UVTON: UV Mapping to Consider the 3D Structure of a Human in Image-Based Virtual Try-On Network

Shizuma Kubo, Yusuke Iwasawa, Masahiro Suzuki, Yutaka Matsuo
The University of Tokyo
{kubo,iwasawa,masa,matsuo}@weblab.t.u-tokyo.ac.jp

Abstract

Image-based virtual try-on is an area of research that is attracting attention as the demand for online apparel shopping continues to increase. The methods proposed thus far have focused on how to generate a dress-up image while preserving the clothing details. However, the posture of the model in the image is limited to an upright position, and other positions frequently do not work well. In this study, based on a kind of generative adversarial network (GAN) that utilizes UV mapping to consider the 3D structure of the human body, we propose a novel virtual try-on method called a UV Try-On Network (UVTON). We use a DensePose to estimate a point corresponding to the 3D surface of a human model for each pixel point of a 2D image and incorporate the estimated information into our model. It is thus possible to change the clothes of users holding various postures. Our proposed method uses UV mapping and two other modules. One module generates parts to be used in the mapping, and the other refines the image and produces a more realistic image. Based on both qualitative and quantitative comparison with existing methods, we experimentally demonstrated that our method achieved better results with various postures.

1. Introduction

Online apparel shopping is becoming increasingly popular owing to its convenience, and virtual try-on has attracted attention as a way for customers to determine whether clothes will suit them before purchase. One of the approaches for virtual try-on is an image-based virtual try-on system. The system generates a new image of a customer complete with new clothes from two images, one of the customer and one of the new clothes.

CAGAN[8] adopts image-to-image translation [7, 16] techniques for image-based virtual try-on. This method successfully transforms information like color and texture but struggles to incorporate geometric changes like the shape of clothes [16]. To tackle this problem, VITON[4] and CP-VTON[15] introduced methods to convert clothes to the shape of the target person. These methods can preserve details well in changing clothes, which is an essential factor to determine whether the clothes fit or not. However, the problem remains that the person in the image is limited to the upright posture when changing clothes. As illustrated in Figure 1, CAGAN and CP-VTON do not work well when the posture of the person in the image is far from upright.

In this paper, we propose a novel image-based virtual try-on method, called a UV Try-On Network (UVTON), which utilizes UV mapping[1] to avoid the aforementioned issue. Our model aligns the target clothes and target person over UV representation space, not on a 2D image space. Because UV representation can consider the 3D structure of the body, this approach helps to preserve body structure even for the complex posture of the human. In our experiments with the dataset collected from Zalando[2], we show the effectiveness of our model in changing the clothes on people in various postures both quantitatively and qualitatively.

2. Method

Our proposed method consists of a mapping stream that utilizes UV mapping and two assistive modules. An overview of the model is shown in Figure 2. Through a *texture mapping stream*, points are mapped to the corresponding points on the target person along with its IUV[3] data obtained by DensePose [3]. The *painting module* generates images of the parts necessary for mapping, and the *refine module* refines the images after mapping.

We use datasets of $\{c_i, h_i\}_{i=1}^{N}$, where the index i indicates the pairing of a human and a set of clothes. The image h_i represents the image of a person wearing the clothes c_i.

[1] UV mapping is a technique used in 3D modelling, which projects a 2D image onto the surface of a 3D model for texture mapping.

[2] A fashion E-Commerce site (https://www.zalando.de).

[3] IUV contains UV coordinate information when mapping textures to a 3D model, and information regarding which part of the body each UV coordinate information belongs to.

Figure 1. Comparison with existing methods. We compared the generated results of our proposed method (UVTON) with the existing methods (CAGAN, CP-VTON). The leftmost column shows images of the person, and the top line shows images of the clothes to be worn. From top to bottom, the AC value increases. When you look at the image at the bottom, the posture of the person in the image is far from upright. In the image of the clothes, the TV norm increases from left to right.

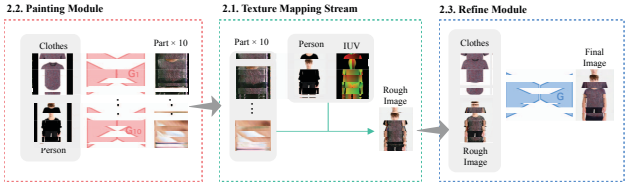

Figure 2. Overview of our proposed model.

2.1. Texture Mapping Stream

In this stream, we estimate a map based on DensePose for UV mapping, and then generated parts (details in Sec. 2.2) are mapped to the body surface of the target person by using the mapping information, i.e., IUV data. The image of the person to which the generated parts are mapped has its upper body removed before mapping [4].

This stream is important when considering the human body structure. If the parts are properly generated, the body and clothes in the image can be mapped appropriately regardless of the posture of the person.

2.2. Painting Module

The first module, the *painting module*, generates parts of the body that represent the 3D surface of the target person when trying on the target clothing. In our experiment, the clothing to be changed is limited to outerwear similar to existing methods of virtual try-on. Following the definition of [3], 10 parts included in the upper body, excluding both hands, are required. We prepare a module for each part to be generated. Each module takes two images to be changed as input. One is the image of the new clothes, and the other is the image of the person. The person's image is added to allow the input to consider human information such as skin color. In addition, the area of the clothes in the person's image is removed in advance because it is unnecessary when generating the parts.

