, the chosen garment c is transformed to be in orientation with ^S_c, that is the garment region of ^S. The clothing-agnostic person representation (I_a, P) and S_c are inputs to the geometric matching module as described in CPVTON (Wang T. C., 2018). As mentioned in the cited paper, GMM embodies four divisions:

(1) Two feature extractors which extract prominent characteristics of I and c.
(2) The features thus extracted are combined into a single tensor using a correlation layer and fed as input to the regressor network.

(3) The spatial trans-formation parameters θ are predicted using the regression network.
(4) The image is warped to output $\hat{c} = T\theta(c)$ using the Thin Plate Spline (TPS) conversion module T.

First, a correlation matrix is generated between the features obtained from (I_a, P) and c. The regression network uses the correlation matrix as the input to predict the TPS transformation parameters. Then c will be warped by. The model use S_c derived from S during the training phase instead of $\hat{}S_c$. The L_1 loss between the distorted garment and the garment I_c recovered from I_c is used in the to train the module. In addition, a quadratic difference constraint is used to reduce the visible distortion of the distorted clothing image caused by the deformation. A common objective function that warps apparels to align with the human figure is described as follows:

$$L_{warp} = ||Ic - W(c, \theta)||1,1 + \lambda_{const}L_{const},$$

Where W is a function that transforms c using θ, L_{const} is a quadratic difference constraint, and λ_{const} is hyperparameters for L_{const}.

3.1.4 ATTIRE FITTING via ALIAS NORMALIZATION

On the grounds of the outputs from the preceding stages a final synthetic image $\hat{}I$ is generated at this stage. In essence, S is used to combine the garment-agnostic human representation (I_a, P) with the distorted garment picture $W(c, \theta)$. Each layer of the generator is fed with I_a, P, $W(c, \theta)$. For S we utilise the ALIgnment-Aware Segment (ALIAS) normalisation as a new conditional normalisation method. (Wang, 2018) By using S and the mask of these regions, ALIAS normalisation allows for the preservation of semantic information as well as the elimination of misleading information from misaligned regions.

ALIAS NORMALIZATION

ALIAS normalization (Choi, 2021) has two inputs: the synthetic segmentation map $\hat{S}$; the misalignment binary mask $M_{misalign} \in L^{(H \times W)}$, which excludes the warped mask of the target clothing image $W(M_c, \theta)$ from $\hat{S}_c$ (M_c denotes the target clothing mask), i.e.,

$$M_{align} = \hat{S}_c \cap W(M_c, \theta) \quad (3)$$

$$M_{misalign} = \hat{S}c - M_{align} \quad (4)$$

First, M_{align} and $M_{misalign}$ from the formula (3) and formula (4) Obtained. The reconstructed edition of $\hat{S}$ is, defined as $\hat{S}_{div}$, and $\hat{S}_c$ is separated into $\hat{S}$ to become M_{align} and $M_{misalign}$. Furthermore, the regions of $M_{misalign}$ and the other regions in h^i are standardised separately with the help of ALIAS normalisation. This is followed by standardized activation modulation using the affine transformation parameters derived from ˆS_{div}.

ALIAS Normalisation is basically utilised for the removal of the misleading information in the misaligned regions.

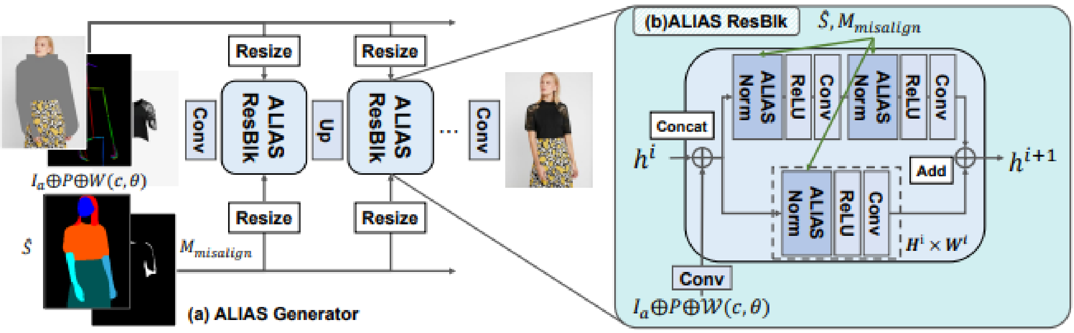


FIG 3.4 Flowchart of ALIAS generator

ALIAS GENERATOR

The 'ALIAS generator' uses a set of residual blocks with an up-sampling layer. Three convolutional layers and three ALIAS normalization layers are used to make up each ALIAS residual block. Since each of the block's work at different resolutions, so the inputs to Normalization Layer, S and $M_{misalign}$, are adjusted before inserting them into each layer. Likewise, the generator input (I_a, P, W $(c,\ \theta)$) is scaled to various resolutions. After passing through the convolutional layer, the resized inputs (I_a, P, W $(c,\ \theta)$) are concatenated with the activation of the preceding layer before each residual block, and each residual block improves activation with concatenated inputs. The network thus performs multi-scale functional level improvements and retains garment details more than a single pixel level improvement. Following SPADE (Park, 2019) and pix2pixHD (Wang T. C., 2018), the ALIAS generators are trained with conditional adversarial losses and feature matching losses and sensory losses.

CHAPTER 4

EXPERIMENTAL RESULTS

4.1 EXPERIMENT SET-UP

4.1.1 DATASET

For the creation of the dataset, we collected images of 1024×768 and inched 13,679 frontal-view female and top apparel picture pairings from an online shopping mall. Training and test sets of 11,647 and 2,032 pairs, respectively, were created from the sets. A paired scenario was evaluated using pairs of people and clothing photos, whereas an unpaired condition was evaluated using randomised clothing images. The paired setting replaces the original apparel item on the individual's picture, whereas the unpaired setting replaces the apparel item on the individual's picture with a separate item.

