# Dress Code: High-Resolution Multi-Category Virtual Try-On

## Supplementary Material

Davide Morelli[1], Matteo Fincato[1], Marcella Cornia[1], Federico Landi[1,*], Fabio Cesari[2], and Rita Cucchiara[1]

[1] University of Modena and Reggio Emilia, Italy
{name.surname}@unimore.it
[2] YOOX NET-A-PORTER, Italy
{name.surname}@ynap.com

## 1 Dress Code Dataset

In Table 1, we report the total number of images and the dimensions of the train/test splits in the Dress Code dataset. Additionally, we detail the dimension of each split with respect to the different macro-categories of the dataset (upper-body clothes, lower-body clothes, and dresses). In our experiments, we train our models on all training pairs of the dataset and test both on each category separately and on the entire test set. Additional sample image pairs from Dress Code are shown in Fig. 1, 2, and 3 with the corresponding pose keypoints, dense pose of the reference model, and segmentation mask of the human body. As it can be seen, images of our dataset have a great variety considering both the body pose of the reference models and category and textures of try-on garments. This can lead to virtual try-on architectures becoming more general and adapting to more challenging scenarios.

## 2 Additional Implementation Details

**Data Pre-Processing.** To extract the person pose representation $p$, we employ either OpenPose [1] or DensePose [2]. Specifically, the keypoints of the human body extracted with OpenPose [1] are used to compute the 18-channel pose heatmap, where each channel corresponds to one body keypoint represented as an $11 \times 11$ white rectangle. While both the 25 channels label map and the 2 channels UV map estimated by DensePose [2] are concatenated and used with no further processing.

In order to create the masked person representation $m$, we remove the information regarding the target clothes and the interested part of the body from $I$. Hence, the model only sees the face, the hair, and the target person part of the body which do not contain ground-truth information. To produce such masked

*Now at Huawei Technologies, Amsterdam Research Center, the Netherlands.

**Table 1.** Number of train and test pairs for each category of the Dress Code dataset.

| | Images | Training Pairs | Test Pairs |
|---|---|---|---|
| **Upper-body Clothes** | 30,726 | 13,563 | 1,800 |
| **Lower-body Clothes** | 17,902 | 7,151 | 1,800 |
| **Dresses** | 58,956 | 27,678 | 1,800 |
| **All** | 107,584 | 48,392 | 5,400 |

representation, we use both the target label map to extract the clothes area and the pose map to extract the area of the limbs. These areas are then merged to form the mask which is then dilated to avoid the model getting information about the target shape. Finally, all the non-modifiable areas in the image (*e.g.* face, hands, hairs, etc.) are subtracted from the generated mask. The final mask is then applied to the image $I$. Note that while dilating the mask introduces more complexity in the paired setting generation task, it is essential in the unpaired one, especially when trying to substitute a garment with another whose shape covers a much larger area of the image.

**Warping Module.** Two feature extraction networks are included in the warping module, with four 2-strided down-sampling convolutional layers with a kernel size of 4 plus two 1-strided ones with a kernel size of 3. The first extraction network takes as input the try-on clothing item $c$, while the second one works on the concatenation between the person representation $m$ and the pose of the reference person $p$. Following [5], a correlation map is then computed between the outputs of the two feature extraction networks and then fed to a convolutional network, consisting of two 2-strided convolutional layers with a kernel size of 4 and two 1-strided convolutional layers with a kernel size of 3. The output is forwarded through a fully connected layer that predicts the parameters of the geometric transformation. In particular, these parameters are the TPS anchor point coordinate offsets having a size of $2 \times 5 \times 5 = 50$. Batch normalization is applied to all convolutional layers. For the high-resolution versions of our model, we add an additional 2-strided down-sampling convolutional layer with a kernel size of 4 to both feature extraction networks.

**Human Parsing Estimation Module.** It is based on the U-Net architecture with four blocks in both encoder and decoder. Each block is composed of two sequences of a convolutional layer with a kernel size of 3, instance normalization, and a ReLU activation function. Each encoding block is followed by a 2-strided max pooling layer with a kernel size of 2, while each decoding block is preceded by a 2-strided transposed convolutional layer with a kernel size of 2 to upsample feature maps. Each encoding block is connected to the corresponding decoding block using skip connections. When training with high-resolution images, we add a U-Net block in both encoder and decoder.

