A PROJECT REPORT ON

# FASHION TRY-ON USING POLY-GAN

delivered by


Shruti Nair (C22018111067)
Meera Narale (C22018111068)
Rajvee Pisey (C22018771865)


**in partial fulfillment for the award of the degree of**

**Bachelor of Technology in**

**ELECTRONICS AND TELECOMMUNICATION of SAVITRIBAI PHULE PUNE UNIVERSITY**

**under the guidance of**
Dr. Mrudul Dixit

**Sponsored by:** In-House



The Department of Electronics and Telecommunication of
CUMMINS COLLEGE OF ENGINEERING FOR WOMEN, KARVENAGAR, PUNE – 411052
(An Autonomous Institute affiliated to
SAVITRIBAI PHULE PUNE UNIVERSITY)


**Academic Year**
2021 – 2022

a) **Project title:** Fashion Try-On using GAN

b) **Subject Area:** Artificial Intelligence/ Deep Learning

c) **Nature of Project:** Software

# CERTIFICATE

**This is to certify that**

Shruti Nair
Meera Narale
Rajvee Pisey

**Have successfully completed the work on their PROJECT TOPIC**

Fashion Try-On using GAN

---

**in partial fulfillment for the award of the degree of**

**Bachelor of Technology in**

**ELECTRONICS AND TELECOMMUNICATION of SAVITRIBAI PHULE PUNE UNIVERSITY,**

**in**

**CUMMINS COLLEGE OF ENGINEERING FOR WOMEN, KARVENAGAR, PUNE - 411052**
(An Autonomous Institute affiliated to
SAVITRIBAI PHULE PUNE UNIVERSITY)

| Internal Guide | Head of Department | Principal |
|---|---|---|
| Dr. Mrudul Dixit | Dr. Prachi Mukherji | Dr. M.B. Khambete |

# ACKNOWLEDGEMENT

We consider ourselves privileged to express our gratitude and respect towards all those who are guiding us through the course of this project work. The satisfaction that accompanies the smooth progress of this project would be incomplete without the mention of the people, whose constant guidance and encouragement make it comfortable. We convey thanks to our project guide, Dr. Mrudul Dixit, for providing the encouragement, guidance and for maintaining good communication during these challenging times of Covid which helps working remotely enjoyable.

We are thankful to Dr. Prachi Mukherjee, Head of the Department Electronics & Telecommunications Engineering, for her support and encouragement for this project work. We would also like to express our gratitude to Dr. M. B. Khambete, Director, Cummins College of Engineering for Women, for providing us the opportunity to work on this project in the first place.

Finally, we take this opportunity to extend our heartfelt gratitude to all the staff members of the Department of Electronics and Telecommunications Engineering and thanks to all those who helped us directly or indirectly in many ways towards the project.

Shruti Nair (C22018111067)
Meera Narale (C22108111068)
Rajvee Pisey (C22018771865)

# Abstract

Generative Adversarial Networks are a fascinating and rapidly evolving field that fulfils the promise of generative models by generating realistic examples in a variety of problem domains, most notably image-to-image translation. They are a clever way to train a generative model by framing the problem as a supervised learning problem with two sub-models: the generator model, which we train to generate new examples, and the discriminator model, which tries to classify examples as real (from the domain) or fake (generated).

Going to a store and trying on clothes has become even more difficult as a result of the COVID-19 situation. Because of changing consumer demands, the store of the future will be very different from what we are used to. Consumers can virtually try on clothes thanks to digital solutions, which will not only improve their shopping experience and change the way people shop for clothes, but will also save retailers money.

Virtual clothing try-on is in major demand in recent years, with the goal of transferring a desired clothing image onto a reference person. The goal of the typical try-on assignment is to match the target clothing item with the provided individual's physique and hence present a try-on look of the person.

We propose a pose-guided virtual try-on technique based on different generative adversarial networks to achieve this goal. Experiments would be conducted on the dataset that is used in VITON and a few images of our own. It contains image pairs, each of which includes a front-view woman image and a top clothing image.

Thus, we will be generating new outfits onto existing human model images while retaining the body shape and pose of the wearer.

# Index

# List of Figures

# List of Tables

# 1. Introduction

Virtual clothing try-on is in major demand in recent years, with the intent of moving an ideal apparel picture onto a reference individual. The objective of the typical try-on assignment is to match the target clothing item with the gave person's build and thus present a try-on look of the individual. Practically speaking, however, individuals might be more intrigued by their take a try-on looks with different postures.

We propose a posture directed virtual try-on strategy in light of based on generative adversarial networks (GANs) to accomplish this objective. By first foreseeing the semantic layout of the reference image, we can determine whether it will be changed after try-on.

Two models comprise GAN, G and D which is the generative model and discriminative model respectively. The G model captures the information distribution, whereas the D model estimates the likelihood that a sample was drawn from the training data rather than G. The training procedure for G is intended to improve the probability of D making a mistake. GANs are used to visualise designs, to translate text to pictures, to transfer artwork style, to generate face images and video game scenes, to create super-resolution images, and so on.

The model predicts semantic format of the input picture and the changes will be seen after try-on, and afterward we will decide if its picture content must be created or preserved by the expected format of the semantic, which will result in photo-realistic try-on and rich attire details. Fashion synthesis may be a difficult process that needs the placement of a reference article of clothing on a input model in an arbitrary posture wearing a special garment.

Experiments would be conducted on the VITON dataset and a few of our own images. It features pairs of images; each pair consists of front-view of a women and an image of top piece of clothes image.

