VIRTUAL CLOTHING TRY-ON

Try Before Buy!

Group Members

BSEF19M003 - Laiba Kamal

BSEF19M030 - Mahreen Asama

BSEF19M034 - Ahmad Sarwar


01/11/2018 07:44:05PM
CAM 6
clideo.com


MA KU SARY KAPRY NA NIKLWAO
MA TO LADIES HU

Let us make life easier for these picky people!

Problem Statement

Whether it is an online shopping app or website, a shopping mall, or a small clothing shop, customers often face difficulties deciding on the product they want to purchase. It also causes problems for the seller, because it decreases sales, increases returns and in case of online stores, it causes shipping waste. Customers also often return unsatisfied.

What is Virtual Clothing Try-On?

- Consumers try on clothing virtually using their phones or tablets.
- The technology uses augmented reality which layers visuals over the real world via the device's camera.

Why is our problem worth solving?

- Online shopping is trendy these days but virtual try-on is not. (in Pakistan)
- Online shopping stores/ apps.
- Shopping malls.
- Makes decision making easy.
- Increased sale and purchase.
- Decreased return rates.

Why is our problem worth solving?

A study says that:

- **61% of online shoppers prefer to shop in stores that offer AR try-ons and visualizations than in stores that do not**
- **And 71% of them would shop more often if AR were offered.**

Proposed Solution

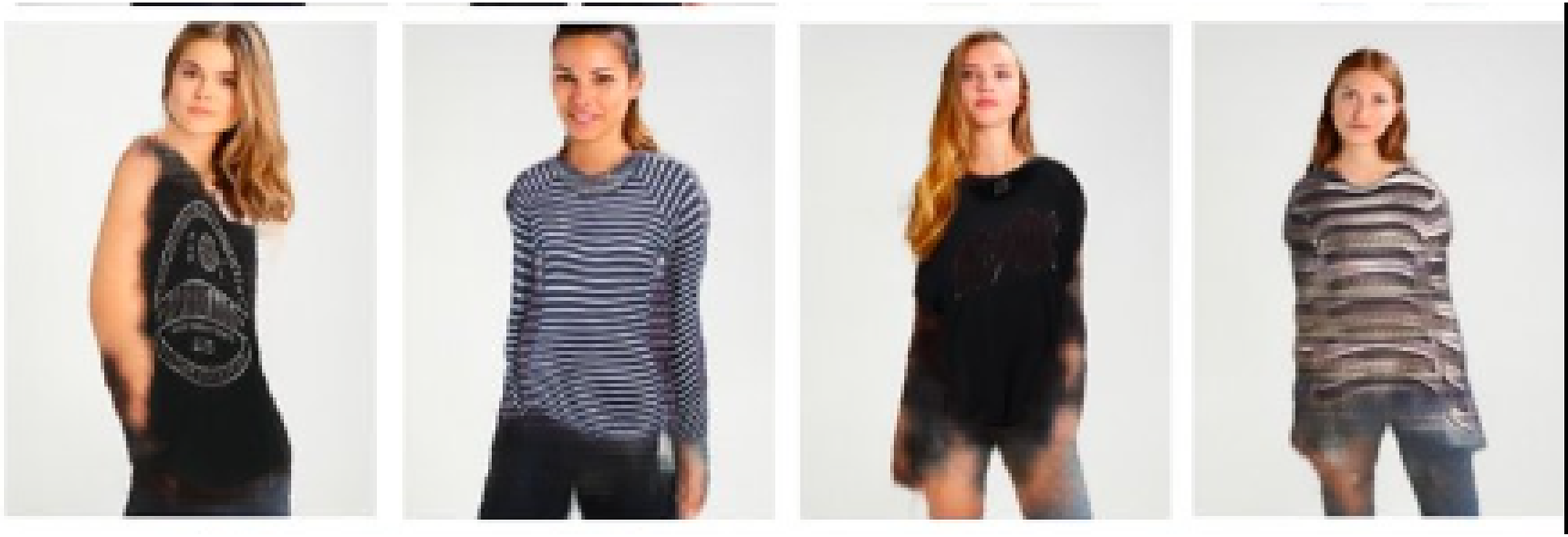
Many approaches such as **CP_VTON**, etc. have been proposed as the solution to this problem in the past, but those were not efficient and have the following drawbacks:

