VIRTUAL OUTFIT CHANGE

Matthew Leinhauser | Mauricio Ferrato
University of Delaware
{mattl, mferrato}@udel.edu

Abstract:
The ability to virtually try-on outfits can be a big boost to online shopping. For this project, we tried to improve upon the work of Wang et al. in their paper, Toward Characteristic-Preserving Image-based Virtual Try-on, by modifying their pipeline. In this project, we simulate a virtual outfit try-on of in-shop clothes onto reference persons in the dataset Virtual Try-on Network (VITON). First, we create a Geometric Matching Module (GMM). The GMM learns a thin-plate spline transformation for transforming the in-shop clothes into fitting the body shape of the target person. Next, we create a Try-On module. The Try-On module alleviates boundary artifacts of warped clothes and learns a composition mask to integrate the warped clothes and the rendered image to ensure smoothness thus ensuring more realistic results. Wang et al. originally trained the GMM on 200,000 steps. In our work, we train both modules on 20,000 steps. Compared to the original work, our work is not as good as Wang et al.

1. Introduction

1.1 Literature Review

Online clothes shopping has continued to rise in popularity. In many countries, due to the COVID-19 pandemic, shopping online for clothing is the only option to purchase new clothes. Fortunately, there has been much work has been done in the area of virtually trying on clothes. For this project, we initially looked at three papers. First, we read “Dress me up: me up! content-based clothing image retrieval” [1]. In this paper, Mustaffa et al. proposed a framework for content-based clothing image retrieval (CBIR). The framework allows for searching and retrieving of similar clothes based on the article of clothing’s color and shape features. They proposed a CBIR framework because most text-based image retrieval is not always accurate depending on the annotations. Additionally, it is expensive to do for very large datasets. The CBIR framework works by first converting the RGB image to the HSV color space. They do this because the HSV color space is closer to what humans see. Next, a color histogram is computed for color representation of the image. Then, high and low thresholding is computed for shape representation. Finally, a similar piece of clothing is selected as a match based on the

Manhattan distance (less distance means more similar). Along with the matching item's image being shown, the brand, price, and size of the clothing item are also shown.

The second paper we read was "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations" [2]. In this paper, Liu and his co-authors created the largest dataset of clothing, DeepFashion, containing 800,000 images with 50 annotation categories per image, four to eight landmarks per clothing item, 1,000 attributes, and 300,000 cross-pose/cross-domain photo pairs. They also created a model, FashionNet which learns clothing features by jointly predicting clothing attributes and landmarks. FashionNet is optimized iteratively and landmarks are used to pool the learned features. The structure of FashionNet is identical to VGG16 except the last layer is replaced by three branches of layers. The first branch captures global features for the article of clothing, the second branch captures local features pooling over the estimated landmarks, and the third branch predicts the landmarks locations and visibility (whether or not the landmarks are occluded in the image). Additionally, the outputs of the first and second branches are concatenated to jointly predict categories, attributes, and to model clothes pairs. FashionNet outperformed existing models by 20% on category classification and 7% on attribute prediction.

The final paper we read has not been officially published yet. In less than one month at CVPR '20, Yang et al. [3] will present their state-of-the-art work on this area of virtual try on. In this paper, they propose a novel visual try-on network, Adaptive Content Generating and Preserving Network (ACGPN) to transfer a target clothing image onto a reference person while morphing the "character" of the clothing image to create a photo-realistic try-on of the article of clothing independent of the reference person's pose and/or occlusions of the clothing item. ACGPN has three modules. The first module is a "semantic layout generation module that utilizes semantic segmentation of the reference image to progressively predict the desired semantic layout after try-on". The second module wraps clothes images to a reference person in an image based on the generated semantic layout. Additionally, in this module, a second-order difference constraint stabilizes the warping process during the training phase. Finally, the third module fuses together the reference image, semantic layout, and warped clothes to produce a virtual-try on image that adapts to the pose of the reference person in the image.