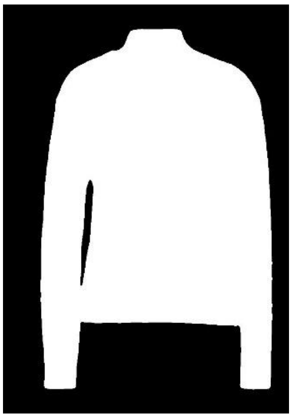


FIG 4.1 Cloth mask

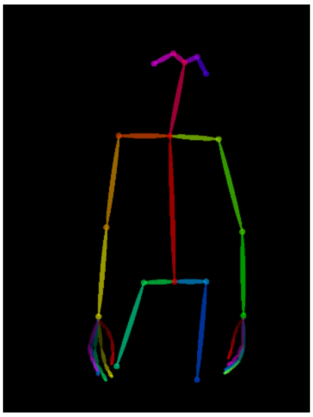


FIG 4.2 Pose map

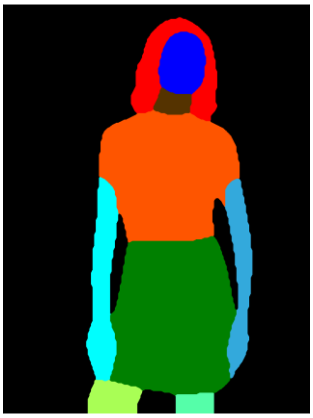


FIG 4.3 Segmentation map

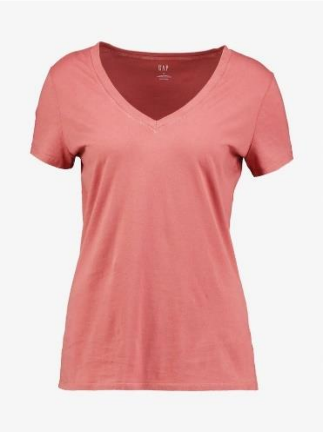


FIG 4.4 Target cloth

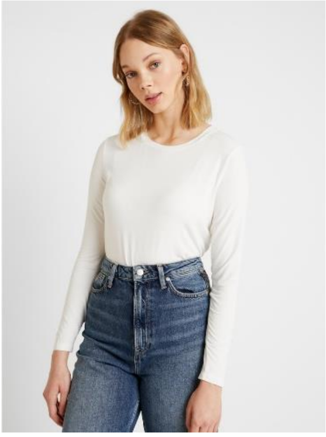


FIG 4.5 Reference image

4.1.2 TRAINING AND INFERENCE

Each stage was trained individually with the aim of reconstructing I from (I_a, c). We used S instead of $\hat{S}$ during training for the geometric matching engine and the ALIAS generator. We trained the segmentation generator and geometric matching engine at 256x192, while aiming to generate a 1024x768 try-on image. After predicting the segmentation map at 256x192, the segmentation generator scales it to 1024x768 in the inference phase and passes it on to the next steps. Similarly, the GMM predicts the TPS parameter θ at 256×192, and the ALIAS generator uses a 1024×768 garment picture warped by the parameters. We found that the performance of these two modules is better than the modules trained at 1024x768 at a reduced memory cost using this method. (Choi, 2021)

4.2 IMPLEMENTATION

4.2.1 PRE-PROCESSING

The details of building the clothing-agnostic person model are covered in this section. To begin, the prediction of a segmentation map S and a posture map P of the image I is conducted using the pre-trained networks.

Here to parse the human figure in an image efficiently in a single pass itself, a detection-free Part Grouping Network (PGN) is employed. PGN (Gong, 2018) rewrites instance-level human parsing by implementing two coupled sub-tasks that can be trained and fine-tuned together with the aid of an integrated network:

(1) instance-aware edge detection to group semantic parts into unique person instances.
(2) semantic part segmentation to assign each pixel as a human part (e.g., face, arms).

As a result, the common intermediate representation includes the capacity to characterise sections with fine details and infer instance characteristics of every individual portion. Subsequently, a basic instance segregation procedure is employed for obtaining final findings at the time of inference, and thus the segmentation map is produced as the resultant image, as shown in the figure below.

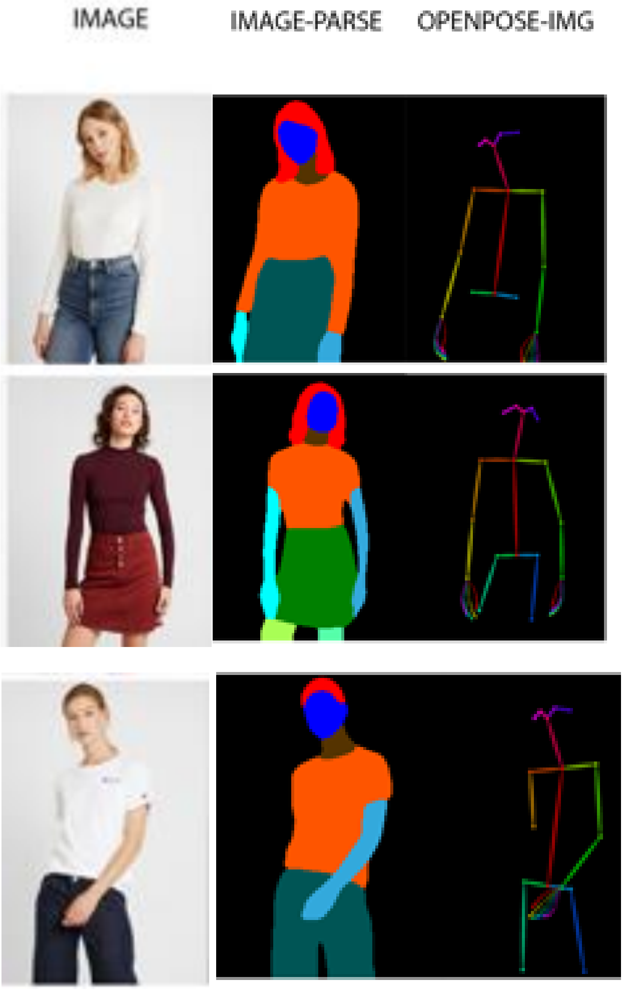


FIG 4.6 Pose map & Segmentation map generation

We used a nonparametric representation known as Part Affinity Fields (PAF's) (Cao, 2016) to efficiently determine the 2D pose of the human figure in the reference image. This has been employed to assimilate the interrelation of human body parts with people in pictures. The design encapsulates universal setting, allowing for a greedy bottom-up parsing step that achieves real-time speed whilst retaining high pose estimate accuracy with state-of-the-art accuracy on many public benchmarks. The said architecture is devised to learn part locations and their associations simultaneously through two divisions of the self-same sequential prediction process.