## 1.1.**Objective:**

The project will demonstrate the effectiveness of our approach through Structural Similarity Index Measure (SSIM) and developing a user interface for generating new clothing on images of the wearer through generative adversarial learning.

## 1.2.**Motivation:**

The main motive of this project arose from our interest in the domain of Artificial Intelligence and Machine Learning to enhance our knowledge around it. Due to the COVID-19 pandemic, going to a store and trying on clothes had become a greater problem. With consumer demands changing, the stores of the future will look vastly different from what we are used to.

Consumers may virtually try on garments using digital solutions, which not only improves their shopping experience by changing the way people browse for clothes, but is also cost effective for retailers. In recent years, virtual clothing try-on has grown in popularity, with the purpose of placing a desired clothing picture onto a reference person. However, individuals might be keener on their try-on looks with different postures.

Generative Adversial Network is an unsupervised modeling approach in Machine Learning. We researched more about GAN and found it to be interesting and compelling to learn new technologies. Thus, we could do a real-life implementation of a virtual try-on in deep learning which is a completely new domain for us.

## 1.3.**Approach:**

Among various GANs available, we have implemented the TRY-ON using Poly-GAN. First, train Poly-GAN with data, where inputs are the images of a model wearing a garment and another input garment (top only). In the following stage, to infer the appearance flows between person representation (RGB model (Parsed)) and the garment image, we conjugate a mask containing the lower body clothes region (garment mask), hair and face, and the individual's body segmentation outcome, and the human posture assessment outcome as the individual representation. Then, at that point, the RGB skeleton is utilized to generate the shape of the wrapping garments. We train a generative module with these wrapped garments, the conserved areas on the model picture, and human posture estimates along channels as inputs to get the person image with the new outfit as our result.

# 2. Literature Survey

The survey is done on the basis of papers published and the available products. Following are the details:

*Table 2.1* Papers referred

| Sr. No. | Title of the Research Paper | Authors | Journal/ Conference in which the Paper is published | Year of Publication | Hyperlink of the Soft-Copy of the Paper |
|---|---|---|---|---|---|
| **1.** | Poly-GAN: Multi - conditioned GAN for fashion synthesis | Nilesh Pandey, Andreas Savakis | Neuro-computing Volume 414, 13, Pages 356-364 | 2020 | https://arxiv.org/pdf/1909.02165.pdf |
| **2.** | Be Your Own Prada: Fashion Synthesis with Structural Coherence | Shizhan Zhu, Sanja Fidler, Raquel Urtasun, Dahua Lin Chen Change Loy | IEEE International Conference on Computer Vision (ICCV) | 2017 | http://personal.ie.cuhk.edu.hk/~ccloy/files/iccv_2017_fashiongan.pdf |
| **3.** | Parser-Free Virtual Try-on via Distilling Appearance Flows | Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, Ping Luo | IEEE/ CVF Conference on Computer Vision and Pattern Recognition (CVPR) | 2021 | https://openaccess.thecvf.com/content/CVPR2021/papers/Ge_Parser-Free_Virtual_Try-On_via_Distilling_Appearance_Flows_CVPR_2021_paper.pdf |

| | | | | | |
|---|---|---|---|---|---|
| **4.** | TryOnGAN: Body-Aware Try-On via Layered Interpolation | Srivatsan Varadharajan, Kathleen M Lewis, Ira Kemelmacher-Shlizerman | ACM Transactions on Graphics, Volume 40, Issue 4 | 2021 | https://dl.acm.org/doi/10.1145/3450626.3459884 |
| **5.** | Changing clothing on people images using generative adversarial networks | Yevhen Pozdniakov | Ukrainian Catholic University | 2020 | https://er.ucu.edu.ua/bitstream/handle/1/1904/Pozdniakov%20-%20Changing%20Clothing%20on%20People%20Images.pdf?sequence=1&isAllowed=y |
| 6. | Improved Techniques for Training GANs | Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, Xi Chen | NIPS'16: Proceedings of the 30th International Conference on Neural Information Processing Systems | 2016 | https://dl.acm.org/doi/pdf/10.5555/3157096.3157346 |
| 7. | Virtual Dressing Room Application | Aladdin Masri, Muhannad Al-Jabi | IEEE Jordan International Joint Conference on Electrical Engineering & Information Technology (JEEIT) | 2019 | https://ieeexplore.ieee.org/document/8717410 |

**[1] Nilesh Pandey, Andreas Savakis, *Poly-GAN: Multi - conditioned GAN for fashion synthesis*, 2020** He stated that Poly-GAN is a multi-input conditioning algorithm that may be used for a variety of applications, including picture alignment, stitching, and inpainting. On the basis of the model's RGB skeleton in any posture, Poly-GAN may conduct a spatial modification of the clothing. Using the DeepFashion dataset, the Inception Score and Structural Similarity Index yielded measurable findings.

**[2] Shizhan Zhu, Sanja Fidler, Raquel Urtasun, Dahua Lin Chen Change Loy, *Be Your Own Prada: Fashion Synthesis with Structural Coherence*, 2017** They proposed a model for solving an image problem: generating fresh clothing for people's photos. This job appears to be critical for the offline/online retail and fashion industries. Changing the clothes on people's photos isn't easy. The image created should be of good quality and free of blur. The drawback is inability to produce long sleeves on photos, such as with T-shirts.

**[3] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, Ping Luo, *Parser-Free Virtual Try-on via Distilling Appearance Flows*, 2021** "Teacher-tutor-student" knowledge distillation is a novel approach for producing profoundly photograph sensible pictures without the requirement for human parsing. The technique uses the parser-based method's artificial photos as "tutor knowledge," with the artifacts being rectified in a self-supervised way by actual "Teacher knowledge" acquired from real human photographs. Extensive evaluations show that the method outperforms previous arts approaches by a wide margin.

