# SVTON: Simplified Virtual Try-On

Tasin Islam
*Dept. of Computer Science*
*Brunel University London*
London, UK
tasin.islam2@brunel.ac.uk

Alina Miron
*Dept. of Computer Science*
*Brunel University London*
London, UK
alina.miron@brunel.ac.uk

XiaoHui Liu
*Dept. of Computer Science*
*Brunel University London*
London, UK
xiaohui.liu@brunel.ac.uk

Yongmin Li
*Dept. of Computer Science*
*Brunel University London*
London, UK
yongmin.li@brunel.ac.uk

***Abstract*—2D based Virtual Try-On (VTON) has been trending towards using human parsing to improve the quality of the try-on image. However, it remains a challenging problem for most existing VTON models to generate realistic images for situations with unpaired candidate-clothing images and body-part occlusions. We have developed a Simplified Virtual Try-On (SVTON) model to rectify the above problem. The SVTON uses refined input data to produce accurate labels and has fewer trainable parameters than existing methods. Also, it is designed with a simplified network architecture for segmentation and an efficient Affine Transform for warping to target clothing. Experiments on benchmark datasets show that the proposed model performs better than the state-of-the-art VTON models for unpaired and occlusion cases, while maintaining the similar overall performance level for normal cases.**

***Index Terms*—Virtual Try-on (VTON), Generative Adversarial Network (GAN), U-Net, Segmentation, Affine Transform**

## I. Introduction

The 2D based Virtual Try-On (VTON) models attempt to synthesise an image of a person wearing the desired clothing. VTON can benefit consumers who shop clothing items online, providing insight into how the garment may look before purchasing improving customer satisfaction. Though significant progress has been made [1]–[3], much still needs to be done to make the synthesised images more genuine and photo-realistic. For example, older VTON models struggle to synthesise a person wearing a long-sleeved garment into the short-sleeved target clothing because it is challenging to generate high-quality arms and hands. More recent VTON models utilise the segmentation module for preserving details from the candidate image [3], [4], but still, fail to produce accurate segment labels. Poor segmentation performance leads to severe problems because it affects the performance of the subsequent modules, and the try-on image will show the candidate wearing the target clothing incorrectly. To address the above problems, we have developed a novel Simplified Virtual Try-On (SVTON) model in this work. Fig. 1 shows how our model is more consistent in applying the target clothing correctly and realistically than existing models (VITON [1], CP-VTON+ [5] and ACGPN [3]), which have failed to generate realistic images under situations of short-sleeves to long-sleeves, occluded body parts and occluded clothing, for example.

The proposed model uses the Predictive Human Parsing Module (PHPM) first. The PHPM uses the binary mask of the

