Virtual Outfit Change

Final Project by Mauricio Ferrato and Matt Leinhauser

Literature Review

- Three papers
  - Towards Photo-Realistic Virtual Try-On by Adaptively Generating ↔ Preserving Image Content [1]
  - Deep Fashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations [2]
  - Dress me up!: content-based clothing image retrieval [3]

[1] 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*.

[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] 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).

Dress me up!: content-based clothing image retrieval

- Propose a framework for content-based clothing image retrieval (CBIR) that allows searching and retrieving of similar clothes based on the article of clothing's color and shape features. Essentially help to find a matching outfit while shopping
  - Convert RGB image to HSV color space
  - Compute the color histogram for color representation
  - Compute high and low thresholding for shape representation
  - Pick a similar piece of clothing that matches with the person's outfit by Manhattan distance
  - Display piece of clothing, brand, size, and price
- Proposed this because text-based image retrieval isn't always accurate depending on annotations. Plus, it's expensive to do for large datasets

Deep Fashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations

- Created DeepFashion - clothes dataset containing 800K images with rich annotations, attributes, clothing landmarks, and different scenes (store view, street view, consumer view)
- Propose a model. FashionNet which learns clothing features by jointly predicting clothing attributes and landmarks
  - Landmarks are used to pool the learned features
  - Optimized iteratively
- Network structure is identical to VGG16 except last layer which is replaced by three branches of layers
  - First branch captures global features for the article of clothing
  - Second branch captures local features pooling over the estimated landmarks
  - Third branch predicts the landmarks locations and visibility (occluded or not in image)
  - Outputs of the first and second branches are concatenated to jointly predict categories, attributes, and to model clothes pairs

Towards Photo-Realistic Virtual Try-On by Adaptively Generating ↔ Preserving Image Content

- 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:
  - Semantic layout generation module utilizes semantic segmentation of the reference image to progressively predict the desired semantic layout after try-on.
  - A clothes warping module warps clothing images according to the generated semantic layout, where a second-order difference constraint is introduced to stabilize the warping process during training.
  - An inpainting module for content fusion integrates all information (e.g. reference image, semantic layout, warped clothes) to adaptively produce each semantic part of human body.

Dataset

- Dataset Used: Virtual Try-On Network (VITON)
  - 19,000 images containing a woman facing the camera (front-view) and images of clothing tops
    - ~16,250 cleaned image pairs

Easy



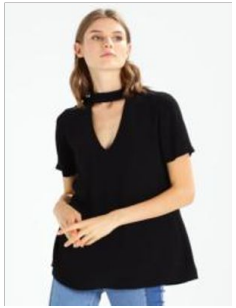
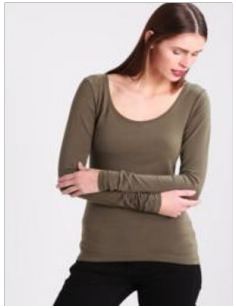