1.2 First Attempt

We attempted to port the work in the paper, "Towards Photo-Realistic Virtual Try-On by Adaptively Generating ↔ Preserving Image Content" and improve their results, if at all possible. Yang and his colleagues programmed ACGPN using PyTorch. We were interested in seeing if using the Keras framework in TensorFlow would make a difference or not. We successfully ported the conditional Generative Adversarial Network (cGAN), the U-Net, and Spatial Transformer Network (STN) from PyTorch to Keras. Additionally, we created functions to read and load the different classes from the dataset. The dataset we used is called Virtual Try-on Network (VITON) [4]. VITON contains 19,000 images containing a woman facing the camera (front-view) and images of clothing tops. Around 16,250 pairs of images are cleaned. The dataset also contains JSON files that contain annotations about the pose of the individual in an

image. Unfortunately, we struggled to get our model to train. We faced issues with the STN, cGANs, and training the pipeline.

First, the implementation of the STN was extremely vague in the paper and we struggled to grasp how to build it. We faced the same issue with the cGANs. When it came to training the network, again, the paper was vague. Looking at the GitHub repository for the paper showed that the authors did not upload the code for training. Due to these challenges, we decided to restart our project.

1.3 Second Attempt

For our second attempt, we read the paper, “Toward characteristic-preserving image-based virtual try-on network” by Wang et al. [5]. In this paper, the authors address the task of image-based virtual try-on as a conditional image generation problem. They do so by introducing a fully-learnable Characteristic-Preserving Virtual Try-on Network (CP-VTON). CP-VTON learns a thin-plate spline (TPS) transformation for transforming the in-shop clothes into fitting the body shape of the target person via a Geometric Matching Module (GMM). The GMM used is an end-to-end neural network directly trained using pixel-wise L1 loss. A second module, the Try-on module, is introduced to alleviate boundary artifacts of warped clothes and make the results more realistic. To do so, the Try-on module learns a composition mask to integrate the warped clothes and the rendered image to ensure smoothness.

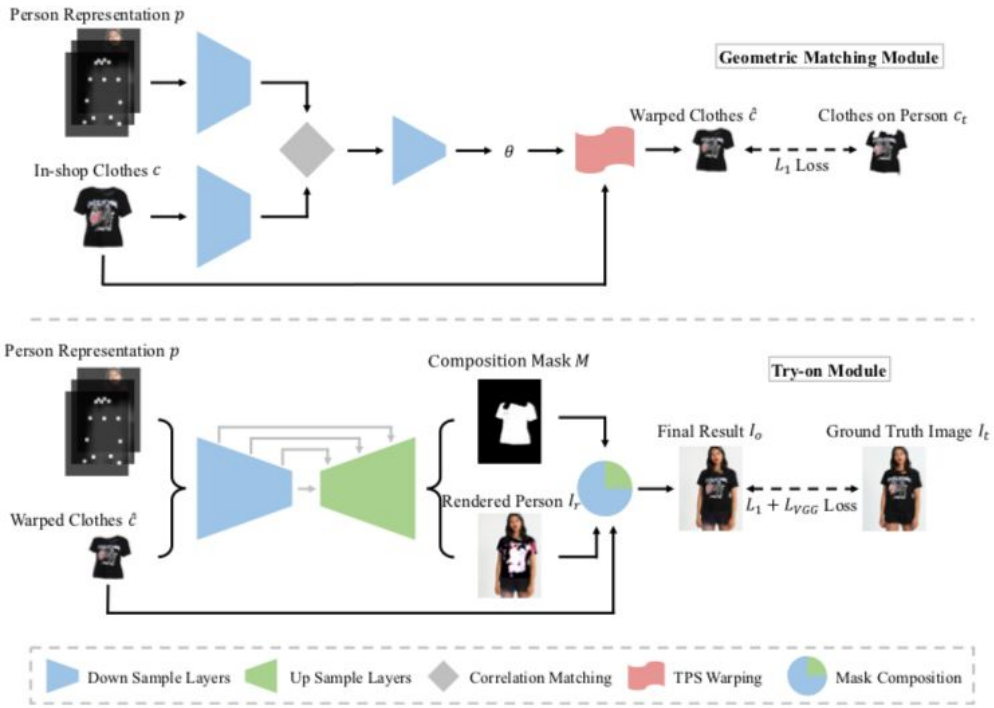

Fig. 1: The network infrastructure from the paper Toward characteristic-preserving image-based virtual try-on network

2. Methodology

We wrote code based on the paper by Wang et al. We originally thought about using Keras, the TensorFlow framework, for carrying out this work, but we opted for PyTorch instead because VGG19 exists as a pretrained model in PyTorch and not Keras. We also wanted to experiment with training the GMM and Try-On modules differently than the authors described.