Part Affinity Fields (PAFs) which are a set of 2D vector fields that record the position and direction of arms and legs over the image domain, are employed to depict the bottom-up representation of association scores. It is demonstrated that concurrently inferring the said bottom-up formulations which are used for recognition as well as connection encode universal setting effectively enough and provide for a greedy parse to produce significant results at a minimum cost. As a result, the open posture map is generated.

The segmentation map S is then used to exclude the original garment portion while the remainder of the image is retained. Based on S and P, this is used to obtain cloth-independent image I_a and a cloth-independent segmentation map S_a. This aids the model in discarding the original cloth information while safeguarding the rest of the image, which would then be handed on to subsequent modules as input to generate the desired results.

4.2.2 TRAINING

The training process starts with segmentation generator G_s. G_s utilizes the apparel-agnostic segmentation map S_a, the pose map P, and the apparel c as keys for the prediction of the segmentation map $^\wedge S$.

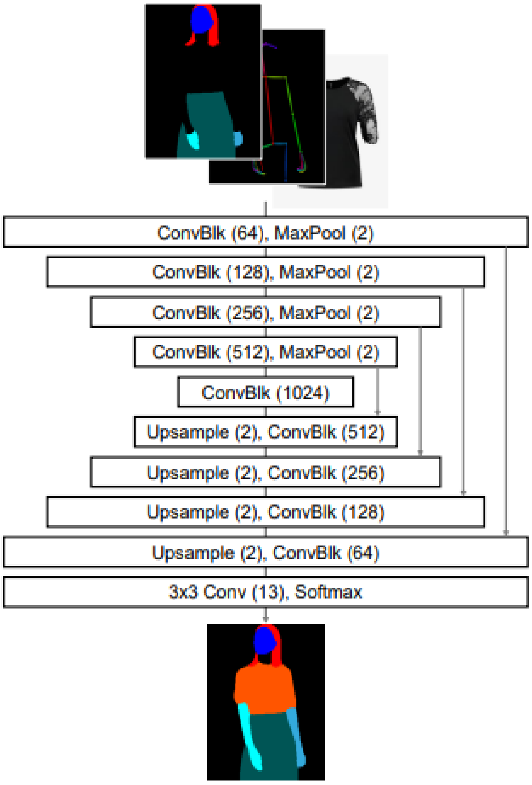

ConvBlk (64), MaxPool (2)
ConvBlk (128), MaxPool (2)
ConvBlk (256), MaxPool (2)
ConvBlk (512), MaxPool (2)
ConvBlk (1024)
Upsample (2), ConvBlk (512)
Upsample (2), ConvBlk (256)
Upsample (2), ConvBlk (128)
Upsample (2), ConvBlk (64)
3x3 Conv (13), Softmax

FIG 4.7 Segmentation generator architecture

To train the segmentation generator, we used the cross-entropy loss and the conditional adversarial loss. Segmentation generator calculates the total loss L_s as follows:

$$\mathcal{L}_S = \mathcal{L}_{cGAN} + \lambda_{CE}\mathcal{L}_{CE}$$

$$\mathcal{L}_{CE} = -\frac{1}{HW}\sum_{k\in C, y\in H, x\in W} S_{k,y,x}\log(\hat{S}_{k,y,x})$$

$$\begin{aligned}\mathcal{L}_{cGAN} = {} & \mathbb{E}_{(X,S)}[\log(D(X,S))] \\ & + \mathbb{E}_X[1 - \log(D(X,\hat{S}))],\end{aligned}$$

Where λ_{CE} is the tuning parameter for the cross-entropy loss. S_{yxk} and $S\hat{}_{yxk}$ are indicating the pixel values of S_a of the source pciture S and $S\hat{}$ relating to the coordinates (x, y) in network k. The characters H, W and C indicate the altitude, thickness, and the number of networks of S. The character X signifies the inputs of the generator (S_a, P, c), and D signifies the discriminator. While training the generator and discriminator, the learning rate used is 0.0004, and the optimizer used is Adam having a batch size of 8. (Choi, 2021)

Next in the training process comes the Geometric Matching Module, which assists in the clothing image deformation process.

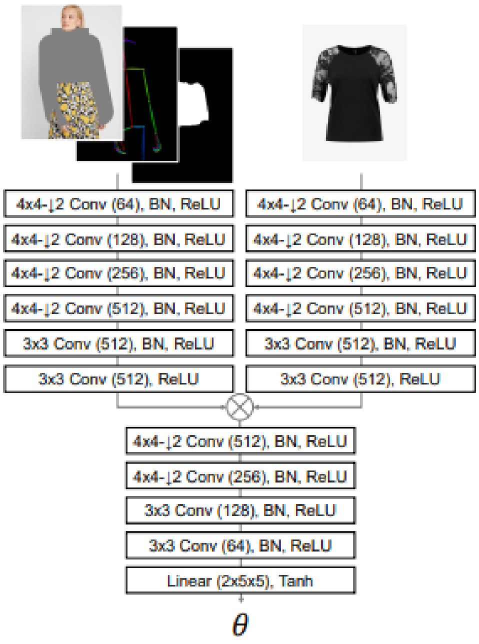

4x4-↓2 Conv (64), BN, ReLU
4x4-↓2 Conv (128), BN, ReLU
4x4-↓2 Conv (256), BN, ReLU
4x4-↓2 Conv (512), BN, ReLU
3x3 Conv (512), BN, ReLU
3x3 Conv (512), ReLU
4x4-↓2 Conv (64), BN, ReLU
4x4-↓2 Conv (128), BN, ReLU
4x4-↓2 Conv (256), BN, ReLU
4x4-↓2 Conv (512), BN, ReLU
3x3 Conv (512), BN, ReLU
3x3 Conv (512), ReLU
4x4-↓2 Conv (512), BN, ReLU
4x4-↓2 Conv (256), BN, ReLU
3x3 Conv (128), BN, ReLU
3x3 Conv (64), BN, ReLU
Linear (2x5x5), Tanh
θ