- **focused on preserving the character of the clothing image**
- **bad results over different human poses**
- **work for clear and straightforward human images only**

In-shop Clothes

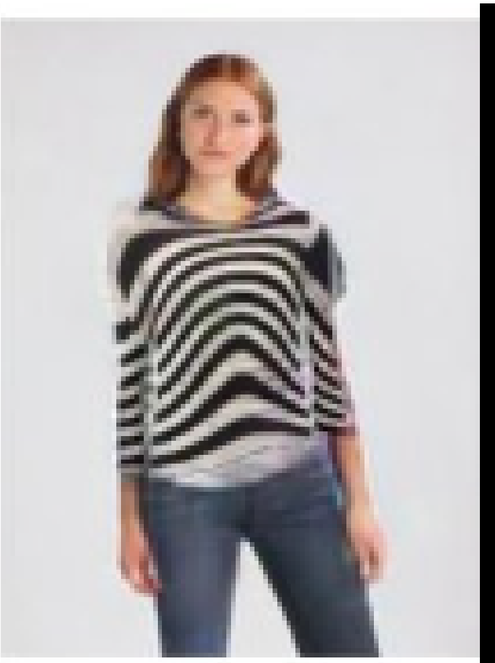
Target Person

VITON



CP-VTON



ACGPN RESULTS

How does ACGPN work?

- Adaptive Content Generating and Preserving Network
- Firstly, it predicts semantic layout of reference image using semantic segmentation
- Next, a cloth warping module warps clothing image according to generated semantic layout
- Lastly, an inpainting module for content fusion integrates all information