After training on the dataset, we ran inference on the training set again to get the warped clothing item and warped mask images. Wang et al. never ran inference on the training set again and instead saved the warped clothing item and warped mask images to a TensorBoard [7]. They then used the TensorBoards as an input to train the Try-On module. Instead we used those images to train the Try-On module (first with the warped clothing images and again with the warped mask images), and ran inference on the test dataset for both the warped cloth and warped mask images. Just like with the GMM, Wang's team saved the resultant images from the Try-On module as TensorBoards. We saved our results as JPG images. We also trained our model on 20,000 steps compared to Wang's 200,000. We trained on a tenth of the steps because we wanted to see if running inference multiple times on the training and testing sets would improve the results showed in the paper.

3. Analysis and Results

All our results are available online in a compressed format on our GitHub repository for this project [6]. We also include a subset of our results as well. Below we provide some sample results for the GMM and Try-On Module.

Fig. 2: In left to right order: the original shirt image, the warped shirt image, and the warped mask of the shirt from the training set of the GMM.

Fig. 3: the original clothing image, the warped clothing image, and the warped mask of the clothes from the testing set of the GMM.

Fig. 4: Our result (Left) compared to a real-life image from the dataset (Right)

Fig. 5: Our result (Left) vs. the original work (Middle) vs. a real-life photo of the model

Unfortunately, our results do not show improvement in the output images as we had hoped. Our outputs display slight color distortion, body part distortion, and more blur than the ground-truth images and the output images Wang et al. produced. However, the background color in our output images did not distort as much as Wang and his colleagues did.

4. Conclusions and Future Work

The ability to virtually try-on outfits can be a big boost to online shopping. For this project, we tried to improve upon the work of Wang et al. in their paper, Toward Characteristic-Preserving Image-based Virtual Try-on Network. In this project, we simulate a virtual outfit try-on of in-shop clothes onto reference persons in the dataset Virtual Try-on Network (VITON). First, we created a GMM. The GMM learned a thin-plate spline transformation for transforming the in-shop clothes into fitting the body shape of the target person. Next, we created a Try-On module. The Try-On module alleviated boundary artifacts of warped clothes and learned a composition mask to integrate the warped clothes and the rendered image to ensure smoothness thus ensuring more realistic results. Wang et al. originally trained the GMM on 200,000 steps. In our work, we trained both modules on 20,000 steps. We also differed from the authors' work by doing the following: Comparing our results to Wang et al., and the ground-truth images, shows that our results are not refined. Our results show slight color distortion, body part distortion, and more blur. We can thus conclude that our refined workflow process does not produce the same quality results as Wang et al. Since this was only a project for a course, we currently do not have any plans in place to continue this work in the future. However, if we were to continue this project in the future, we would see if increasing the number of training steps from 20,000 to 200,000 would make a difference in our outputs. Additionally, we would also incorporate functions to sharpen our output images so they like more similar to the ground-truth images.

5. Bibliography

[1] Mustaffa, M. R., Wai, G. S., Abdullah, L. N., & Nasharuddin, N. A. (2019, January). Dress me up! content-based clothing image retrieval. In *Proceedings of the 3rd International Conference on Cryptography, Security and Privacy* (pp. 206-210).

[2] Liu, Z., Luo, P., Qiu, S., Wang, X., & Tang, X. (2016). Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1096-1104).

[3] Yang, H., Zhang, R., Guo, X., Liu, W., Zuo, W., & Luo, P. (2020). Towards Photo-Realistic Virtual Try-On by Adaptively Generating $\leftrightarrow $ Preserving Image Content. *arXiv preprint arXiv:2003.05863*.

[4] xthan/VITON: Code and dataset for paper "VITON: An Image-based Virtual Try-on Network"

[5] 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).

[6] https://github.com/mferrato/FinalProjectCISC642

[7] tensorflow/tensorboard: TensorFlow's Visualization Toolkit