We define the following equation (1) as the loss of each set of generator and discriminator. In the equation, G_k and D_k represent functions of the generator and discriminator respectively. The index k indicates the type of body part ($1 \leqq k \leqq 10$). The function $V_{parts}(D_k, G_k)$ of D_k and G_k are optimized using a min-max game independently for each part.

$$\min_G \max_D V_{parts}(D_k, G_k) = L_{cGAN_{parts}}(G_k, D_k) + L_1(G_k) + L_{per}(G_k). \quad (1)$$

The image $\tilde{c}_i$ is obtained by converting the image c_i into a square, and the image $\tilde{r}_i$ is obtained by removing the area of the clothes from the image of the person h_i and converting the image into a square. During the training phase, we also use y_i^k and m_i^k. y_i^k represents an image of a part obtained by converting the image of a person h_i wearing the clothes c_i along with the IUV data. In addition, m_i^k represents a mask of the existing area of the image of the part y_i^k. $L_{GAN_{parts}}(G, D)$ and $L_1(G)$ are defined through the following equations (2) (3) respectively. The output of the generator is masked in the existence area of part m_i^k during the training phase.

[4] Although both hands are removed during this process, this area does not need to be generated. Thus, it is restored along with the IUV data.

$$L_{GAN_{parts}}(D_k, G_k) = \mathbb{E}[\log D_k(y_i^k, \tilde{c_i}, \tilde{r_i})] + \mathbb{E}[\log(1 - D_k(G_k(\tilde{c_i}, \tilde{r_i}) \odot m_i^k, \tilde{c_i}, \tilde{r_i}))]. \quad (2)$$

$$L_1(G_k) = ||y_i^k - G_k(\tilde{c_i}, \tilde{r_i}) \odot m_i^k)||_1. \quad (3)$$

In addition, we add $L_{per}(G_k)$ to our loss function, which is called the perceptual loss and consists of the sum of the differences between two feature maps, to effectively reflect the pattern of the clothing in the generated result, as in [9, 4, 15].

$$L_{per}(G_k) = \sum_{i=1}^{5} \lambda_i ||\phi_i(G_k(\tilde{c_i}, \tilde{r_i}) \odot m_i^k) - \phi_i(y_i^k)||_1, \quad (4)$$

where $\phi_i(I)$ denotes the feature map of image I of the $i\text{-}th$ layer in the pretrained VGG19[14] network.

2.3. Refine Module

Although the try-on image obtained through the *texture mapping stream* takes into consideration the physical structure, it is not optimized as an entire image, and thus it lacks realistic details. The *refine module* produces a sophisticated image complete with realistic details.

Two images act as input for this module. One is an image of the new clothes to be worn, and the other is an image generated after texture mapping. We define the following equation (5) as a loss of the *refine module*.

$$\min_D V(D) = L_{GAN_{refine}}(D)$$
$$\min_G V(G) = L_{GAN_{refine}}(G) + L_1(G) + L_{per}(G). \quad (5)$$

$L_{GAN_{refine}}(G)$ and $L_{GAN_{refine}}(D)$ are shown in the following equations (6). This terms are similar to the existing method CAGAN [8]. We apply adversarial training using LSGAN [11] for higher quality and stable training.

$$L_{GAN_{refine}}(D) = \frac{1}{2}\mathbb{E}[(D(h_i, c_i) - 1)^2] + \frac{1}{2}\mathbb{E}[(D(G(t_{ij}, c_j), c_j))^2] + \frac{1}{2}\mathbb{E}[(D(h_i, c_j)))^2]$$
$$L_{GAN_{refine}}(G) = \frac{1}{2}\mathbb{E}[(D(G(t_{ij}, c_j), c_j) - 1)^2], \quad (6)$$

where c_j denotes an image of new clothes to be worn and t_{ij} denotes a rough dressing image generated through the *texture mapping stream*. The index ij indicates that the person in the image originally wears clothes c_i and changes into clothes c_j.

For $L_1(G)$ and $L_{per}(G)$, instead of changing into different clothes, the target person is set to wear the same original clothing. We formulate $L_1(G)$ and $L_{per}(G)$ as the following equations.

$$L_1(G) = ||h_i - G(t_{ii}, c_i)||, \quad (7)$$

$$L_{per}(G) = \sum_{i=1}^{5} \lambda_i ||\phi_i(G(t_{ii}, c_i)) - \phi_i(h_i)||_1, \quad (8)$$

where t_{ii} denotes a rough image in which the wearer has changed into the same original clothes.

3. Experiments

3.1. Dataset

A dataset of image pairs of a person and clothing to be worn by the person was acquired from Zalando. In addition, the size of the images are set to 256×192. We removed noisy images. The total number of prepared images is 9,286 pairs. In total, 9,000 pairs are used for training and 286 pairs are used for testing. In addition, we use the segmentation data obtained by [2] and the IUV data obtained by [3].