**[4] Kathleen M Lewis, Srivatsan Varadharajan, Ira Kemelmacher-Shlizerman, *TryOnGAN: Body-Aware Try-On via Layered Interpolation*, 2021** They demonstrated that internal interpolation coefficients per layer can be learned using StyleGAN2 to create an experience of try-on. While encouraging, the technique falls short in postures and clothing that are underrepresented. Because interpolation assumes perfect projection, poor projection of real images has a direct impact on interpolation results.

**[5] Yevhen Pozdniakov, *Changing clothing on people images using generative adversarial networks*, 2020** Built on generative adversarial learning, it has offered a unique technique for producing new clothes on a person. The model "redresses" a person based on a picture of the person and a statement suggesting an alternative attire. At the same time, the wearer's position and attitude remain unaltered. Virtual Try-on Network and Liquid Warping GAN models were used to change the apparel on the people's photos.

**[6] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, *Improved Techniques for Training GANs*, 2016** It provides a set of ideas for expanding

on the architectural parameters outlined in the DCGAN paper. It facilitates the understanding of the instability of GAN's most successful theory. Many strategies are provided to stabilize DCGAN training. That includes matching feature, discrimination of minibatch, historical averaging, virtual batch normalization, and one-sided label smoothing. Adding these recommendations to a basic implementation of DCGANs might be a wonderful way to learn more about GANs.

**[7] Aladdin Masri, Muhannad Al-Jabi, *Virtual Dressing Room Application*, 2019** He showed one of the most important steps in solving the problem is detecting the user and body parts. In the user interface, for detecting body parts, monitoring skeletal movements, and estimating posture. The Project is written in C\# and is intended to be a real-time, Kinect hacking application. In conjunction with Microsoft Kinect, the middleware of the Kinect driver is utilized for a range of core operations as well as the tracking process. Presents a virtual dressing room application that makes use of the Microsoft Kinect sensor.

Table 2.2 contains details of the similar technologies currently in use.

*Table 2.2* Existing technologies

| Sr. No. | Name of the existing product | Specifications & Features | Place where it is in-use | Cost | Hyperlink to its website |
|---|---|---|---|---|---|
| **1.** | TryNDBuy | Enables online shoppers to wear the clothes on their virtual body model to check size, fit, and look. | Celio*, Vero Moda, Jack n Jones | Depends on users & active products | https://tryndbuy.com/ |
| **2.** | TriMirror | Enables consumers and designers to experience real-life clothes on their accurate virtual models in motion, as well as the instant fit visualization on online, desktop, or mobile platforms. | E-commerce | Licensed | https://www.trimirror.com/ |

# 3. Specifications

## 3.1. Software Requirements:

3.1.1. **Language:** Python

3.1.2. **IDE:** Any Python programming IDE

3.1.3. **OS:** Windows 7 of higher

3.1.4. **Database:** MySQL

3.1.5. **Server:** XAMPP Server

3.1.6. **Other Tools/ Libraries used:** tensorflow, os, sys, gc, argparse, random, cv2, numpy, json, torch, skimage, PIL, matplotlib, django, tkinter

3.1.7. **Dataset:** A few of our images along with VITON Dataset
https://www.kaggle.com/rkuo2000/viton-dataset

## 3.2. Hardware Requirements:

3.2.1. **Processor:** i3 or higher processor

3.2.2. **Memory:** 1 GB

3.2.3. **Hard Drive:** 50 GB

# 4. Methodology

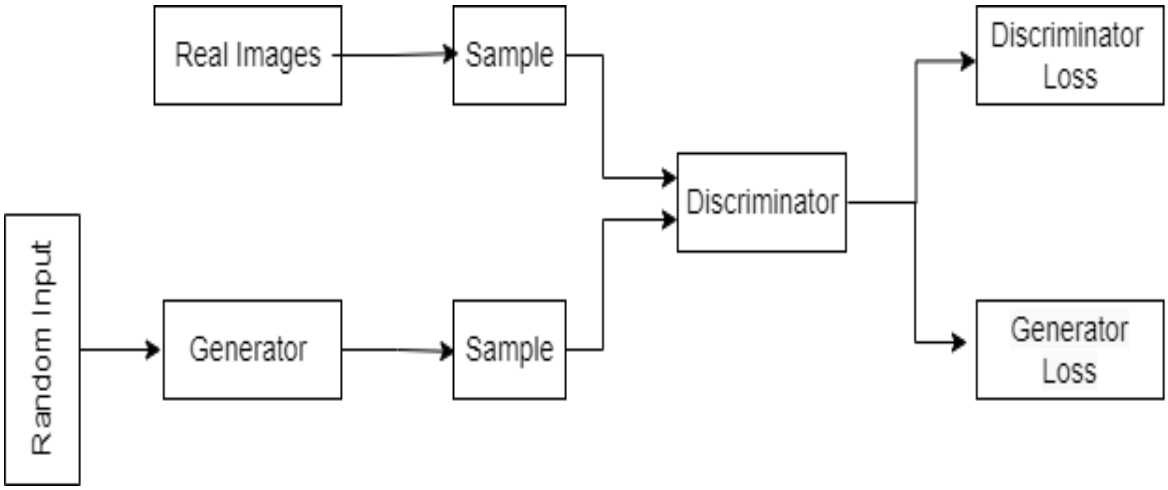


*Figure 4.1* General GAN Architecture

The figure 4.1 shows Generative Adversarial Network (GAN) which comprises of G and D models – generative model and discriminative model. The discriminative model calculates the likelihood that a sample originated from the training data rather than the generative model, whereas the generative model captures the data distribution. The architecture is analogous to a minimax game of two-player, using G's training approach maximising the likelihood of D committing an error.