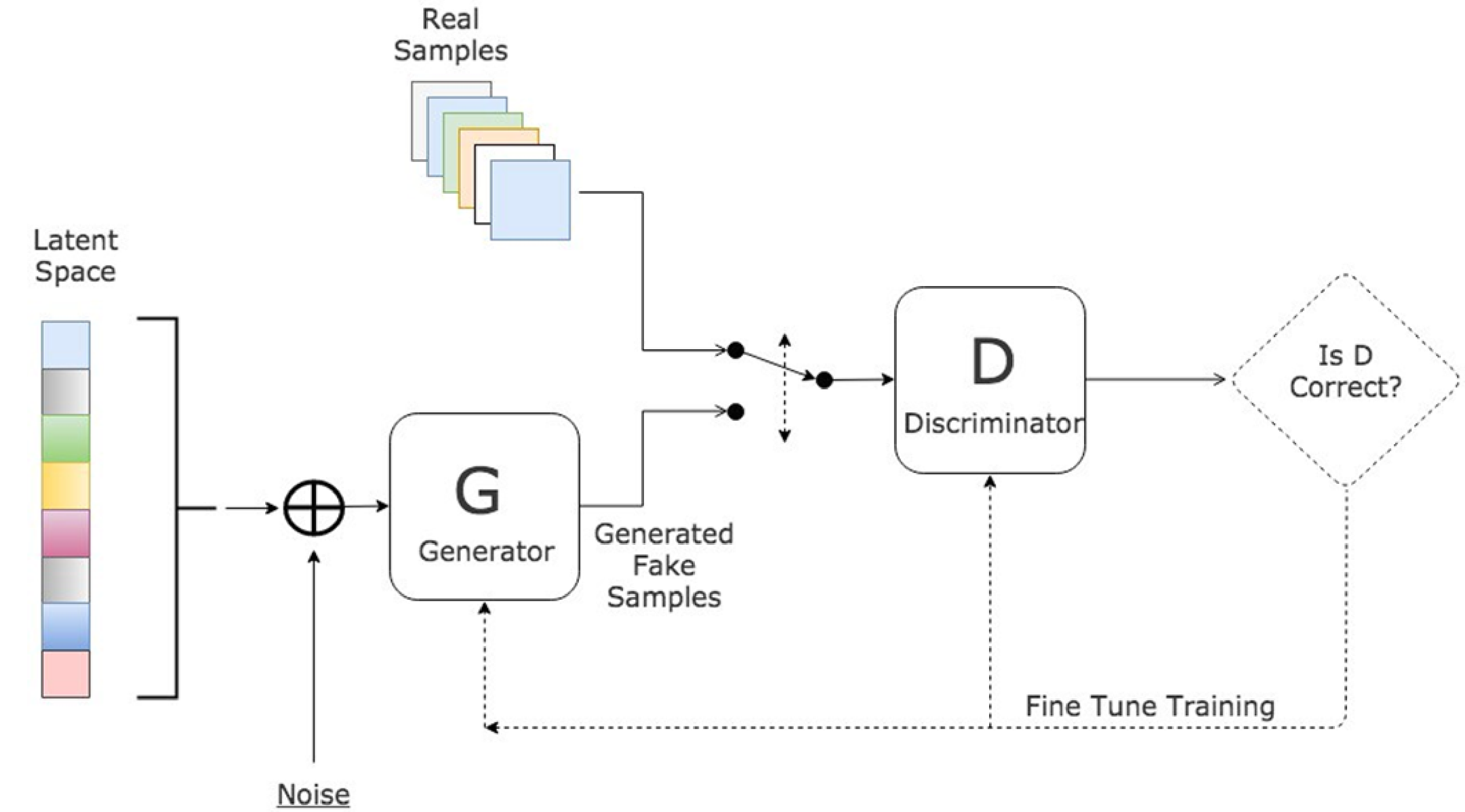
ACGPN

ACGPN or **Adaptive Content Generating and Preserving Network**, is a **generative adversarial network** for virtual try-on clothing applications.

Generative Adversarial Networks

- Generative Adversarial Networks or **GANs** are a type of generative model that are used to **generate realistic images**.
- It simultaneously train two models, a **Generator G** and a **Discriminator D**.
- The generator generates fake images from real ones, while discriminator distinguishes between them.

Generative Adversarial Network



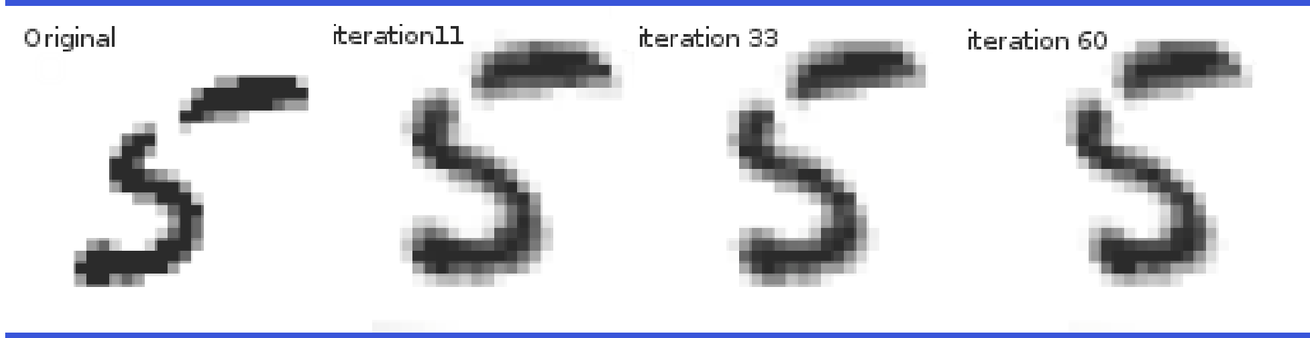
Conditional GANs

CGAN

CGAN or **Conditional Generative Adversarial Networks** are deep learning neural networks in which some additional parameters are used. Labels are also put in inputs of discriminator in order to classify the input correctly.