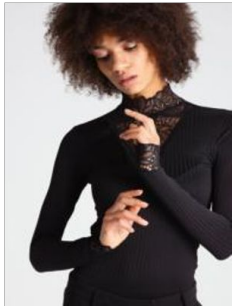

Medium



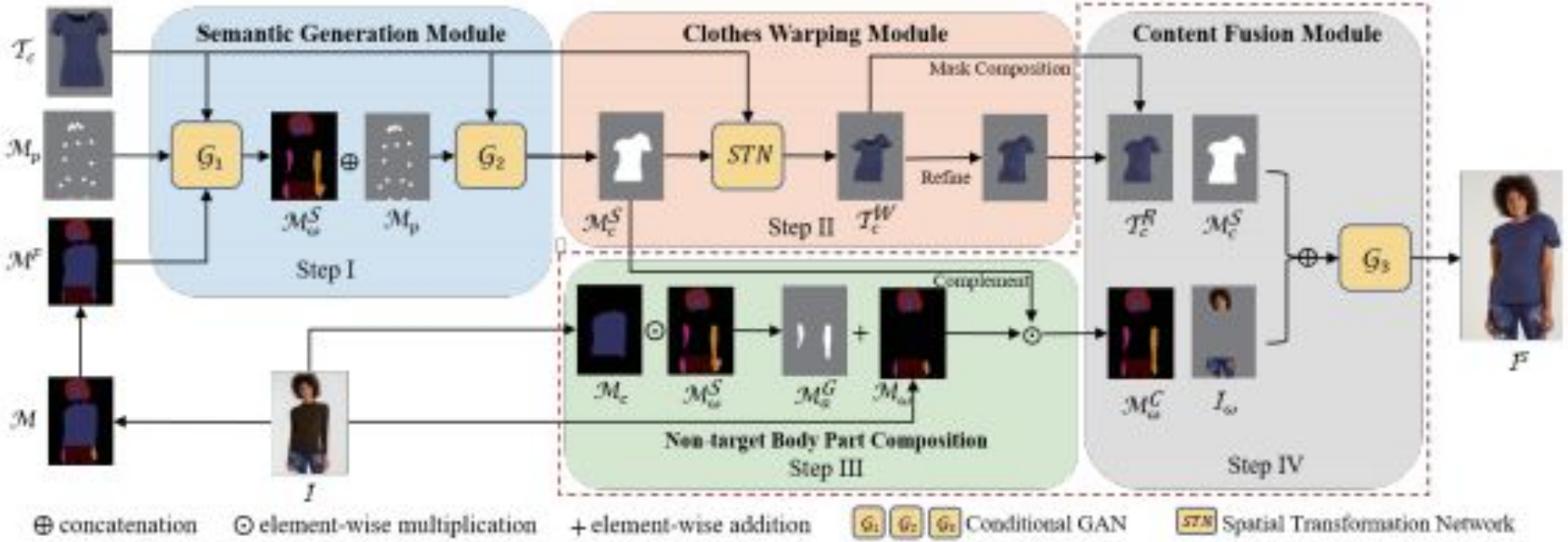



Hard

Network Architecture



Our Work

- GOAL: Recreate the work from the paper Towards Photo-Realistic Virtual Try-On by Adaptively Generating ↔ Preserving Image Content using Keras
- Ported the structures U-Net, STN, and GAN to keras
- Successfully created functions to read from the different dataset classes (reference person (JPG, PNG, PPM, BMP, TIFF), target clothes item (JPG, PNG, PPM, BMP, TIFF), pose data (JSON))