From the literature survey we came across various types of GAN few being DCGAN, CycleGAN, StytleGAN, ACGPN and Poly-GAN. DCGAN is a Deep Convolutional GAN and it is one of the most widely used, powerful, and successful GAN architecture types. ConvNets are used instead of a Multi-layered perceptron to implement it. ConvNets are built with a convolutional stride and no max pooling, and the layers in this network are not completely connected.

The CycleGAN model is intended to solve the problem of image-to-image conversion. The image-to-image translation challenge’s purpose is to use a training set of matched picture pairings to learn the mapping between an input image and an output image. Getting matched samples, on the other hand, isn't always possible. CycleGAN uses cycle-consistent adversarial networks to learn this mapping without requiring paired input-output images.

A GAN for virtual try-on apparel applications is the Adaptive Content Generating and Preserving Network (ACGPN). It is divided into three sections. First, following try-on, a semantic layout generating module follows the anticipated semantic layout using semantic segmentation of the input image. Second, the clothes warping module warps clothing images based on the semantic layout established, employing a second-order difference constraint to make the warping process consistent throughout training. Third, a content fusion inpainting module integrates all necessary data (for example, reference image, semantic layout, deformed clothes) to adaptively create each semantic section of the human body. When compared to current approaches, ACGPN can produce photorealistic images with significantly higher perceptual quality and richer fine-details.

The StyleGAN is an expansion of the continuously evolving GAN that is a methodology for preparing generator models fit for orchestrating extremely enormous high-quality pictures during the training cycle through the steady expansion of both discriminator and generator models from small to large images.

The peculiarity of Poly-GAN permits us to work on various inputs and ideal for a wide range of applications such as inpainting, stitching and image alignment.

Current solutions employ a three-stage channel to align the garment with the human posture, sewing the aligned garment, and then refining the resulting image. Poly-GAN simplifies the procedure by doing all three jobs in a single architecture. This design upholds the criteria on the encoders' layers and employs skip connections from the coarse layers of the encoder to the decoder's appropriate layers.

Poly-GAN can conduct a dimensional change of the clothing based on the model's RGB skeleton in any posture. When the garment mask contains irregular holes, Poly-GAN may also execute image stitching and inpainting on it, inconsiderate of the clothes orientation.

The Try-On model would anticipate the input image's sematic layout and decide if its image content needs to be developed or maintained, resulting to virtual photo-realistic try-on.

# 5. Detail Design

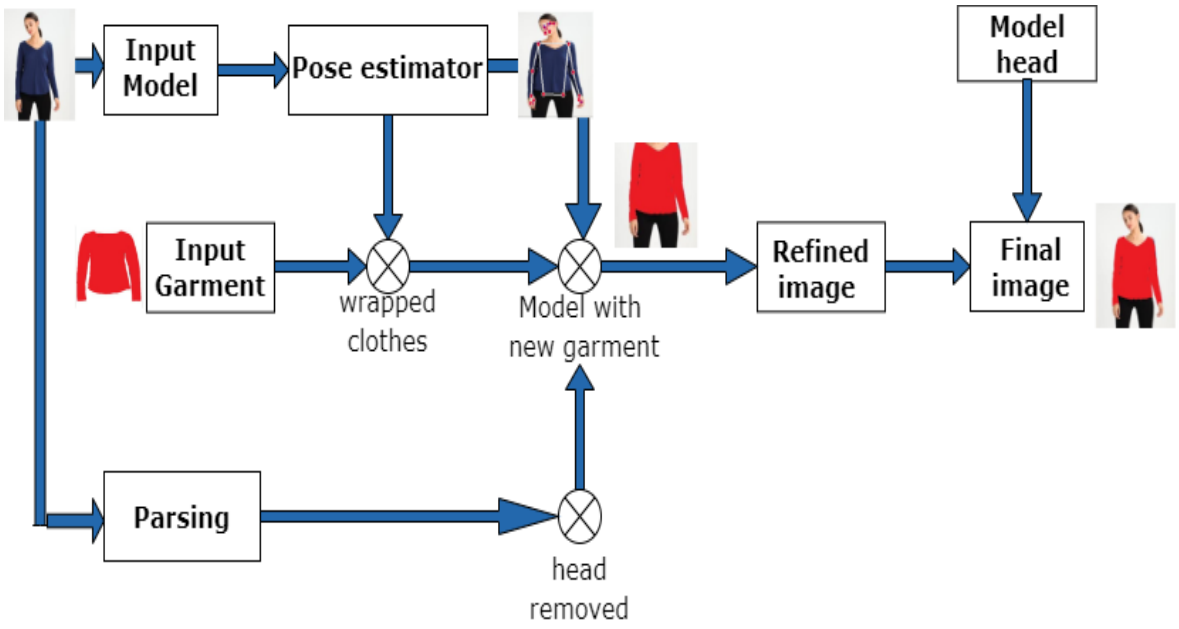


*Figure 5.1* Block Diagram

The design (as shown in Fig. 5.1) is implemented using Poly-GAN to achieve the objective of try-on.