STNs

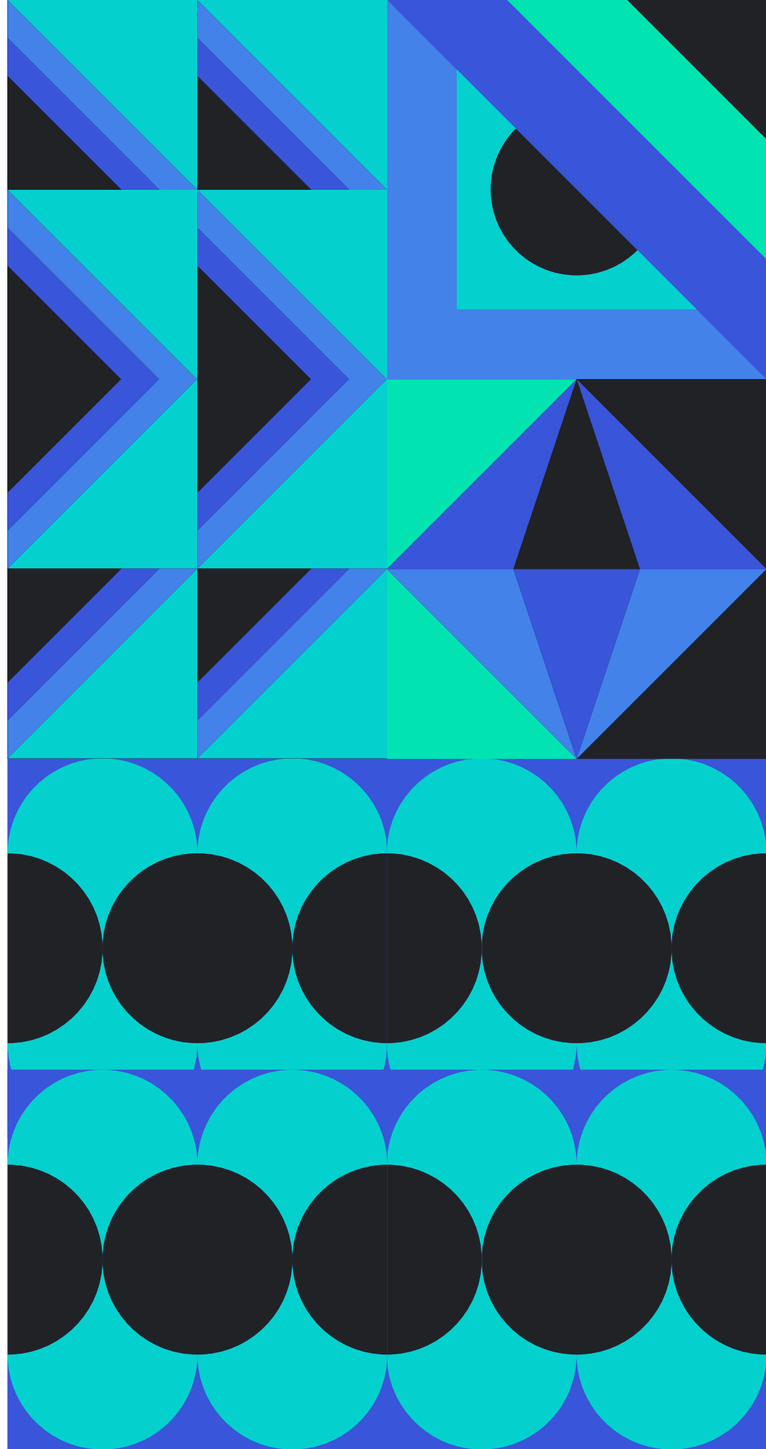
STNs or **Spatial Transformation Networks** allow a neural network to learn how to perform spatial transformations on the input image in order to enhance the geometric invariance of the model.



ALGORITHM

ACGPN working is divided into **3** modules:

- Semantic Generation Module **(SGM)**
- Clothes Warping Module **(CWM)**
- Content Fusion Module **(CFM)**



ACGPN MODULES

STEP 1

In **Step 1**, the Semantic Generation Module **(SGM)** generates the mask of the body parts and the mask of the warped clothing regions via semantic segmentation.

STEP 2

In **Step 2**, the Clothes Warping Module **(CWM)** generates warp target clothing image according to warped clothing mask through Spatial Transformation Network (STN).