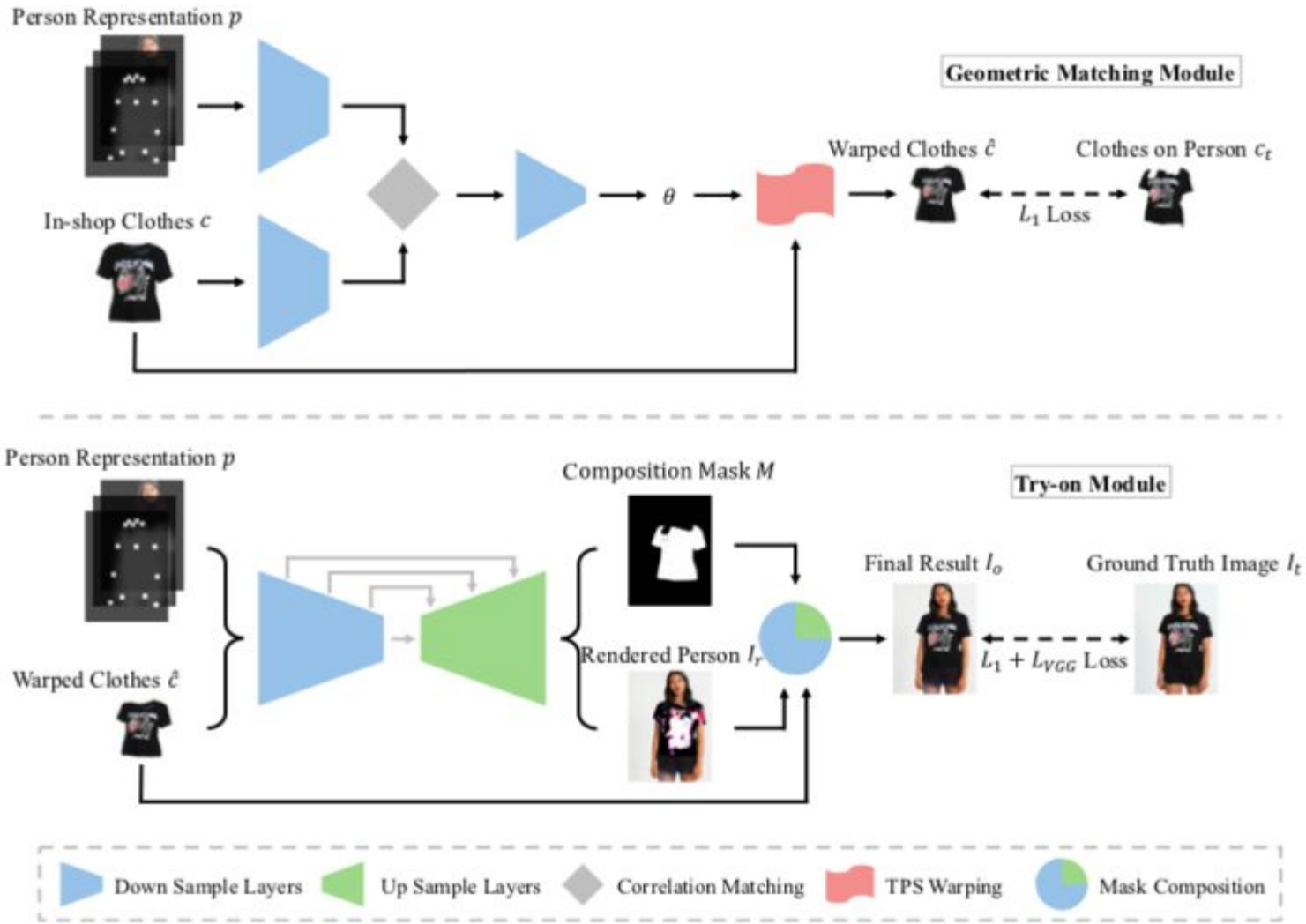
Challenges

- We weren’t completely sure how to implement the STN or the cGANs the way they implemented it in the paper
- Authors’ pipeline was pretty intense
  - Virtually no information on how to train
- Dataset was different than we thought

Starting Over

New Paper: Toward Characteristic-Preserving Image-based Virtual Try-On Network[1]

- Address the task of image-based virtual try-on as a conditional image generation problem
- Propose a new fully-learnable Characteristic-Preserving Virtual Try-On Network (CP-VTON)
  - Learns a thin-plate spline transformation for transforming the in-shop clothes into fitting the body shape of the target person via a new Geometric Matching Module (GMM)
    - GMM is a end-to-end NN directly trained using pixel-wise L1 loss
  - To alleviate boundary artifacts of warped clothes and make the results more realistic, we employ a Try-On Module that learns a composition mask to integrate the warped clothes and the rendered image to ensure smoothness.


Person Representation p
In-shop Clothes c
Geometric Matching Module
θ
Warped Clothes $\hat{c}$
Clothes on Person c_t
L_1 Loss
Person Representation p
Warped Clothes $\hat{c}$
Composition Mask M
Rendered Person I_r
Try-on Module
Final Result I_o
Ground Truth Image I_t
$L_1 + L_{VGG}$ Loss
Down Sample Layers
Up Sample Layers
Correlation Matching
TPS Warping
Mask Composition

Geometric Matching Module

Takes as input the image representation (image of person, pose map, segmentation map) and image of cloth piece we want to put on the person

4 steps:

- Two networks that gives feature maps from person and from the clothing
- A correlation layer that takes the feature maps and combines into one Tensor
- A regression network that predicts the spatial transformation parameters
  - With the tensor from the previous step as input
- A Thin-Plate Spline transformation module to warp the clothing image