- Input Model: The image of the reference model is taken as one input.
- Input Garment: The image of the reference garment is taken as another input.
- Pose Estimator (Skeleton): Leaving the head while keeping the hip and hands, an array of bone list created which is drawn to create the skeleton.
- Wrapped Clothes: The input garment takes the shape of the input model.
- Parsing: The input model is parsed by converting the image (which is in string format) into an integer array. The resultant image is converted to Cuda Tensor and its contrast is improved.
- Head Removal: The head is removed after parsing the input model image.
- Model with new Garment: The output of the shape stage is concatenated with the removed head image to get the stitched stage output.
- Refined image using Poly-GAN.
- Model head: The model head which was removed earlier is added to the refined model image.

- Final Image: After adding the head to the refined image, we get the final output as the model with new garment.

First, we use data to train Poly-GAN, with inputs being images of a model wearing a garment and another input garment (top clothing only). In the following stage, to infer the appearance flows between person representation (RGB model (Parsed)) and the garment image, we conjugate a mask containing the lower body clothes region (garment mask), hair and face, and the individual's body segmentation outcome, and the human posture assessment outcome as the individual representation. The warped clothes with garment images are generated using the RGB skeleton. The generative module is trained to get the model with new garment by concatenating these warped garments, the preserved areas on the model picture, and human posture estimation as inputs.

# 6. User Interface

The user interface was developed using libraries and various tools like Django and Tkinter along with MySQL as the database and the model was deployed on the local host with the help of XAMPP Server.

Django is an open-source and free high-level web framework based on Python that adheres to the model – template – views architectural paradigm and allows for the quick building of safe and stable websites. Tkinter is the standard Python GUI framework that gives clients a basic method for making GUI components utilizing the gadgets found in the Tk toolbox. Tk widgets may be used to generate buttons, menus, data fields, and so on in a Python application.

A database is a structured collection of information. Based on structured query language (SQL), Oracle's MySQL is an open-source relational database management system. XAMPP is a free and open-source cross-platform web server solution stack bundle built by Apache Friends. XAMPP is a well-known cross-platform web server that enables developers to build and run their work on a local webserver.