**Try-On Module.** The encoder has four U-Net blocks, each having two convolutional layers with a kernel size of 3 and a 2-strided max pooling layer with a kernel size of 2. The decoder is symmetric but, instead of max pooling, the feature maps are up-sampled using a 2-strided transposed convolutional layer with

**Table 2.** High-resolution results on the Dress Code test set

| | Upper-body | | | Lower-body | | | Dresses | | | All | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Model** ($512 \times 384$) | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **IS** ↑ |
| CP-VTON [5] | 0.850 | 48.24 | 3.365 | 0.826 | 57.38 | 4.00 | 0.845 | 24.04 | 0.891 | 0.831 | 29.24 | 1.671 | 3.096 |
| CP-VTON$^{\dagger}$ [5] | 0.916 | 14.37 | 0.442 | 0.910 | **12.54** | **0.432** | 0.861 | 21.82 | 0.720 | 0.896 | 10.08 | 0.425 | 3.277 |
| **Ours (Patch)** | **0.943** | 13.48 | 0.346 | **0.936** | 19.86 | 0.717 | **0.893** | 20.35 | 0.623 | **0.923** | 9.44 | **0.246** | 3.310 |
| **Ours (PSAD)** | 0.936 | **11.65** | **0.180** | 0.931 | 17.83 | 0.643 | 0.884 | **15.99** | **0.324** | 0.916 | **7.27** | 0.394 | **3.320** |
| | **Upper-body** | | | **Lower-body** | | | **Dresses** | | | **All** | | | |
| **Model** ($1024 \times 768$) | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **SSIM** ↑ | **FID** ↓ | **KID** ↓ | **IS** ↑ |
| CP-VTON [5] | 0.862 | 60.40 | 4.730 | 0.840 | 60.35 | 4.236 | 0.858 | 24.44 | 0.873 | 0.853 | 36.68 | 2.379 | 3.155 |
| CP-VTON$^{\dagger}$ [5] | 0.931 | 14.63 | 0.387 | 0.930 | **16.46** | **0.393** | 0.877 | 23.80 | 0.832 | 0.912 | 9.96 | 0.338 | 3.300 |
| **Ours (Patch)** | **0.944** | 13.38 | 0.273 | 0.933 | 19.97 | 0.654 | **0.890** | 24.14 | 0.807 | **0.922** | 9.99 | 0.370 | 3.34 |
| **Ours (PSAD)** | 0.941 | **12.10** | **0.171** | **0.935** | 19.02 | 0.641 | 0.882 | **17.93** | **0.425** | 0.919 | **7.70** | **0.236** | **3.357** |

a kernel size of 2. Also in this case, when training with high-resolution images, we add a U-Net block in both encoder and decoder.

**Discriminator.** PSAD works at pixel-level, classifying each pixel as one of the $N$ semantic classes of the human parser or as fake. The architecture is composed of 6 downsampling and 6 upsampling blocks arranged according to the U-Net architecture. The last layer is a $1 \times 1$ spatial convolution that brings the feature dimensionality to $N + 1$.

When training the Patch-based baseline, we instead employ PatchGAN [3] as our discriminator, which does not operate at pixel-level but instead classifies square image patches as real or fake, averaging all predictions to get the final result. It consists of three 2-strided down-sampling convolutional layers and one 1-strided down-sampling convolutional layer, all having a kernel size of 4. We use a convolutional layer to generate a scalar output in the last layer. Except for the first, we utilize batch normalization and apply Leaky ReLU with a 0.2 slope after each layer.

**Training.** The experiments with low-resolution images are performed using a batch size of 32, while we use a batch size of 16 when training with high-resolution images for both $512 \times 384$ and $1024 \times 768$ resolutions. All experiments with $256 \times 192$ and $512 \times 384$ images are performed on 4 NVIDIA V100 GPUs, taking 10 hours to train the human parsing estimation module, one day for the warping module training stage, and around two days to train the try-on module. When instead training with full-resolution images, we split the batch size on 16 GPUs.