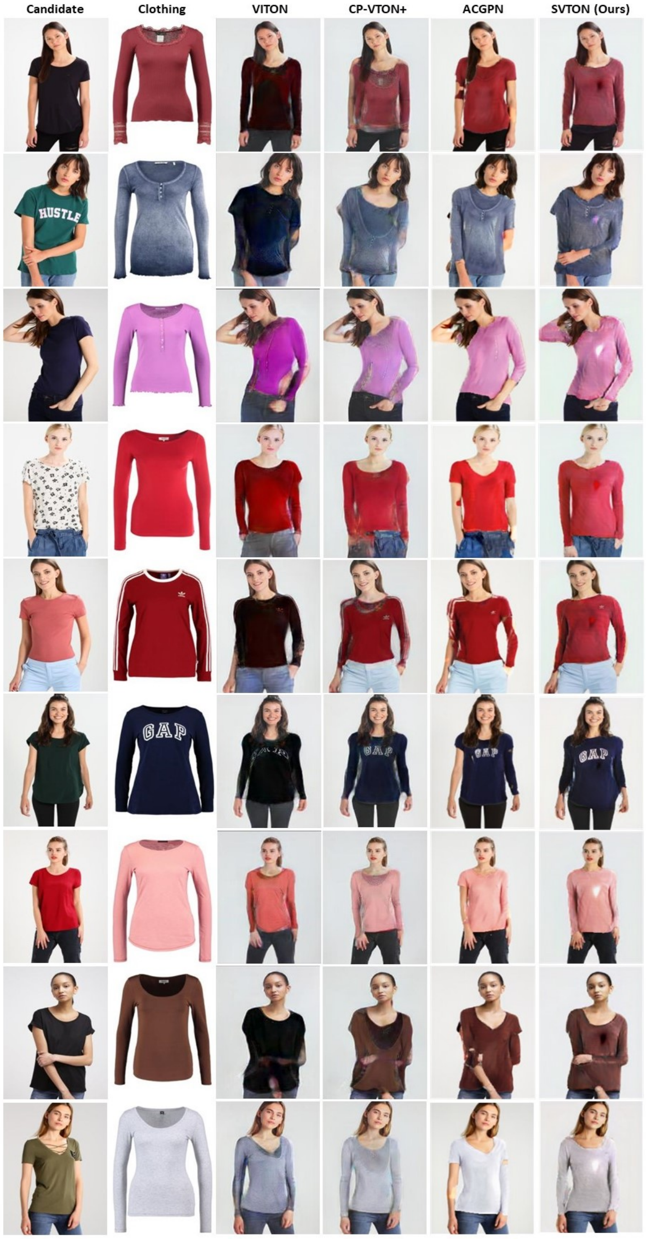

Fig. 1. Comparison of our model against VITON, CP-VTON+ and ACGPN. The proposed model performs better than the previous models, especially for cases with unpaired candidate-clothing (e.g. short-sleeves to long-sleeves) and occlusions of body parts.

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICMLA2022.XXXXXXX, 2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)

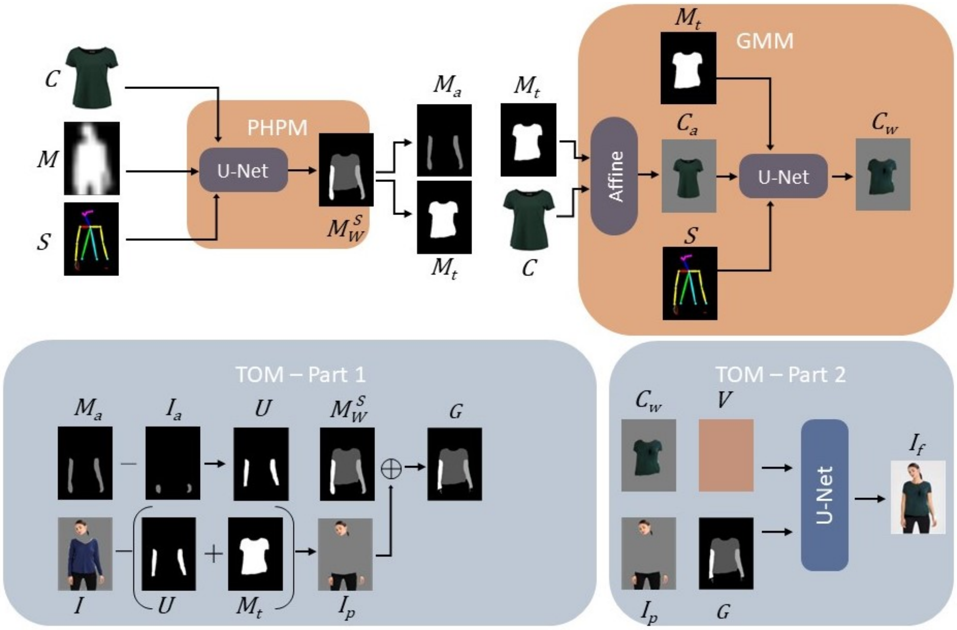

Fig. 2. An overview of the network architecture of our SVTON, which is comprised of three modules of PHPM, GMM and TOM.

candidate image and clothes to predict the label for the arm and torso. The next step is for the Geometric Matching Module (GMM) to warp the garment to fit inside the torso label generated by PHPM. The GMM uses an Affine Transform to guide the neural network about positioning texture, logo, and pattern on the warped clothing. Lastly, the Try-On Module (TOM) will merge the images produced by PHPM and GMM into a final try-on image. PHPM output will help TOM decide what it can preserve from the candidate image and where it may need to generate the arms in the try-on image.