FIG 4.8 Geometric Matching Module

c, P, clothing-agnostic image I_a, and $\hat{}S_c$, the apparel region of $\hat{}S$ are the inputs of the GMM. The thin-plate spline (TPS) transformation parameter θ is the output produced by the GMM. The main objective of the GMM function is written as follows:

$$\mathcal{L}_{warp} = ||I_c - \mathcal{W}(c, \theta)||_{1,1} + \lambda_{const}\mathcal{L}_{const}$$

$$\mathcal{L}_{const} = \sum_{p \in \mathbf{P}} | (| \, ||pp_0||_2 - ||pp_1||_2| + | \, ||pp_2||_2 - ||pp_3||_2|) \\ +(|\mathcal{S}(p, p_0) - \mathcal{S}(p, p_1)| + |\mathcal{S}(p, p_2) - \mathcal{S}(p, p_3)|),$$

In the above equations, *W* is the work which distorts *c* utilizing *θ* and I_c is the garment extricated from the reference picture *I*. L_{const} and λ_{const} is a second-order difference constraint and hyperparameters for $L_{const,}$ respectively. For the model's training, we have set the λ_{const} to 0.04. The character *p* designates a sampled thin-plate spline control point from the complete control points set *P*, and *p0*, *p1*, *p2*, and *p3* are upper, bottom, left, and right p, respectively. The function *S (p, pi)* signifies the slope between *p* and *pi*. For the GMM, the learning rate was 0.0002. (Choi, 2021)

The final phase presents the ALIAS Generator, which aims to produce the final synthetic picture $^{\wedge}I$ after the application of the conditional normalization method. The ALIAS normalization holds the semantic information.

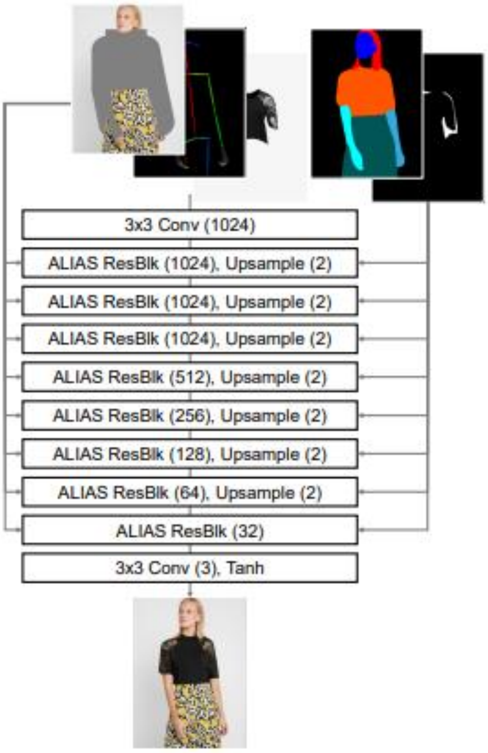

3x3 Conv (1024)
ALIAS ResBlk (1024), Upsample (2)
ALIAS ResBlk (1024), Upsample (2)
ALIAS ResBlk (1024), Upsample (2)
ALIAS ResBlk (512), Upsample (2)
ALIAS ResBlk (256), Upsample (2)
ALIAS ResBlk (128), Upsample (2)
ALIAS ResBlk (64), Upsample (2)
ALIAS ResBlk (32)
3x3 Conv (3), Tanh

FIG 4.9 ALIAS Generator

The segmentation map *S* and the misalignment mask $M_{misalign}$ are delivered to the generator via ALIAS. SPADE (Park, 2019) and pix2pixHD (Choi, 2021) are followed by the loss functions of the ALIAS generator. It is done because it comprises the conditional adversarial loss L_{cGAN}, the feature corresponding loss L_{FM} and the perceptual loss $L_{percept}$. The total loss L_{1} of the generator can be stated as:

$$\mathcal{L}_I = \mathcal{L}_{cGAN} + \lambda_{FM}\mathcal{L}_{FM} + \lambda_{percept}\mathcal{L}_{percept}$$

Where λ_{FM} and $\lambda_{percept}$ are the tuning parameters. The learning rate of the generator and discriminator are 0.0001 and 0.0004, respectively. (Choi, 2021)

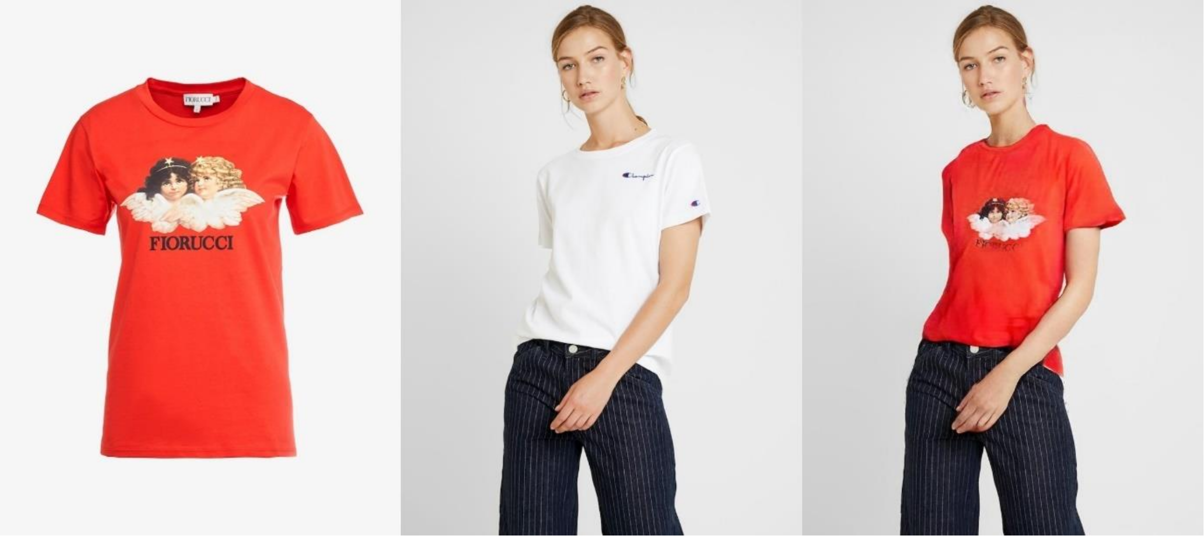


FIG 4.10 Result-1 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image.