## 3 Additional Results

**Low-Resolution Results.** In Fig. 4, we show some failure cases, while some additional qualitative comparisons between PSAD and the corresponding Patch-based baseline are shown in Fig. 5. In Fig. 6, 7, and 8, we report further try-on results on sample image pairs respectively extracted from upper-body clothes, lower-body clothes, and dresses by comparing our model with previously proposed try-on architectures re-trained on our newly collected dataset.

**High-Resolution Results.** Table 2 shows the complete try-on performances when generating high-resolution images while, in Fig. 9, we report some qualitative results.

**Fig. 1.** Sample images of upper-body clothes and reference models from Dress Code.

**Fig. 2.** Sample images of lower-body clothes and reference models from Dress Code.

**Fig. 3.** Sample images of dresses and reference models from Dress Code.


**Fig. 4.** Failure cases on the Dress Code test set.


**Fig. 5.** Qualitative comparison between PSAD and the Patch-based baseline.


CP-VTON†
[5]
WUTON
[4]
ACGPN
[6]
Ours
(PSAD)

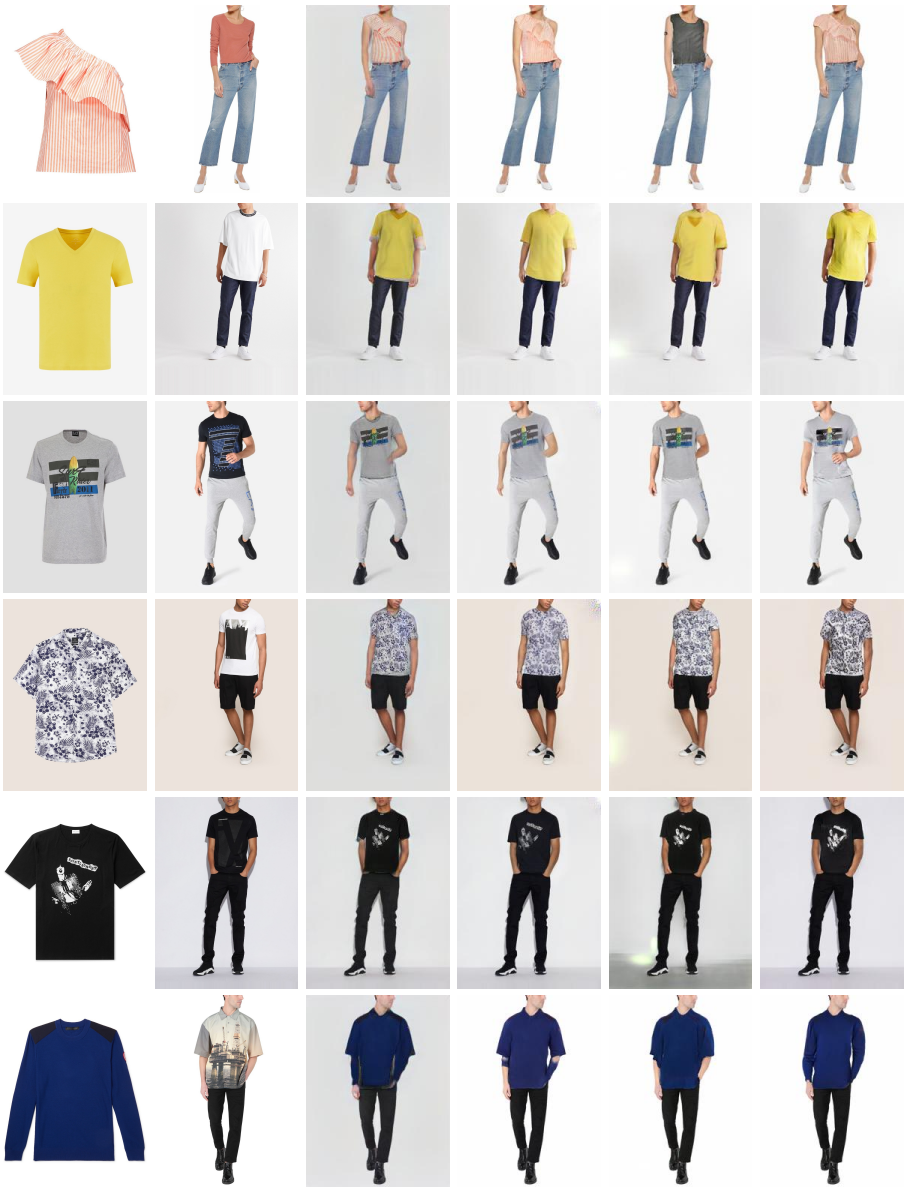

**Fig. 6.** Sample try-on results using upper-body clothes and reference models from the Dress Code test set.

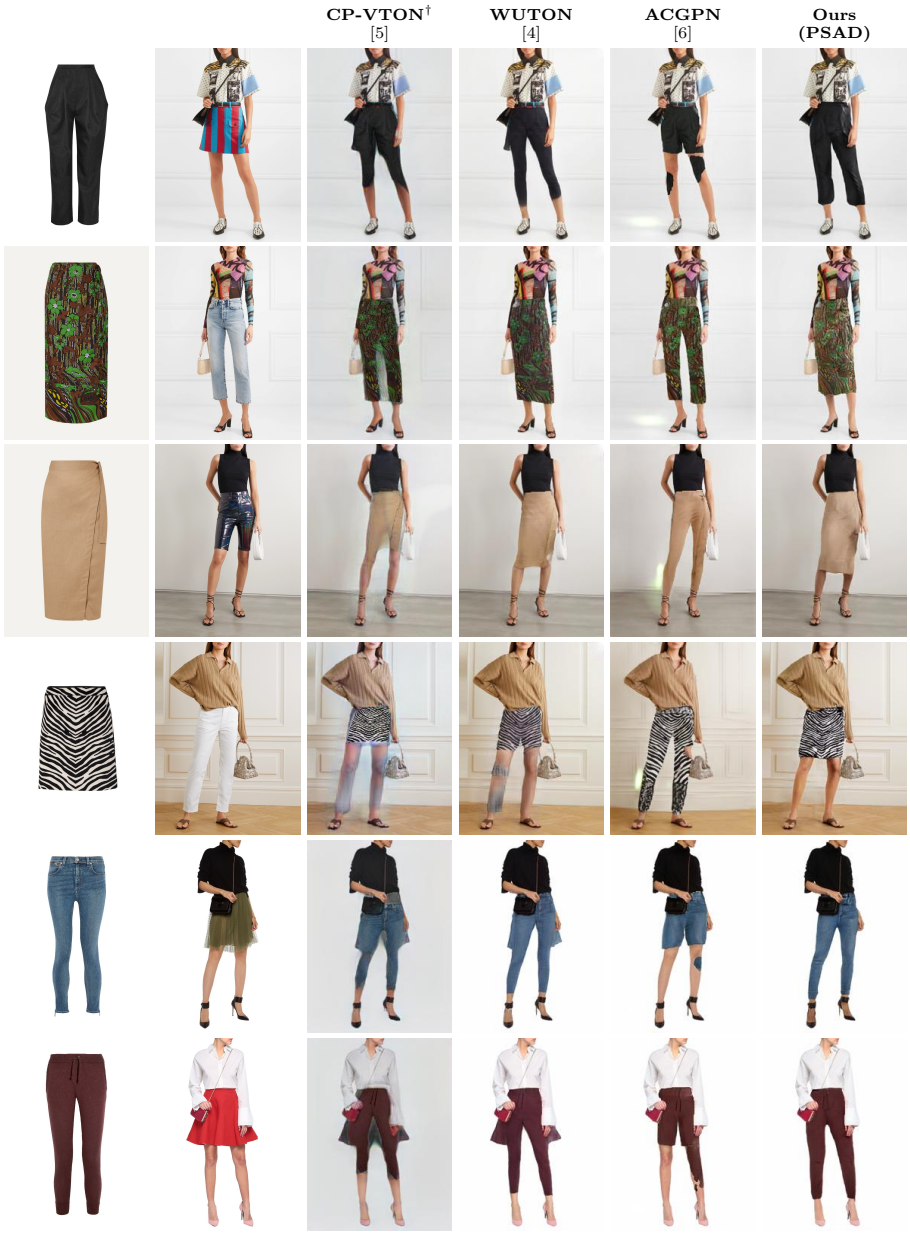
CP-VTON† [5]
WUTON [4]
ACGPN [6]
Ours (PSAD)