The novel contributions of the work are as follows: 1) A new method to generate segmentation for the relevant body parts by refining the input data from both candidate and clothing images and designing a different network architecture to perform the segmentation, 2) an Affine Transform to assist the neural network responsible for warping onto the target clothing.

The rest of the paper is organised as follow. Section II reviews the relevant previous studies on this topic. The proposed model SVTON is described in Section III. Experimental results and analysis are provided in Section IV. Finally conclusions are drawn in Section V. The source code of this paper can be found at https://github.com/1702609/SVTON.

## II. Background

Generative Adversarial Networks (GAN) has made a significant breakthrough in image synthesis and generation [6], [7]. Goodfellow trained two neural networks adversarially that allows the generator to produce data that resemble the dataset [8]. Conditional Generative Adversarial Network (cGAN) [9] has shown how the neural network takes images to influence the outcome. VTON depends on conditions (i.e. an image of the person and clothes), making cGAN valuable to VTON.

The development of VTON started by using 3D measurements of a person's body to fit the target clothing onto the person. Drape [10] and Sekine et al. [11] utilise a 2D image of clothing and 3D information of a person or avatar to synthesise the try-on image. 3D based VTON is not suitable for online scenarios because it is difficult for consumers to provide 3D information about their body shape.

The first 2D approach of VTON can be traced back to 2017 when Jetchev and Bergmann proposed CAGAN [12] to swap a person's original clothes with the target clothes. However, the model requires both the target and original clothing to change during testing, making it infeasible in practical scenarios. VITON [1] uses Thin-Plate Spline (TPS) to warp the garment and merges it with the coarse body shape of the person to generate the try-on image. CP-VTON [2] improves the TPS performance by using a neural network to predict TPS parameters rather than directly relying on images. It is common for VTON to suffer from body-part occlusions such as the arms not being preserved. VITON-GAN [13] has used a similar model of CP-VTON but trained with a discriminator to solve the occlusion problem slightly. Newer VTON models generate body labels that suit the target clothing and perform better in occlusion cases. For example, SwapNet [14] and VTNFP [4] have shown that the segment provides guidance on the alignment of the target clothes and allows for better preservation of the person's body shape and pose and generation of body features. Generated labels allow VTON to preserve complex body parts such as the hand, which increases the quality of the try-on image, as demonstrated by ACGPN [3] and VITON-HD [15]. VITON-HD focuses on the misalignment of the warped garment. They argue that geometrical transformation (such as TPS and Affine Transformation) can never align with a person's body; therefore, the misalignment region needs to be generated. Many researchers have turned away from using 3D-based methods due to practical issues. CloTH-VTON [16] utilises the advantages of 2D and 3D-based image synthesis. Their approach warps a 3D garment model that provides more realistic deformation than 2D into the person. CloTH-VTON uses the 2D method for generating or preserving body parts. Similarly, M3D-VTON [17] also utilise the benefits of 2D and 3D approaches. Their method uses 2D image-based virtual try-on and then makes inferences to create a 3D person wearing the desired clothes. Despite the significant development in this area, VTON remains a challenging problem, especially for cases with unpaired candidate-clothing and body-part occlusion.

## III. Method

The proposed SVTON model is inspired by ACGPN [3]. Fig. 2 shows the overall architecture of SVTON. There are three modules: the Predictive Human Parsing Module (PHPM), Geometric Matching Module (GMM), and the Try-On Module (TOM). We have adapted the original U-Net [18] for efficiency purposes.