ACGPN MODULES

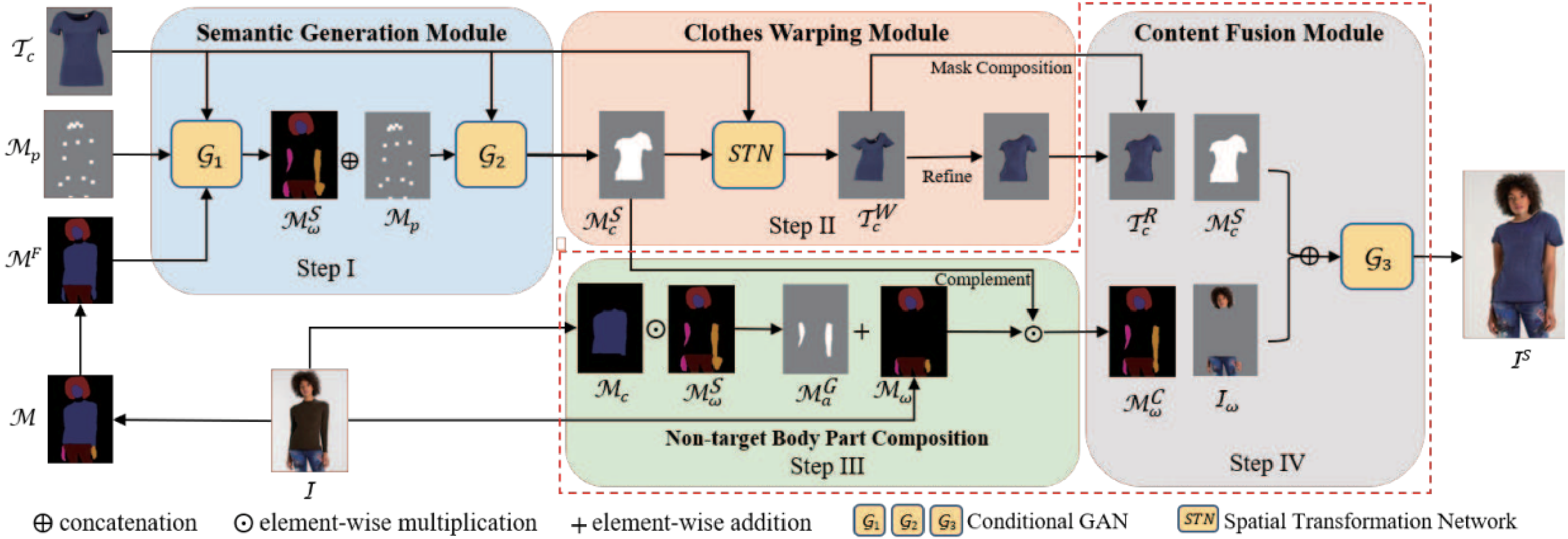
STEP 3

Content Fusion Module **(CFM)** is composed of Step 3 and 4. In particular, **Step 3** is designed to fully maintain the untargeted body parts as well adaptively preserve the changeable body parts (i.e, arms)

STEP 4

Step 4 fills in the changeable body parts by utilizing masks and images generated from previous steps through an inpainting based fusion GAN.