3.2. Implementation Details

3.2.1 Training Setup

We use the optimization method from Adam[10] with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The learning rate is 0.0001. For training, the *painting module* requires 30K steps, and the *refine module* needs 90K steps with a batch size of 16.

3.2.2 Architecture

We use a similar architecture as Unet [12] for both modules. In addition, we use transposed convolutions (deconvolutions) for upsampling and add batch normalization[6] to the networks after each convolutional layer. For fairness, we change the CAGAN network to the same architecture as *refine module* so that the output size is the same as our proposed method (256×196). Note that the input and output size of *painting module* is 128×128.

3.3. Quantitative Evaluation

We conducted a user study for a quantitative evaluation. Similar to [4], the study was conducted on the Amazon Mechanical Turk (AMT) platform. For each trial, the participants were presented with a person's image, an image of the clothes to be worn, and two images of the result of the try-on by two different models. The participants were then

asked which appears more real and shows a correct change of clothes. Note that IS [13] and FID [5] are often used for quantitative evaluations for the synthesis quality of images generated by various models. However, they do not reflect whether the target clothes have been naturally transferred.

To demonstrate the effectiveness for various postures, we provided the evaluations by separating the postures. For this separation, we define an indicator, that is, the *average cosine (AC)*, to evaluate human postures by using keypoints obtained by [1]. We calculate the indicator based on the bending angle of both arms in a 2D image, in which the angles are determined as θ_1 and θ_2 and AC is calculated as $AC = \frac{1}{2}(cos\theta_1 + cos\theta_2)$. When AC is small (defined as SMALL), the image of the person is almost upright because the arm is not bent. When the AC is large (defined as LARGE), the arm is bent and the image is different from an upright pose. Our proposed method is expected to provide good results even for an image defined by LARGE. In addition, because a clothes pattern is also an important factor in changing clothes, we considered the complexity of each pattern. To quantify the complexity of a pattern, we use TV norm in the same manner as in [15]. If the clothes pattern is complicated, it is expressed as LARGE. By contrast, if the clothes pattern is monotonous, it is expressed as SMALL.

In our experiment, we applied each posture (LARGE or SMALL) and the detailed richness of the clothes (LARGE or SMALL). There are four combinations in total. We compared our proposed methods with CAGAN and CP-VTON. Table 1 shows the result. When the posture of the person is LARGE, that is, when the posture is different from upright, our proposed method achieves better results than the other methods, regardless of the detailed richness of the clothes.

posture	LARGE		SMALL	
clothes	LARGE	SMALL	LARGE	SMALL
CAGAN	38.28%	46.88%	42.97%	50.78 %
CP-VTON	52.34%	45.31%	**60.94 %**	**58.59 %**
UVTON	**59.38%**	**57.81%**	46.09%	40.68%

Table 1. Results of pairwise comparisons of try-on methods. A higher value is better. The rate of chance is 50%.

3.4. Qualitative Evaluation

We also compare the images generated by each method as a qualitative evaluation. The results generated by each method are shown in Figure 1. When the AC value and TV norm are small (images in the upper left), all methods have relatively good results. When AC is small and TV norm is large (the upper-right images), CP-VTON and UVTON are successful. Further, when the AC is large (images at the bottom), images generated by CAGAN and CP-VTON have the patterns of clothes on the person's arms. However, UVTON achieves better results even under this type of case.

4. Conclusion

We proposed UVTON, a GAN-based virtual try-on method that utilizes UV mapping to consider the 3D surface of a body structure. Based on our experiments, we showed that our proposed method can naturally change the clothes of a person in various postures, which has been difficult to achieve thus far.

References

[1] Z. Cao, T. Simon, S. Wei, and Y. Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *CVPR*, 2017.

[2] K. Gong, X. Liang, D. Zhang, X. Shen, and L. Lin. Look into Person: Self-supervised Structure-sensitive Learning and A New Benchmark for Human Parsing. *CVPR*, 2017.

[3] R. A. Guler, N. Neverova, and I. Kokkinos. DensePose: Dense Human Pose Estimation In The Wild. *CVPR*, 2018.

[4] X. Han, Z. Wu, Z. Wu, R. Yu, and L. S. Davis. VITON: An Image-based Virtual Try-on Network. *CVPR*, 2018.

[5] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *NeurIPS*, 2017.

[6] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ICML*, 2015.

[7] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. *CVPR*, 2017.

[8] N. Jetchev and U. Bergmann. The Conditional Analogy GAN: Swapping Fashion Articles on People Images. *ICCVW*, 2017.

[9] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. *ECCV*, 2016.

[10] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *ICLR*, 2015.

[11] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley. Least Squares Generative Adversarial Networks. *ICCV*, 2017.

[12] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *MICCAI*, 2015.

[13] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved Techniques for Training GANs. *NeurIPS*, 2016.

[14] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *ICLR*, 2015.

[15] B. Wang, H. Zheng, X. Liang, Y. Chen, L. Lin, and M. Yang. Toward Characteristic-Preserving Image-based Virtual Try-On Network. *ECCV*, 2018.

[16] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. *ICCV*, 2017.