**Fig. 7.** Sample try-on results using lower-body clothes and reference models from the Dress Code test set.


CP-VTON† [5]
WUTON [4]
ACGPN [6]
Ours (PSAD)

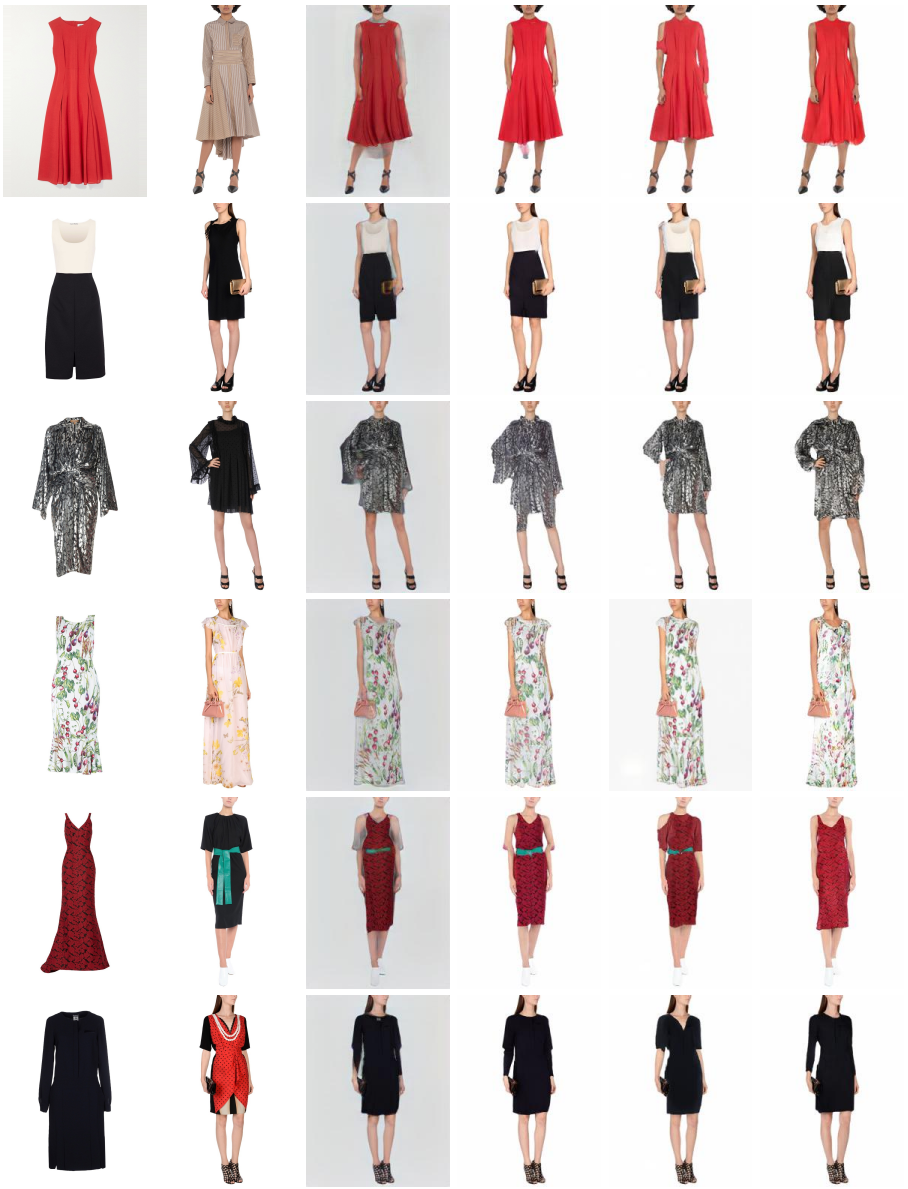

**Fig. 8.** Sample try-on results using dresses and reference models from the Dress Code test set.

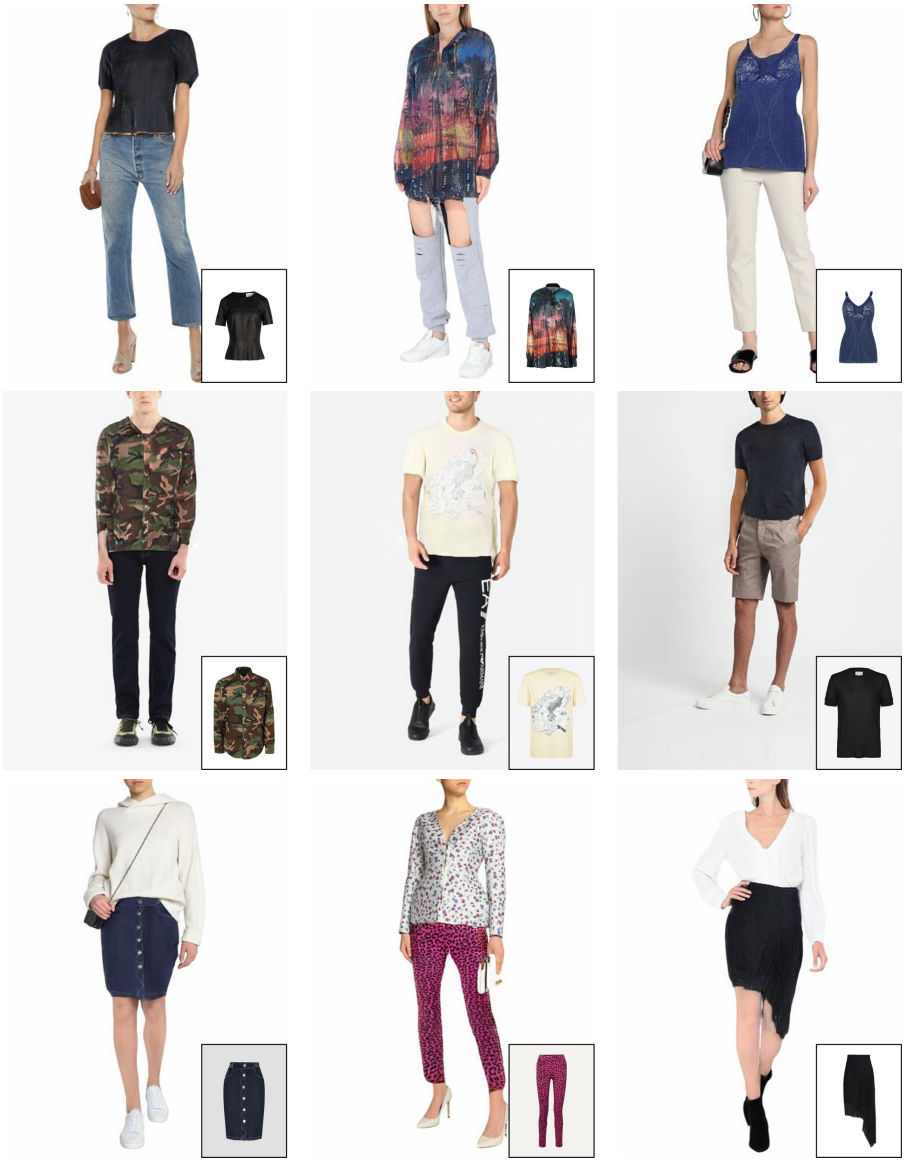

**Fig. 9.** Sample high-resolution results on the Dress Code test set.

## References

1. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. In: CVPR (2017) 1
2. Güler, R.A., Neverova, N., Kokkinos, I.: DensePose: Dense Human Pose Estimation In The Wild. In: CVPR (2018) 1
3. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-To-Image Translation With Conditional Adversarial Networks. In: CVPR (2017) 3
4. Issenhuth, T., Mary, J., Calauzènes, C.: Do Not Mask What You Do Not Need to Mask: a Parser-Free Virtual Try-On. In: ECCV (2020) 8, 10, 12
5. Wang, B., Zheng, H., Liang, X., Chen, Y., Lin, L., Yang, M.: Toward characteristic-preserving image-based virtual try-on network. In: ECCV (2018) 2, 3, 8, 10, 12
6. Yang, H., Zhang, R., Guo, X., Liu, W., Zuo, W., Luo, P.: Towards Photo-Realistic Virtual Try-On by Adaptively Generating-Preserving Image Content. In: CVPR (2020) 8, 10, 12