### *A. Predictive Human Parsing Module (PHPM)*

PHPM analyses the target clothing to generate the torso and arms segment. The clothing image $C$, candidate's mask $M$ and RGB pose skeleton $S$ are the input data for the PHPM. $M$ will be blurred to help PHPM lose the coarse body shape of the original clothing and allow the U-Net to generate an appropriate label for the torso that complements $C$. Fig. 3 illustrates the difference when using regular or blurred M. The binary mask is shrunk by a factor of 16 and then resized back to the original dimension, giving the effect of the boundaries being blurry. PHPM should generate a 4-channel output $M_W^S$ showing the distinct segment of the torso and arms. The VTON dataset has included the original segmentation label $M_{\mathrm{gt}}$ for every body part, which we have extracted the torso and arms from them and used as the ground truth when training PHPM. Individual segments from $M_W^S$ can be extracted further as torso $M_t$ and arms $M_{\mathrm{ra}}$, $M_{\mathrm{la}}$. $W$ is depicted as ($W$ = $b$, $t$, $ra$, $la$ ($b$:background, $t$:torso, $ra$:right arm, $la$:left arm)).

Cross-entropy loss is useful for neural networks that predict probabilities for multiple classes [19]. PHPM generates four labels, and cross-entropy can calculate how well each segment matches the ground truth. The loss function for PHPM is formulated as $L_{\mathrm{PHPM}}$:

$$L_{\mathrm{PHPM}} = \lambda_1 L_{\mathrm{entropy}} \quad (1)$$

where $L_{\mathrm{entropy}}$ is the cross-entropy loss [19], and $\lambda_1$ is the parameter to magnify the loss.

### *B. Geometric Matching Module (GMM)*

We introduce Affine Transform in our Geometric Matching Module (GMM). Our Spatial Transformation Network (STN) utilises Affine Transform to align $C$ with $M_t$. STN acts as a preliminary stage since its purpose is to guide the subsequent U-Net about the positioning of the clothing and its vital characteristics like the logo, texture and embroidery. RGB pose skeleton shows how the poses are connected and help GMM differentiate the torso and the arm in occlusion cases. We have illustrated this in Fig. 4 where it clearly shows the advantages of using RGB pose skeleton over individual pose map. Our experiment shows that Affine Transform helps retain clothing detail when warped. Unlike in previous methods, we oppose using Thin-Plate Spline (TPS) because they have more trainable parameters, making them difficult to train. Though TPS offers a higher degree of freedom of shape deformation, Fig. 5 shows that extra shape deformation does not improve the performance of U-Net. The transformed garment $C_w$ will go through the U-Net to match $M_t$. The $S$ and $M_t$ are auxiliary data inputted into the U-Net.

A discriminator is used to train the GMM. We use the cGAN

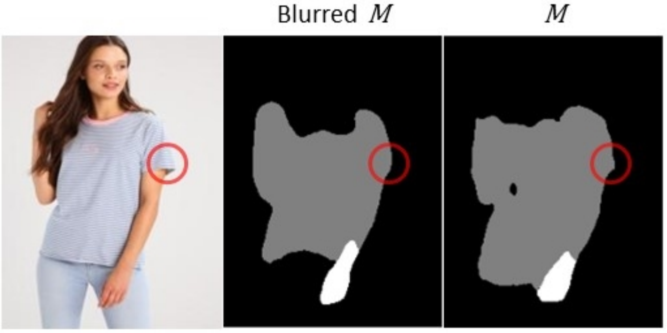

Fig. 3. Blurring the mask helps PHPM to remove the torso label's clothing shape. Without blurring, the segments may capture the undesired shape from the original clothing (e.g. the bulge at the sleeve shown here) and render the final generated images unrealistic.