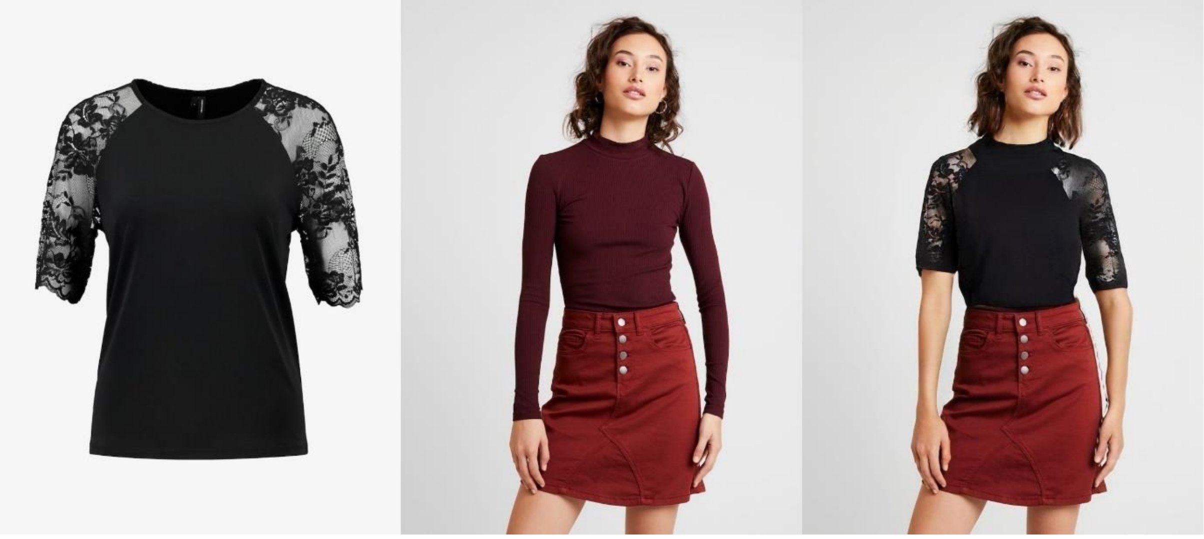


FIG 4.11 Result-2 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image.

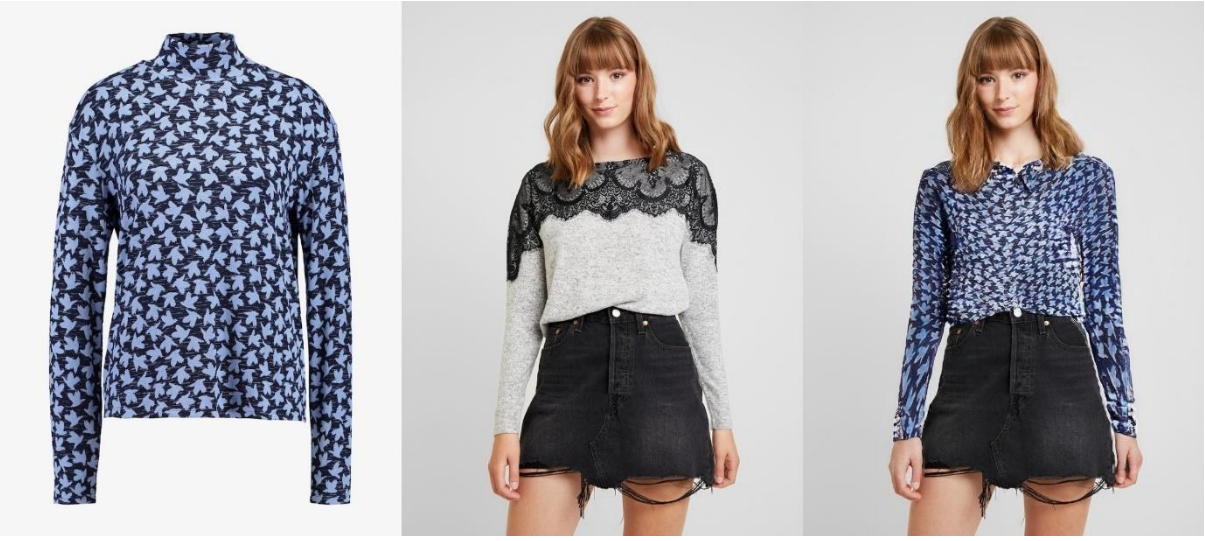


FIG 4.12 Result-3 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image.