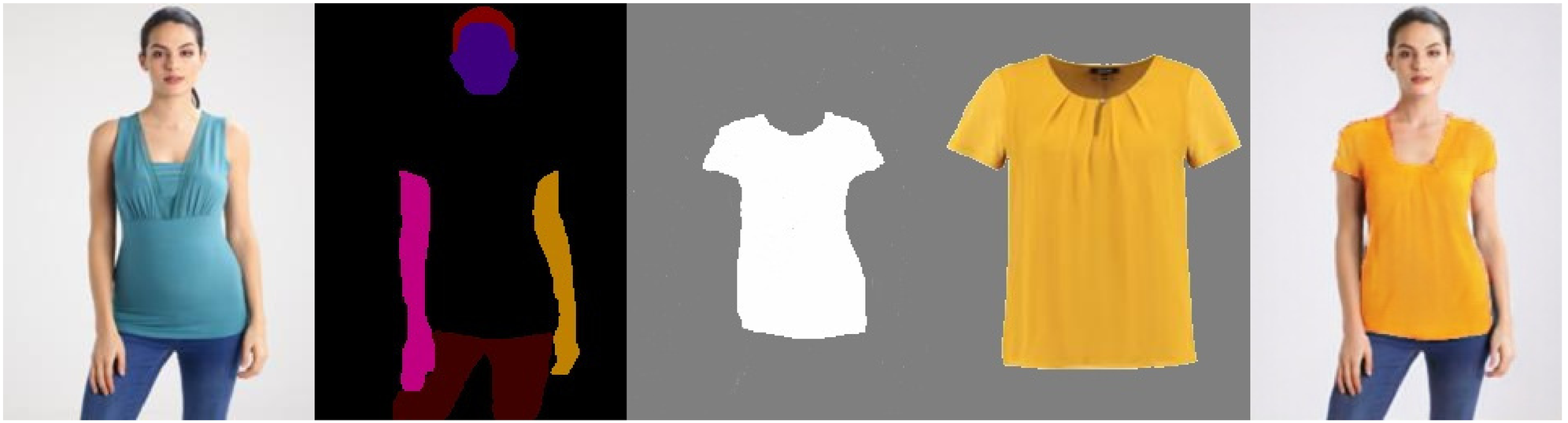
OVERALL ARCHITECTURE OF ACGPN



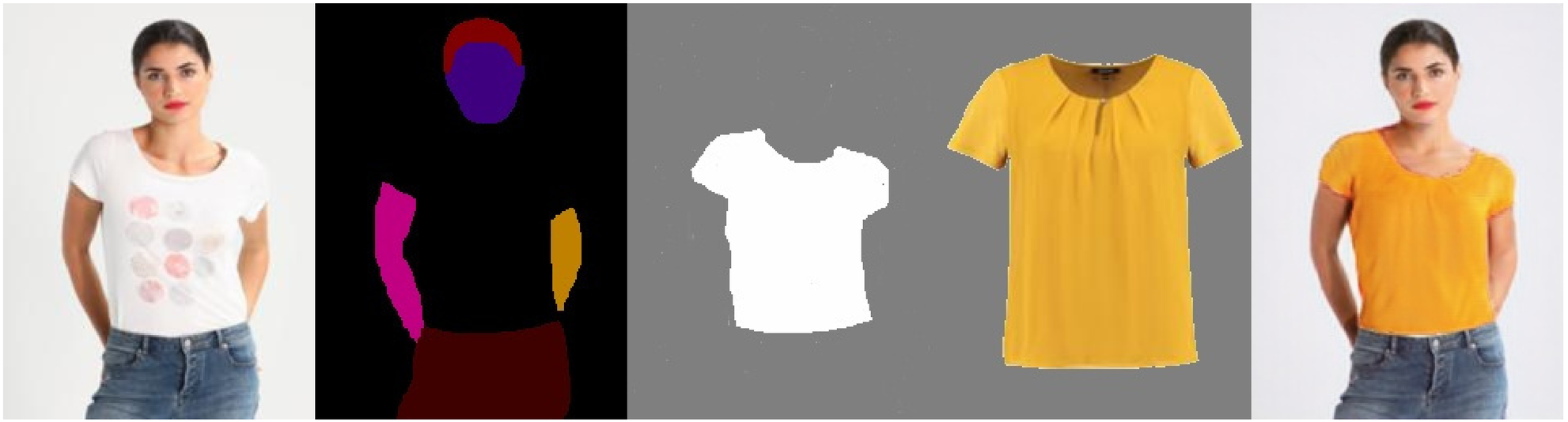
Implementation Details

- Dataset **VITON** is used.
- Resolution for all images in training and testing is **256 x 192**.
- Each module in proposed method is trained for **20 epochs**.
- The network is optimized by **Adam** optimizer.
- Code is implemented by deep learning toolkit **PyTorch**.

RESULTS (EASY)



RESULTS (MEDIUM)



RESULTS (HARD)

RESULTS (HARD)

RESULTS (CUSTOM)

RESULTS (CUSTOM)

Draw Backs

- ACGPN is **gender** biased :)
- It gives good results with **Good Camera**
- It take almost **30sec** in generating Keypoints on body, hence can not be used in live camera
- If there's some **Background** it'll not work properly

Future Direction

- We can integrate it in live camera by improving the speed of key point generation.
- We can build a plugin of this project that can be ingrates in any e-commerce websites.
- We can embed it in a System that can be placed at cloth shop.

FUTURE DIRECTION

TEAM CONTRIBUTION

LAIBA KAMAL 30%

- VITON drawbacks, comparison with ACGPN

MAHREEN ASAMA 30%

- ACGPN and its algorithm (working)

AHMAD SARWAR 40%

- Code execution, camera integration
- Background Removing

REFERENCES

- "Towards Photo Realistic Virtual Try-On by Adaptively Genarating <-> Preserving Image Content" [cv. CV - 2020]

- https://github.com/switchablenorms/DeepFashion_Try_On

- https://www.kaggle.com/code/ma00753045/tryon/

THANK YOU!


ANY QUESTION?