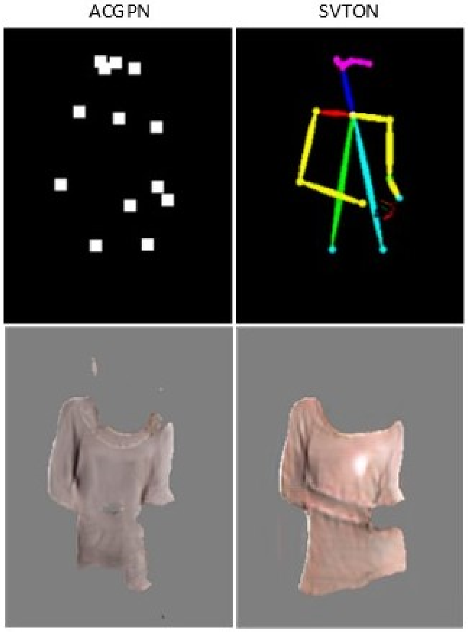

Fig. 4. RGB skeleton shows how joints are connected and make it easy for the U-Net to distinguish between the arm and torso in occlusion cases.

This article has been accepted for publication in a future proceedings of this conference, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/ICMLA2022.XXXXXXX, 2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)

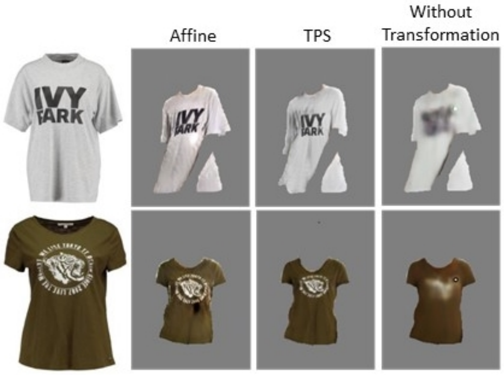

Fig. 5. Comparison of warping results from different methods. The difference between TPS and Affine Transform is insignificant, but the model complexity of Affine is much lower, and therefore it is easier to train.

loss [9], which they formulated as:

$$L_{\mathrm{GAN}}(x, y) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_x[\log(1 - D(x, G(x)))] \tag{2}$$

where $x$ represents data fed into the generator and $y$ is the ground truth.

To calculate the loss of GMM, we utilise the L1 and VGG loss functions. We formulate the loss function as $L_{\mathrm{GMM}}$:

$$L_1(x, y) = |x - y| \tag{3}$$

$$L_{\mathrm{VGG}}(x, y) = \lambda_2 |\phi_5(x) - \phi_5(y)| \tag{4}$$

$$L_{\mathrm{GMM}} = L_1(x, y) + L_1(\hat{x}, y) + L_{\mathrm{VGG}}(x, y) + L_{\mathrm{GAN}}(f, y) \tag{5}$$

where $x$, $\hat{x}$, $y$, and $f$ denotes $C_w$, $C_a$, the ground truth of the warped garment $C_{\mathrm{gt}}$ and data we input to GMM. $L_{\mathrm{VGG}}$ is the VGG perceptual loss [20] in which $\phi$ represents the feature map of $C_w$ and $C_{\mathrm{gt}}$ from the pre-trained VGG19 model. We use the 5th layer of the VGG network. Lambda is a parameter to control the loss value.

### C. Try-On Module (TOM)

TOM merges the images generated from the previous module into a final try-on image $I_f$. The U-Net expects an input of $C_w$, $G$, preserved body part $I_p$ and average skin colour $V$. $I_p$ contains the head, hair, bottom clothes and the preservable region of the arm. We generate $I_p$ the same way as in ACGPN [3]. TOM creates $U$ by performing element-wise subtraction on $M_a$ with the original arm label $I_a$. We perform element-wise subtraction on $I$ by $U$ and $M_t$ to preserve the desired region $I_p$. It is essential to remove the hand region from $M_W^S$ because we trained the U-Net to generate the arm only if the arm label is present. To produce a handless segment $G$, we perform element-wise multiplication between $M_W^S$ and $I_p$.

