For the "Try On" functionality, CP-VTON+ library was used, a deep learning-based method specializing in image-based virtual try-on. Its primary objective is to seamlessly transpose the shape and texture of clothing onto a target person's body while maintaining a realistic appearance. It is implemented using PyTorch.

The current implementation of the same in flask version is specifically trained on the dataset of tops.

The CP-VTON+ pipeline comprises two key modules:

GMM (Generative Matching Module): It focuses on matching the appearance of the clothing to the person's pose, ensuring a realistic and well-fitted virtual try-on.

and TOM (Try-On Module): Blending the warped cloth onto the target person, taking into account various visual aspects to enhance the realism of the final result

In the Flask version, the network heavily relies on the GMM and TOM models. The training involves the integration of OpenPose, which aids in obtaining pose points for the clothing.

The dataset structure consists of the folders:

1. image: This folder contains images of individuals wearing the clothing. Each image must have a corresponding clothing image in the folder mentioned in the 4th point.

2. image-parse: These images depict the segmentation of the human body, with each part shaded in a distinct color. The latest Facebook segment anything model, LIP\_JPPNet, or CIHP\_PGN pretrained networks can be employed to generate these images.

3. cloth: This folder comprises only the clothing images.

4. cloth-mask: Binary masks of the clothing images, generated using simple Pillow/OpenCV functions. The cloth is represented in white, and the background is in black.

5. pose: Pose keypoints of the individual, generated using OpenPose COCO 18 model.

To perform the virtual try-on, users need to input both their image and the clothing image.

The system has demonstrated higher accuracy when the input images align with the dataset it was trained on.

The network can be trained on the dataset from the specific ecommerce shop itself for the best accuracy.

Following this it can be trained to work on other types of clothing items

For developing 3D body models and obtaining accurate body measurements, libraries such as SMPL (and its subsequent versions) can be utilized. These libraries enable the extraction of detailed body shape information, providing mesh vertices and facilitating the creation of an OBJ file for further measurements.

Alternative like PifuHD offer speed and simplicity in generating 3D models, it has a limitation by including clothing in the output, potentially leading to inaccurate measurements.

Two notable repositories, <https://github.com/DavidBoja/SMPL-Anthropometry> and <https://github.com/vcarlosrb/3d-body-measurements>, have been found working on SMPL. If the functionality of SMPL for custom 2D image input to generate a 3D model output is figured out, these repositories can be used.

While numerous libraries like Shapy exist for 3D body modeling, there is currently no reference found for the measurements of the body parts using these libraries.