Trained using pixel-wise L1 loss between the warped image and ground truth

Training Setup

Training set: ~14,000 images

Test set: ~2,000 images

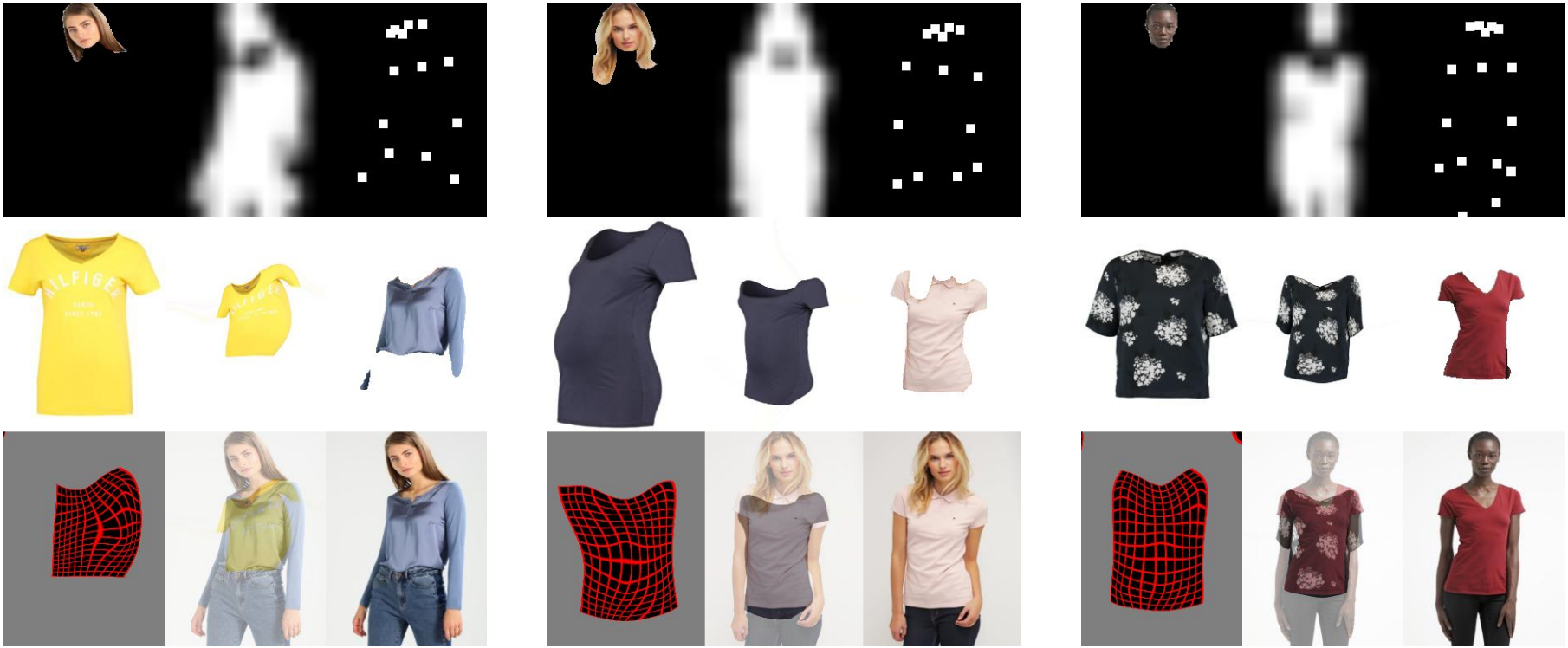
20,000 steps, lr = 0.0001 (Decays after 10,000 steps), batch size = 4, Adam optimizer

Images resized to: 256 x 192

Trained using a NVIDIA V100 from one of our in-house lab machines

```
step:       40, time: 0.389, loss: 0.113744
step:       60, time: 0.388, loss: 0.129006
step:       80, time: 0.399, loss: 0.184615
step:      100, time: 0.393, loss: 0.121065
step:      120, time: 0.392, loss: 0.074770
step:      140, time: 0.375, loss: 0.117801
step:      160, time: 0.394, loss: 0.087797
step:      180, time: 0.408, loss: 0.181944
step:      200, time: 0.383, loss: 0.210272
step:      220, time: 0.381, loss: 0.153568
step:      240, time: 0.391, loss: 0.080294
step:      260, time: 0.371, loss: 0.137610
step:      280, time: 0.385, loss: 0.085587
step:      300, time: 0.406, loss: 0.099048
```

Outputs



Try-On Module

Goal: fuse together the warped clothes with the reference person in the image and align with that person’s body shape

- i.e. minimize the discrepancy between the ground truth and output

Working on getting the try-on module to work

Takes person representation and warped cloth image

Two steps:

- Train a U-Net to get a composition mask for the clothes and person
- Matrix-multiplication to combine the images