TOM uses the VGG and L1 loss functions to train, and we formulate the loss as:

$$L_{\mathrm{TOM}} = L_1(x, y) + L_{\mathrm{VGG}}(x, y) + L_{\mathrm{GAN}}(f, y) \tag{6}$$

where $x$, $y$, and $f$ denotes $I_f$, $I$ and data we input to TOM.

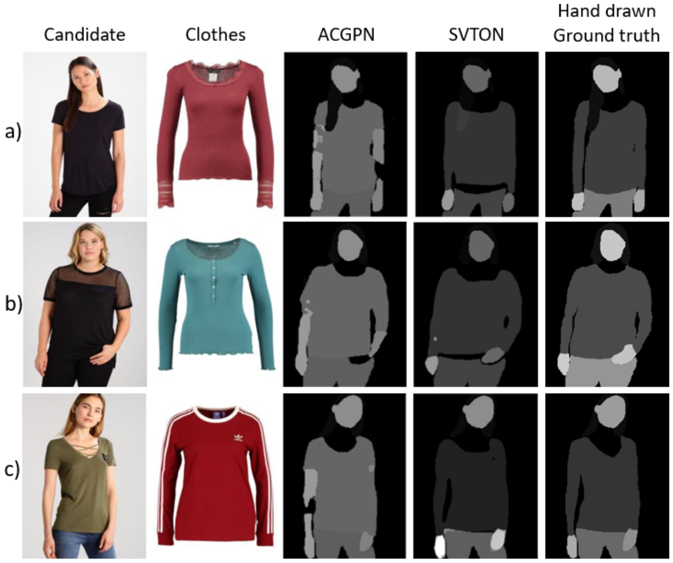

Fig. 6. Apparent differences are shown between ACGPN and SVTON. ACGPN generates labels for short-sleeved to long-sleeved poorly.

## IV. Experiments

### A. Dataset

We trained our model using the VTON dataset [1]. The dataset consists of 12,821 candidate-clothing pairs as the training set, 1400 image pairs for the validation and a further 2032 image pairs for the testing set. The resolution of the images is 256 x 192. The dataset consists of the candidate images, paired clothing images, binary masks, segmentation and RGB Skeleton.

### B. Implementation

U-Net architecture [18] is adopted across all the three modules of PHPM, GMM and TOM, but with fewer convolutional layers in the former two.

The U-Net of PHPM and GMM has seven convolutional layers with a kernel size of 3, and their respective number of filters are 64, 128, 256, 512, 512, 1024, 1024 for their encoder. The decoder has ten convolutional layers with a kernel size of 3, and their respective number of filters are 512, 512, 512, 256, 256, 128, 128, 64, 64, 3. We used skip connections in this U-Net. The additional STN used by GMM has the same architecture as in [21]. TOM has the same architecture as the original U-Net. The discriminator has four convolutional layers with a kernel size of 4, and their respective number of filters are 64, 128, 256, 1 with sigmoid function in the end.

We have trained the modules separately to assess how the individual modules are performing. We trained PHPM and GMM for 20 epochs and TOM for 100 epochs in a paired setting, which means SVTON trained to synthesise candidate images with their original clothes. We used the Adam optimiser to optimise the U-Nets with a hyperparameter of 0.0002 for the learning rate and set $\beta_1$ = 0.5 and $\beta_2$ = 0.999.