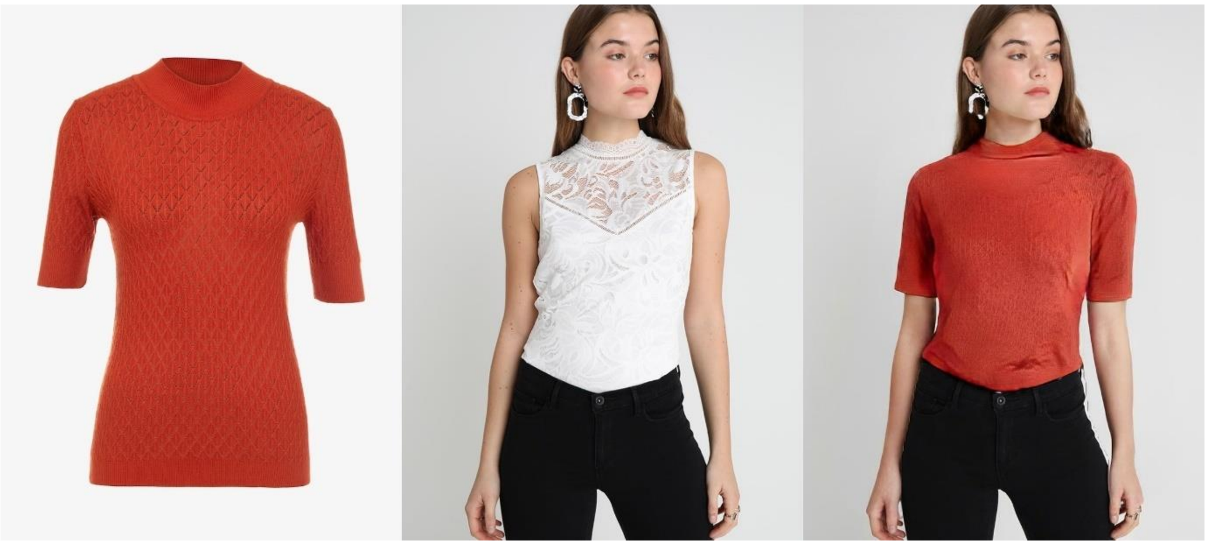
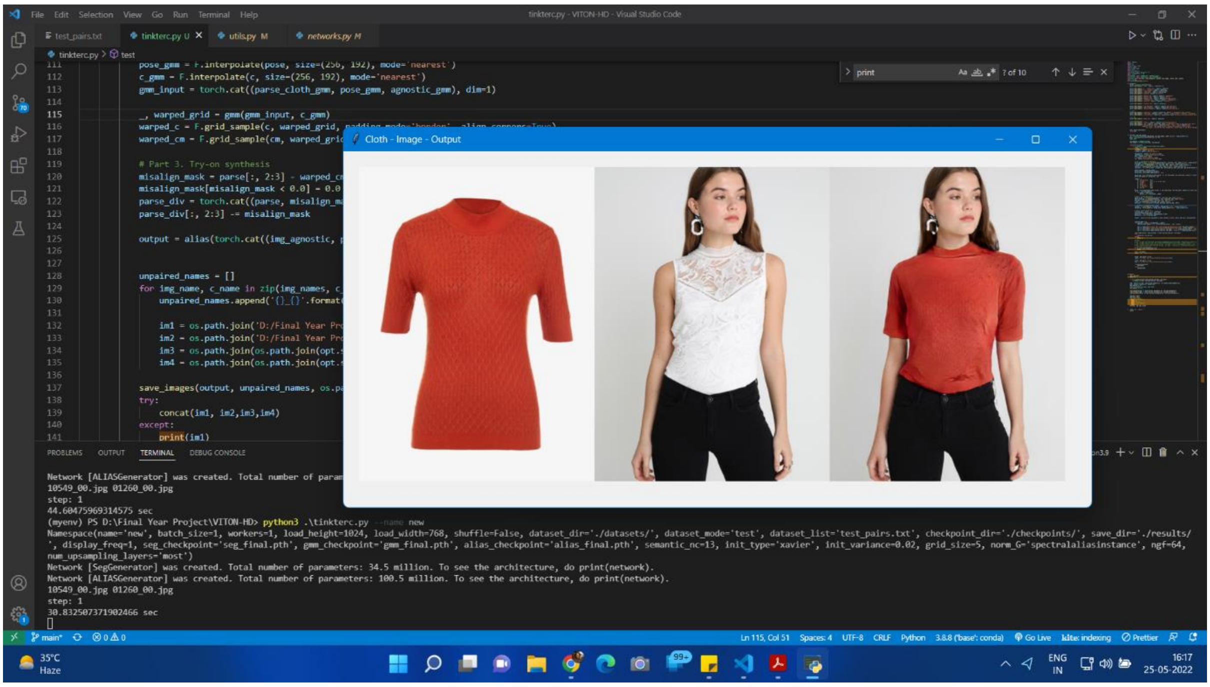


FIG 4.13 Result-4 of Attire Fit-In (Start left) The target apparel, the reference image, and the synthetic image.

To show our results to the customers, we are presenting the output using the *Tkinter* GUI library for python. In the output presented, the leftmost image represents the target clothing item to be put on the picture, the middle image is the reference picture of the customer, and the rightmost image presents the synthetic image with the garment put on.



```
        pose_gmm = F.interpolate(pose, size=(256, 192), mode='nearest')
        c_gmm = F.interpolate(c, size=(256, 192), mode='nearest')
        gmm_input = torch.cat((parse_cloth_gmm, pose_gmm, agnostic_gmm), dim=1)

        _, warped_grid = gmm(gmm_input, c_gmm)
        warped_c = F.grid_sample(c, warped_grid, padding_mode='border', align_corners=True)
        warped_cm = F.grid_sample(cm, warped_gric

        # Part 3. Try-on synthesis
        misalign_mask = parse[:, 2:3] - warped_cr
        misalign_mask[misalign_mask < 0.0] = 0.0
        parse_div = torch.cat((parse, misalign_ma
        parse_div[:, 2:3] -= misalign_mask

        output = alias(torch.cat((img_agnostic, p


        unpaired_names = []
        for img_name, c_name in zip(img_names, c_
            unpaired_names.append('{}_{}'.format(

            im1 = os.path.join('D:/Final Year Pro
            im2 = os.path.join('D:/Final Year Pro
            im3 = os.path.join(os.path.join(opt.s
            im4 = os.path.join(os.path.join(opt.s

        save_images(output, unpaired_names, os.pa
        try:
            concat(im1, im2,im3,im4)
        except:
            print(im1)
```

```
Network [ALIASGenerator] was created. Total number of param
10549_00.jpg 01260_00.jpg
step: 1
44.60475969314575 sec
(myenv) PS D:\Final Year Project\VITON-HD> python3 .\tinkterc.py --name new
Namespace(name='new', batch_size=1, workers=1, load_height=1024, load_width=768, shuffle=False, dataset_dir='./datasets/', dataset_mode='test', dataset_list='test_pairs.txt', checkpoint_dir='./checkpoints/', save_dir='./results/
', display_freq=1, seg_checkpoint='seg_final.pth', gmm_checkpoint='gmm_final.pth', alias_checkpoint='alias_final.pth', semantic_nc=13, init_type='xavier', init_variance=0.02, grid_size=5, norm_G='spectralaliasinstance', ngf=64,
num_upsampling_layers='most')
Network [SegGenerator] was created. Total number of parameters: 34.5 million. To see the architecture, do print(network).
Network [ALIASGenerator] was created. Total number of parameters: 100.5 million. To see the architecture, do print(network).
10549_00.jpg 01260_00.jpg
step: 1
30.832507371902466 sec
```

FIG 4.14 The results as displayed to the user using Tkinter

(Left to Right) The target apparel, the reference image, and the synthetic image

Following we present the output of the implementation of the model on the real-time inputs of the reference images:

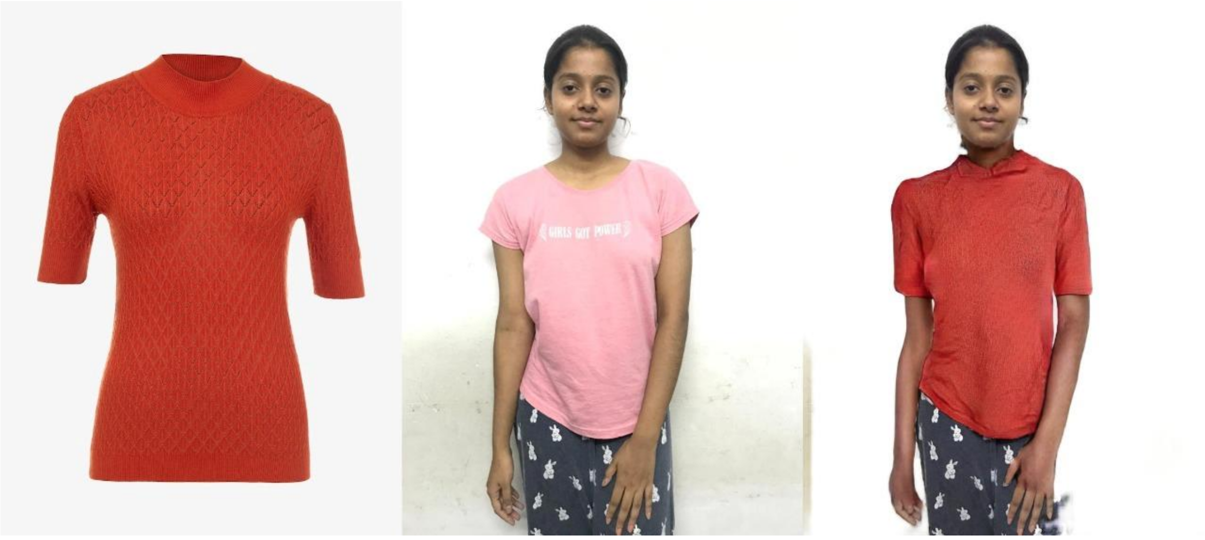


FIG 4.15 Real time implementation – 1

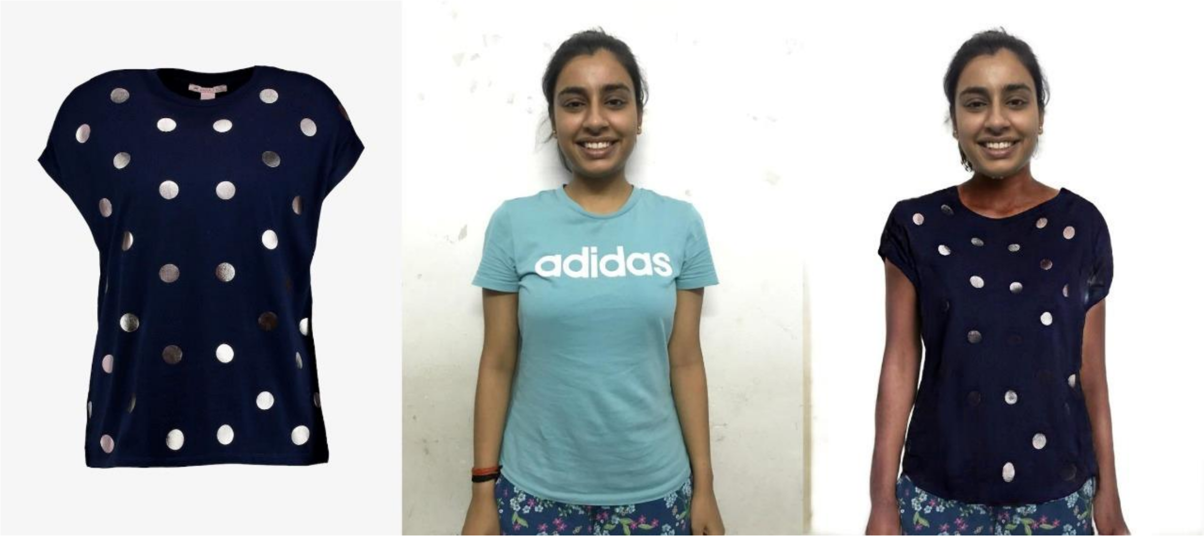


FIG 4.16 Real time implementation - 2

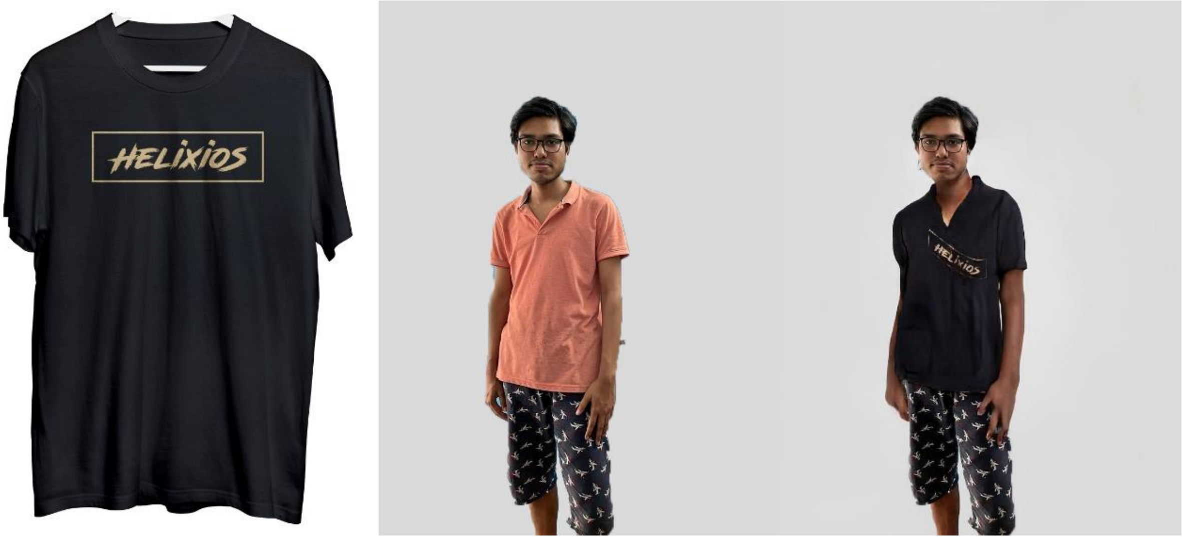


FIG 4.17 Real time implementation - 3

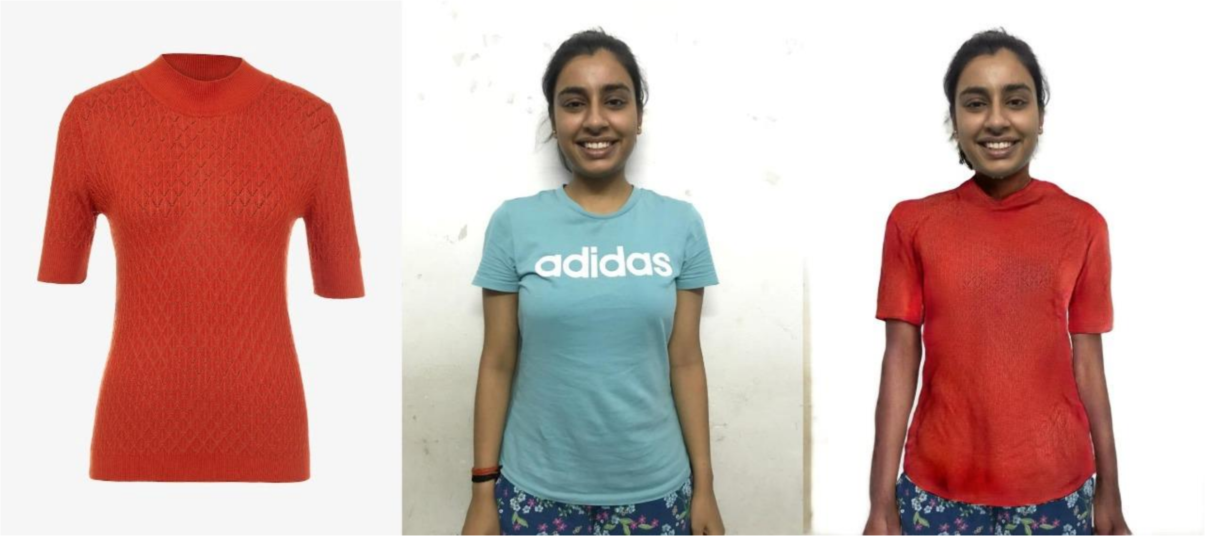


FIG 4.18 Real time implementation - 4

CHAPTER 5

5.1 CONCLUSION

As we have approached the end of the project, we can successfully wrap the images of clothing garments onto the target person's body by deforming it following the client's body shape and posture, thereby aiding the customer to visualize the attire's fitting virtually.

To synthesize the photo-realistically, we have used the Attire Fit-In for the virtual try-on. The ALIAS generator can normalize and process staggered areas, propagate semantic information all through the ALIAS generator, and retain clothing facts through multiscale improvements.

5.2 FUTURE WORKS

We propose this model be fitted in a system and be installed in shopping complexes. It would allow people to capture their pictures through a camera and select the clothing they wish to try from a screen instead of wearing them. This model could replace shopping complexes, as the shops could instead just set up their systems that could showcase all their clothing pieces, and the customer would use it to select their choice of clothing.

REFERENCES

(Ronneberger, 2015) Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.

(Ioffe, 2015) Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448-456). PMLR.

(Ulyanov, 2016) Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2016). Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*.

(De Vries, 2017) De Vries, H., Strub, F., Mary, J., Larochelle, H., Pietquin, O., & Courville, A. C. (2017). Modulating early visual processing by language. *Advances in Neural Information Processing Systems*, *30*.

(Huang, 2017) Huang, X., & Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision* (pp. 1501-1510).

(Park, 2019) Park, T., Liu, M. Y., Wang, T. C., & Zhu, J. Y. (2019). Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2337-2346).

(Zhu, 2020) Zhu, P., Abdal, R., Qin, Y., & Wonka, P. (2020). Sean: Image synthesis with semantic region-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 5104-5113).

(Wang, 2018) Wang, B., Zheng, H., Liang, X., Chen, Y., Lin, L., & Yang, M. (2018). Toward characteristic-preserving image-based virtual try-on network. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 589-604).