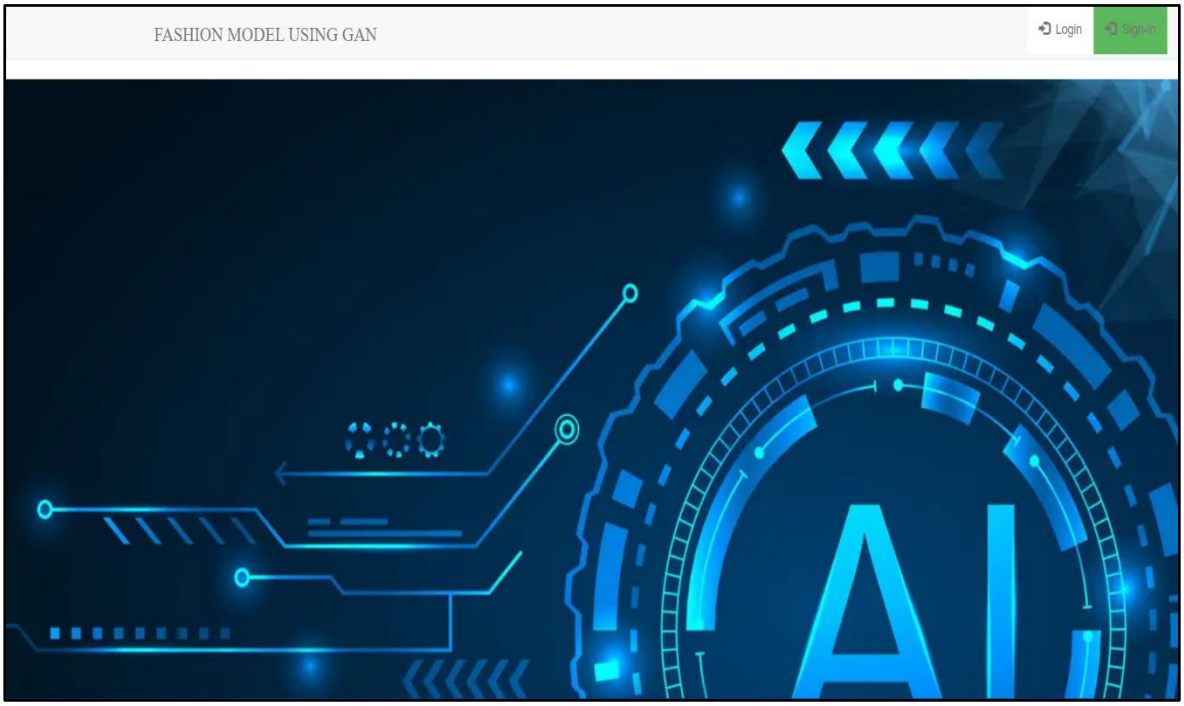


*Figure 6.1* Home Page

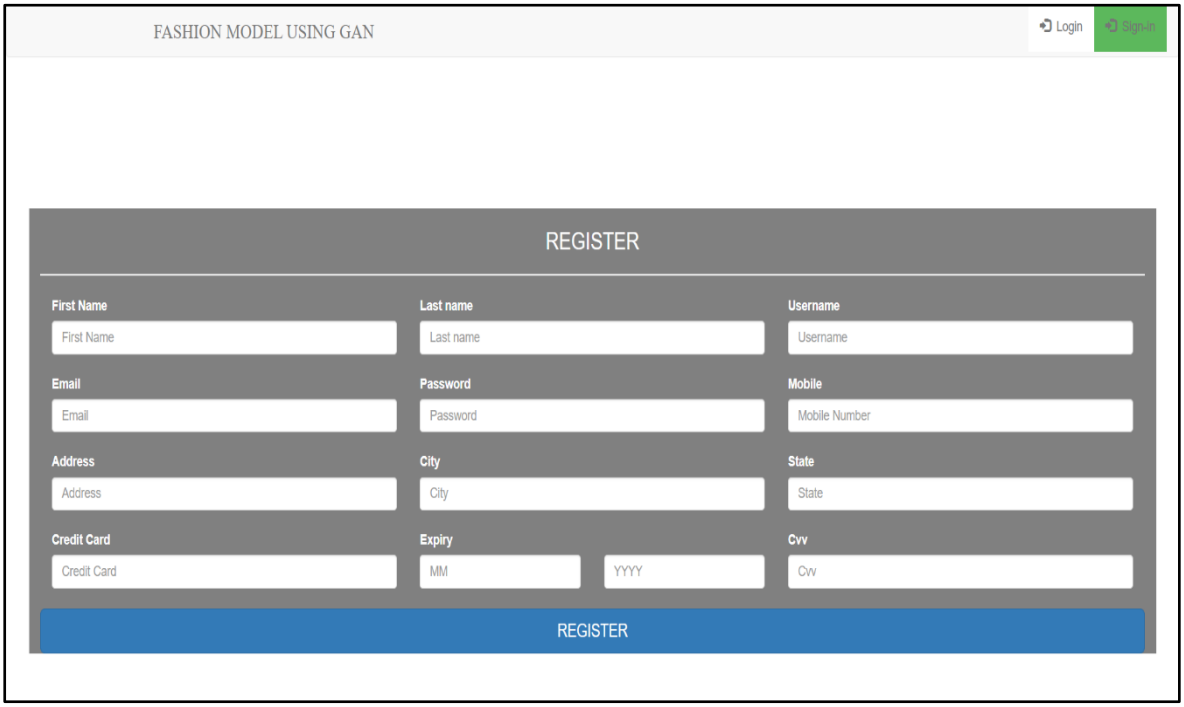


*Figure 6.2* New user registration window

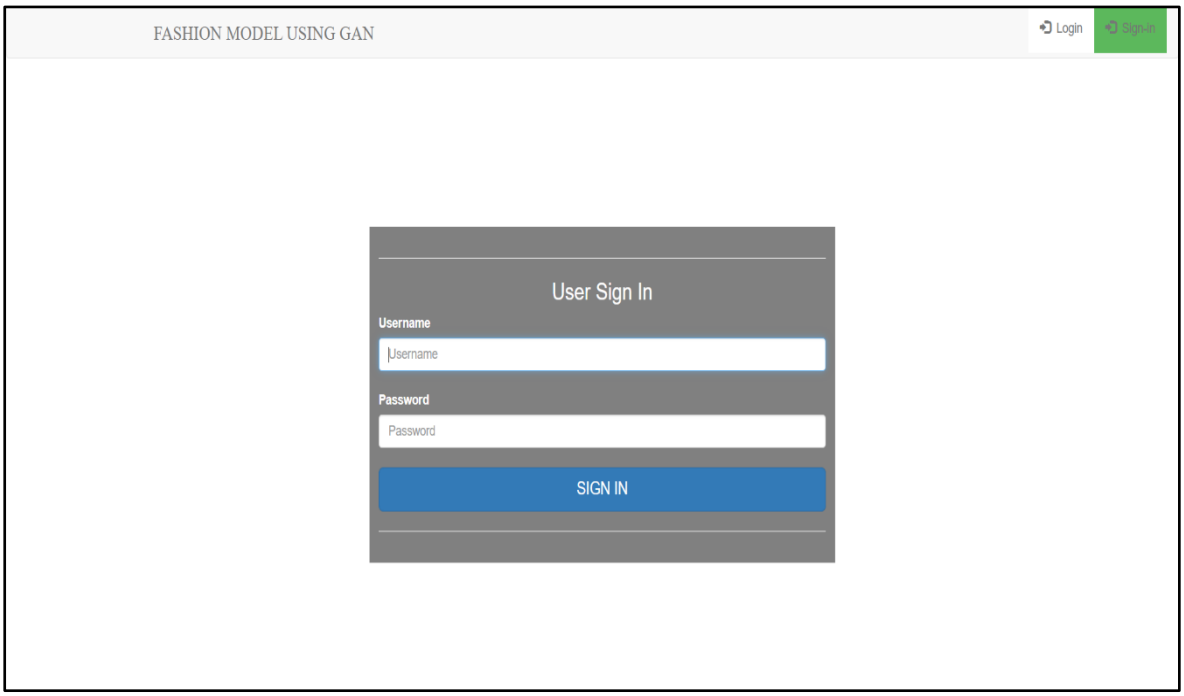


*Figure 6.3* User Sign In

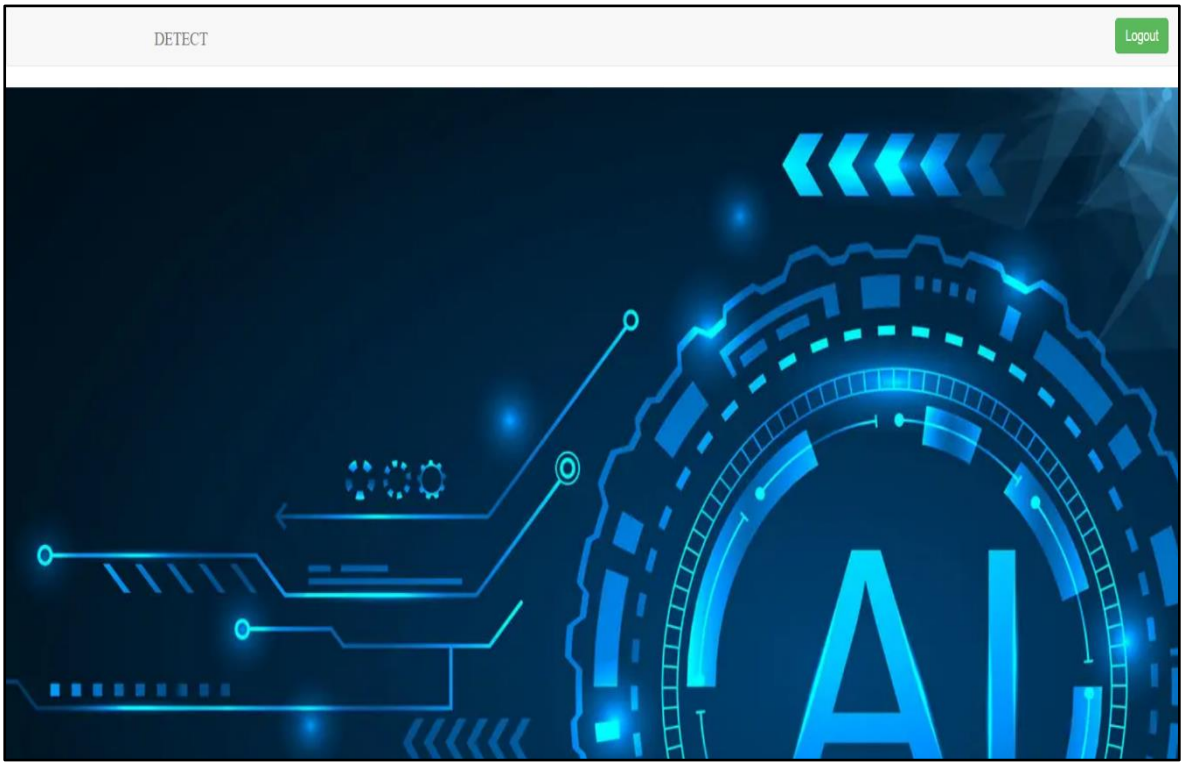


*Figure 6.4* Home Page after login

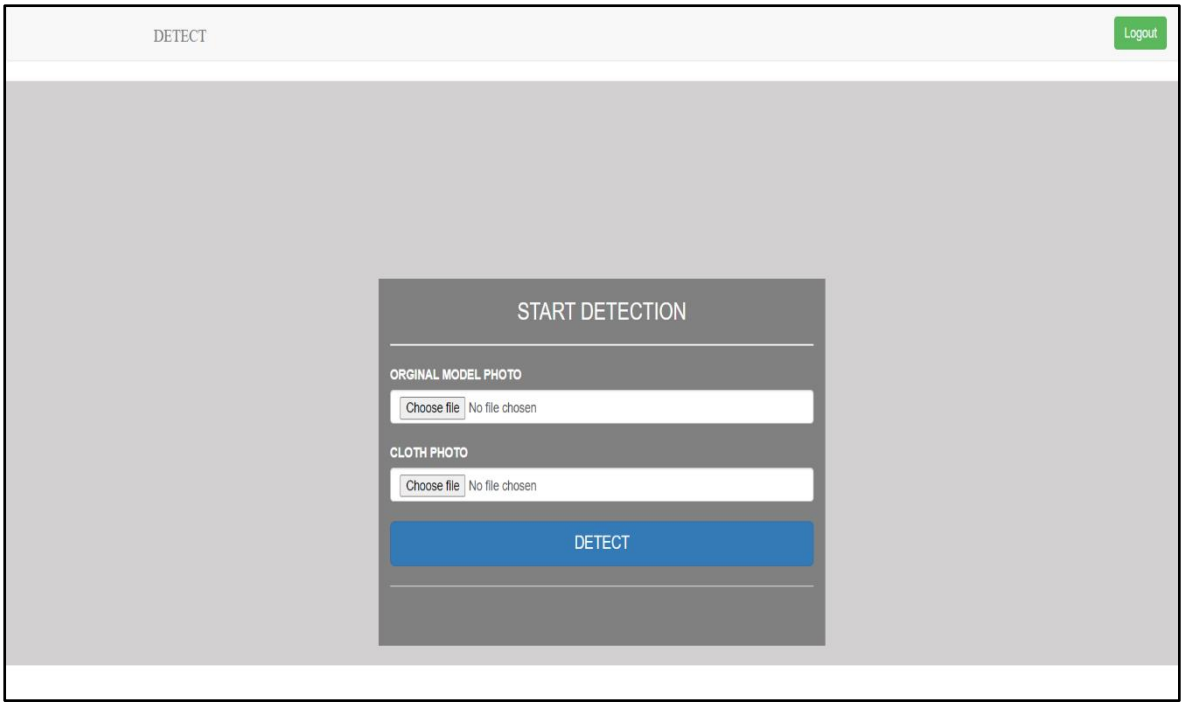


*Figure 6.5* Detection Window

# 7. Results


*Figure 7.1* Poly-GAN results from left to right for four test cases

Fig. 7.1 (b) shows the pose estimated (skeleton) of the input image of the model (as in Fig. 7.1 (a)). Fig. 7.1 (c) is the input image of the garment which takes the shape of the input model using the key points of the skeleton and is stitched on the input model as shown in Fig. 7.1 (d). Fig. 7.1 (e) depicts the final output image after refinement which is produced with the input model wearing the input garment.

Table 7.1 shows the Structural Similarity Index Measure (SSIM) of the two stages – Shape Stitch and Final Refined Output, for the four test cases. The model's accuracy was determined to be between 75% and 90%, indicating that it can successfully serve as a virtual try-on.

