The Fitting Room In Your Dream

Team 30%: Ming Zhang, Binqi Shen, Hanyu Cai, Junpeng Jiang, Xuan Yu, Chelsea Zhang | MSiA 432 Deep Learning | Spring 2021 | Northwestern University

Problem Statement

- With the aim of preventing viral transmission, virtual try-ons have started to gain more popularity. **Our project aims to provide a virtual fitting room function for customers who want to change clothes online.**
- This problem is hard to solve because it is hard to accurately pinpoint clothing pixels on an image. Predecessors who worked on this in the past usually built models which allows shoppers to try on existing clothes and designs. A **key advantage we try to build** with our model is that users will be able to explore how famous artwork could be incorporated into their own clothing designs and gain purchasing insights.

Dataset

- **Training:** iMaterialist Fashion 2019 at FGVC6
  Fine-grained segmentation data for fashion and apparel. Labels of clothes category is provided for 50000+ pictures.
- **Testing:** Female Dresses
  2000+ full face images of women in dress
- **Cleaning:**
  - resize all images to contain the same width and length of pixels.
  - Filter out images that does not meet the pixel requirement

Challenges:

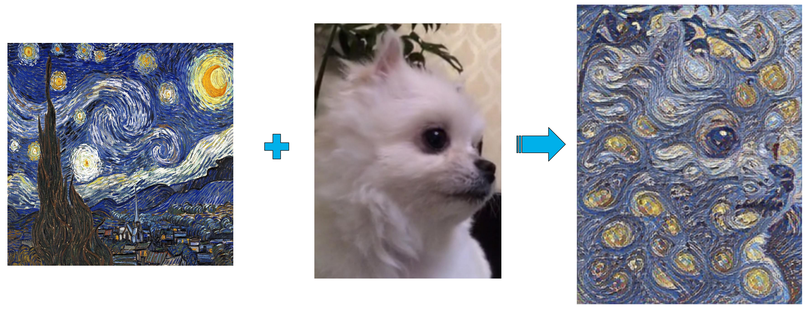
- **Computation power** is needed for to pinpoint exact pixels on the image
- **Labels** of clothes category on pixel level (needed for training) unavailable for images from Pinterest
- Unable to detect **special poses** or overly **complicated dress design**
- Need to up sampling pixels by increasing the resolutions

- **Process:** combination of Google Colab and local CPU
- If we use our dataset to train the model, we don't have to do any like shifting or cropping, or augmenting our dataset. However, we might need to up sampling pixels for each image in order to increase the resolution of the image. Bi-linear interpolation, cubic interpolation or nearest neighbor interpolation might be useful for up sampling.

Technical Approach

Step 1

Generate Stylized Designs:
Pretrained arbitrary image stylization model

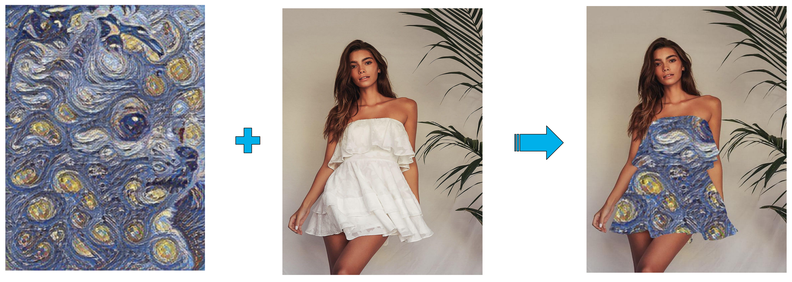


Step 2

Pinpoint the Clothing Shape
Apply UNET for clothes segmentation

Step 3

Make Your Dream Come True:
Iteratively replace pixels on dress image



- **Step 1:** Use Neural Style Transfer (NST) to generate a new design based on two pictures: a famous artwork and a content image uploaded by the user (could be picture of a pet or favorite designer brand logo). NST helps us in separating the style representations and content representations in a CNN.
- **Step 2:** Use Python albumentations package with pretrained parameters to locate the area of clothes in the picture. UNET enables us to convert feature maps into vectors and reconstruct images from the vectors.
- **Step 3:** Our masked picture have positive values only on clothes part. Thus, we recursively replace the pixel on original image with pixel on new masked image if the value is positive to get the output image.

Results

- The newly generated designs are **satisfactory** most of the time.
- Based on the output pictures shown, our pipeline is able to **accurately identify shapes of clothing** within a picture and replace the original color with newly generated designs.
- 6 of us in our team and 10 volunteers tried our model. **15 out of 16 people really liked the resulting synthesized image and appreciated the newly generated design.**

Conclusion

- **Summary:** The generated designs are considered fashion and our model is able to pinpoint the clothes on the image uploaded and help users try on clothes they designed.
- **Limitation 1:** The current model and pipeline only focus on one clothes type: dress.
- **Limitation 2:** The model occasionally identifies shoes and other accessories as part of the clothing
- **Limitation 3**: The pattern design can combine features in the artwork and the image uploaded, but cannot be controlled over the way of combining
- **Improvement Potentials**: Add more clothes types to attract more users; Adjust the transparency of newly generated clothes before plugging it back to the picture to maintain features like texture

References and Related Work

- Iglovikov, Vladimir. Binary Segmentation of Various Cloths, GitHub, 2020. https://github.com/ternaus/cloths_segmentation
- Rai, Manas. "A Model with an 'Eye' for Fashion." *Medium*, Medium, 14 Dec. 2019,medium.com/@manasrai/a-model-with-an-eye-for-fashion-d1aedbadee8c.
- TensorFlow: Large-scale machine learning on heterogeneous systems,2015. Software available from tensorflow.org.