(Guan, 2012) Guan, P., Reiss, L., Hirshberg, D. A., Weiss, A., & Black, M. J. (2012). Drape: Dressing any person. *ACM Transactions on Graphics (TOG)*, *31*(4), 1-10.

(Sekine, 2014) Sekine, M., Sugita, K., Perbet, F., Stenger, B., & Nishiyama, M. (2014, October). Virtual fitting by single-shot body shape estimation. In *Int. Conf. on 3D Body Scanning Technologies* (pp. 406-413). Citeseer.

(Pons-Moll, 2017) Pons-Moll, G., Pujades, S., Hu, S., & Black, M. J. (2017). ClothCap: Seamless 4D clothing capture and retargeting. *ACM Transactions on Graphics (ToG)*, *36*(4), 1-15.

(Patel, 2020) Patel, C., Liao, Z., & Pons-Moll, G. (2020). Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 7365-7375).

(Han, 2018) Han, X., Wu, Z., Wu, Z., Yu, R., & Davis, L. S. (2018). Viton: An image-based virtual try-on network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7543-7552).

(Han X. H., 2019) Han, X., Hu, X., Huang, W., & Scott, M. R. (2019). Clothflow: A flow-based model for clothed person generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 10471-10480).

(Yu, 2019) Yu, R., Wang, X., & Xie, X. (2019). Vtnfp: An image-based virtual try-on network with body and clothing feature preservation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 10511-10520).

(Yang, 2020) Yang, H., Zhang, R., Guo, X., Liu, W., Zuo, W., & Luo, P. (2020). Towards photo-realistic virtual try-on by adaptively generating-preserving image content. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 7850-7859).

(Dong, 2019) Dong, H., Liang, X., Shen, X., Wang, B., Lai, H., Zhu, J., ... & Yin, J. (2019). Towards multi-pose guided virtual try-on network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 9026-9035).

(Jetchev, 2017) Jetchev, N., & Bergmann, U. (2017). The conditional analogy gan: Swapping fashion articles on people images. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (pp. 2287-2292).

(Choi, 2021) Choi, S., Park, S., Lee, M., & Choo, J. (2021). Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 14131-14140).

(Wang T. C., 2018) Wang, T. C., Liu, M. Y., Zhu, J. Y., Tao, A., Kautz, J., & Catanzaro, B. (2018). High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8798-8807).

(Gong, 2018) Gong, K., Liang, X., Li, Y., Chen, Y., Yang, M., & Lin, L. (2018). Instance-level human parsing via part grouping network. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 770-785).

(Cao, 2016) Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017). Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7291-7299).

(Wang Z. B., 2004) Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, *13*(4), 600-612.