Testing has the same procedure as training, but the target clothing can differ from what the candidate initially wears.

| | | ACGPN | | SVTON | |
|---|---|---|---|---|---|
| | | Left Arm | Right Arm | Left Arm | Right Arm |
| a) | Dice | 45.3% | 47.5% | 91.0% | 86.7% |
| | IoU | 63.5% | 64.4% | 91.6% | 88.2% |
| b) | Dice | 49.4% | 8.6% | 90.7% | 81.3% |
| | IoU | 65.4% | 51.1% | 91.4% | 84.0% |
| c) | Dice | 42.5% | 82.6% | 90.9% | 96.9% |
| | IoU | 62.2% | 85.0% | 91.6% | 95.7% |

TABLE I

COMPARISON OF DICE AND IOU SCORES BETWEEN THE ACGPN AND THE PROPOSED MODEL SVTON ON A SUBSET OF THE DATA WITH UNPARIED SETTINGS. THE SVTON SCORES ARE HIGHER THAN ACGPN FOR BOTH METRICS OF DICE AND IOU.

### C. Qualitative Analysis

We have compared our method with VITON [1], CP-VTON+ [5] and ACGPN [3], as shown in Fig. 1. VITON only preserves the face and the hair when training or evaluating the try-on network. Providing insufficient data to the neural network causes the synthesised image to show changes in undesired regions like VITON changing the colour of the trouser. With CP-VTON+, ACGPN and SVTON, we included the bottom clothes when feeding data into the neural network, stopping the trouser from being modified. VITON and CP-VTON+ have generated low-quality arms and hands because they do not have a method that guides image generation like human parsing. ACGPN tends to struggle to generate appropriate labels when changing a candidate wearing a short-sleeved garment into a long-sleeved one. ACGPN forces the long-sleeved to fit inside the incorrectly generated torso label; therefore, their try-on is incorrect. Our method performs better than ACGPN when applying the clothing item onto the person.

Fig. 7 shows a common problem where PHPM produces the incorrect size of the torso dimension. This is because the blurred $M$ does not clearly distinguish between the torso and legs. The subsequent modules will be affected, producing unnatural colouring in the gaps or, in rare cases, it will render blanks, as seen on the bottom example of the figure. VITON-HD [15] provides a distinct label of the head and legs to the segmentation module, which will fix the torso issue.

### D. Quantitative Analysis

We have used the Dice coefficient and IoU to compare our segmentation performance against ACGPN [22]. We used the testing set (paired setting) to show that ACGPN has beaten our model around 10% for Dice and 8% for IoU when generating the segment for the arm. When we experimented on three handcrafted segmentation labels of unpaired settings, we showed that our model outperformed ACGPN significantly. Fig. 6 shows two examples where ACGPN has seriously failed to produce the correct segment for both hands/arms. We include an extreme case in Fig. 6b where ACGPN has failed to generate the label for the right hand and scored only 8.6% for the Dice coefficient, as shown in Table 1.

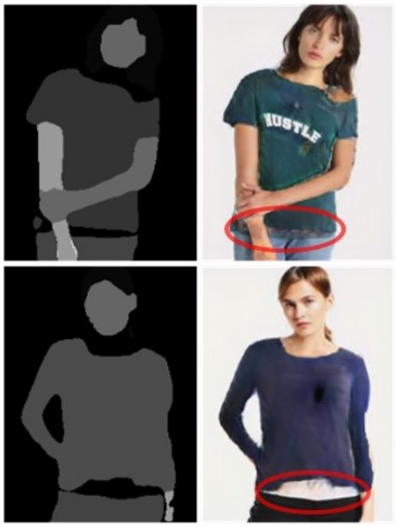

Fig. 7. The short label of the torso causes the U-Net to produce undesirable try-on.

### E. Discussions

The proposed model performs better than the previous models, mainly for two novel contributions. First, it generates the segment labels of the arms more accurately on unpaired settings. We developed a new segmentation module using different input data and network architecture. The results have shown that our model performs better at synthesising clothes with any clothing. The ACGPN takes a different approach since they differentiate a single-labelled image into multiple body parts, producing inaccurate results even when training their segmentation module for longer. Our work showed that the segmentation module needs to produce the correct label; otherwise, the subsequent modules will warp the garment wrongly, and the try-on will be incorrect.

Second, the proposed model offers a more efficient method for warping the garment. Previous VTON models [1], [2], [4], [3] utilise TPS transformation to warp the garment. There are disadvantages to using TPS, such as, in some difficult cases, TPS can distort the texture of the clothing. ACGPN rectified the problem by introducing a constraint to TPS to stabilise the warping module. However, we showed that using an even simpler approach, such as Affine transformation, can perform just as well as TPS and is more efficient.

## V. CONCLUSIONS

In this paper, we have presented the SVTON, a new Virtual Try-On model, to address the challenging problem of generating realistic images for unpaired candidate-clothing images and body-part occlusions. The model is comprised of three modules of PHPM, GMM and TOM, where the U-Net architecture and generative models are adopted. The proposed model differs from the previous models in that refined input data are used, and different network architecture is adopted. Also, the warping to target clothing is based on an Affine Transform which is more computationally efficient. We have experimented on the benchmark VTON dataset [1]. Improved results have been demonstrated over a number of previous models such as VITON [1], CP-VTON+ [5], and ACGPN [3], especially under situations with unpaired candidate-clothing settings and significant body-part occlusions.

## References

[1] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis, "Viton: An image-based virtual try-on network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7543–7552.

[2] Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, Liang Lin, and Meng Yang, "Toward characteristic-preserving image-based virtual try-on network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 589–604.

[3] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7850–7859.

[4] Ruiyun Yu, Xiaoqi Wang, and Xiaohui Xie, "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*, 2019, pp. 10511–10520.

[5] Matiur Rahman Minar, Thai Thanh Tuan, Heejune Ahn, Paul Rosin, and Yu-Kun Lai, "Cp-vton+: Clothing shape and texture preserving image-based virtual try-on," in *CVPR Workshops*, 2020.

[6] Tero Karras, Samuli Laine, and Timo Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[7] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[9] Mehdi Mirza and Simon Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[10] Peng Guan, Loretta Reiss, David A Hirshberg, Alexander Weiss, and Michael J Black, "Drape: Dressing any person," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, pp. 1–10, 2012.

[11] Masahiro Sekine, Kaoru Sugita, Frank Perbet, Björn Stenger, and Masashi Nishiyama, "Virtual fitting by single-shot body shape estimation," in *Int. Conf. on 3D Body Scanning Technologies*. Citeseer, 2014, pp. 406–413.

[12] Nikolay Jetchev and Urs Bergmann, "The conditional analogy gan: Swapping fashion articles on people images," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 2287–2292.

[13] Shion Honda, "Viton-gan: Virtual try-on image generator trained with adversarial loss," *arXiv preprint arXiv:1911.07926*, 2019.

[14] Amit Raj, Patsorn Sangkloy, Huiwen Chang, Jingwan Lu, Duygu Ceylan, and James Hays, "Swapnet: Garment transfer in single view images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 666–682.

[15] "Viton-hd: High-resolution virtual try-on via misalignment-aware normalization," 3 2021.

[16] Matiur Rahman Minar and Heejune Ahn, "Cloth-vton: Clothing three-dimensional reconstruction for hybrid image-based virtual try-on," in *Asian Conference on Computer Vision (ACCV)*, 2020.

[17] Fuwei Zhao, Zhenyu Xie, Michael Kampffmeyer, Haoye Dong, Songfang Han, Tianxiang Zheng, Tao Zhang, and Xiaodan Liang, "M3d-vton: A monocular-to-3d virtual try-on network," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 13239–13249.

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[19] Kevin P Murphy, *Machine learning: a probabilistic perspective*, MIT press, 2012.

[20] Justin Johnson, Alexandre Alahi, and Li Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*. Springer, 2016, pp. 694–711.

[21] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al., "Spatial transformer networks," *Advances in neural information processing systems*, vol. 28, 2015.

[22] Abdel Aziz Taha and Allan Hanbury, "Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool," *BMC medical imaging*, vol. 15, no. 1, pp. 1–28, 2015.