*Table 7.1* Accuracy Scores

| Sr. No. | Stage | Accuracy |
|---|---|---|
| (i) | Shape Stitch | 0.86 |
| | Final Refined | 0.83 |
| (ii) | Shape Stitch | 0.80 |
| | Final Refined | 0.84 |
| (iii) | Shape Stitch | 0.79 |
| | Final Refined | 0.76 |
| (iv) | Shape Stitch | 0.81 |
| | Final Refined | 0.87 |

Evaluation Parameter used is the Structural Similarity Index Measure (SSIM) which is a method for predicting the perceived quality of digital television and cinematic pictures, as well as other kinds of digital images and videos. It is a perceptual metric that quantifies image quality degradation caused by processing such as data compression or by losses in data transmission. SSIM is used for measuring the similarity between two images. The SSIM values ranges between 0 to 1, 1 means perfect match the reconstruct image with original one.

# 8. Conclusion

The Poly-GAN is trained with inputs being images of a model wearing a garment and another input garment (top clothing only). In the following stage, to infer the appearance flows between person representation (RGB model (Parsed)) and the garment image, we conjugate a mask containing the lower body clothes region (garment mask), hair and face, and the individual's body segmentation outcome, and the human posture assessment outcome as the individual representation. Then, at that point, the RGB skeleton is utilized to generate the shape of the wrapping garments. We train a generative module with these wrapped garments, the conserved areas on the model picture, and human posture estimates along channels as inputs to get the person image with the new outfit as our result.

The User Interface was deployed on local host using XAMPP Server with MySQL as a backend database. The accuracy using the Structural Similarity Index Measure (SSIM) of the model was found to be between 75% to 90% which indicates the model can effectively work as a virtual try-on. We conclude by highlighting the robustness of the implemented model as well as the high quality of the images produced.

# 9. Future Scope

- The model can be trained on other datasets in the future, such as for men’s clothing.
- The model has potential to improve the image generation for the reference person in various poses and views (side view, back view).
- The size of the apparel and the size of the reference person are two more factors that might be included in future work that will benefit customers.
- The model could be used to generate photo-realistic try-on images for online shopping and other websites.

# 10. Bibliography


[1] Nilesh Pandey, Andreas Savakis, "Poly-GAN: Multi - conditioned GAN for fashion synthesis", published in Neurocomputing Volume 414, 13, Pages 356 – 364, Nov – 2020 https://arxiv.org/pdf/1909.02165.pdf

[2] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu and Ping Luo, "Parser-Free Virtual Try-on via Distilling Appearance Flows", published 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) https://openaccess.thecvf.com/content/CVPR2021/papers/Ge_Parser-Free_Virtual_Try-On_via_Distilling_Appearance_Flows_CVPR_2021_paper.pdf

[3] Rahul Singh, Aman Bindal, Md Azad Khan, Rakshith MR and Prof. Divakara N, "Virtual Clothing Try-on Using Generative Adversarial Networks", published in IJARIIE-ISSN(O)-2395-4396, 2021 https://ijariie.com/AdminUploadPdf/Virtual_Clothing_Try_on_Using_Generative_Adversarial_Networks_ijariie14446.pdf

[4] Han Yang, Ruimao Zhang, Xiaobao Guo, Wei Liu, Wangmeng Zuo and Ping Luo, "Towards Photo-Realistic Virtual Try-On by Adaptively Generating Preserving Image Content" published in Cornell University, March 2020 https://arxiv.org/pdf/2003.05863.pdf

[5] Shion Honda, "VITON-GAN: Virtual Try-on Image Generator Trained with Adversarial Loss", published in Eurographics Digital Library, 2019 https://diglib.eg.org/bitstream/handle/10.2312/egp20191043/009-010.pdf

[6] Shizhan Zhu, Sanja Fidler, Raquel Urtasun, Dahua Lin Chen Change Loy, "2017 IEEE International Conference on Computer Vision (ICCV)", published in IEEE International Conference on Computer Vision (ICCV), 2017 http://personal.ie.cuhk.edu.hk/~ccloy/files/iccv_2017_fashiongan.pdf

[7] Kathleen M Lewis, Srivatsan Varadharajan, Ira Kemelmacher – Shlizerman," TryOnGAN: Body-Aware Try-On via Layered Interpolation", published in Cornell University, Jan 2021 https://arxiv.org/abs/2101.02285

[8] Gaurav Kuppa, Andrew Jong, Xin Liu, Ziwei Liu, Teng-Sheng Moh," ShineOn: Illuminating Design Choices for Practical Video-based Virtual Clothing Try-on", published in 2021 IEEE Winter Conference on Applications of Computer Vision Workshops(WACVW) https://www.computer.org/csdl/proceedingsarticle/wacvw/2021/196700a191/1sZ3rKHyrHa

[9] Yevhen Pozdniakov, Orest Kupyn," Changing clothing on people images using generative adversarial networks", Master Thesis under Ukrainian Catholic University https://er.ucu.edu.ua/bitstream/handle/1/1904/Pozdniakov%20-%20Changing%20Clothing%20on%20People%20Images.pdf?sequence=1&isAllowed=y

[10] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, Larry S. Davis," VITON: An Image-based Virtual Try-on Network", published in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) https://openaccess.thecvf.com/content_cvpr_2018/papers/Han_VITON_An_Image-Based_CVPR_2018_paper.pdf

[11] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen," Improved Techniques for Training GANs", published in NIPS'16: Proceedings of the 30th International Conference on Neural Information Processing Systems, December 2016 https://arxiv.org/abs/1606.03498

[12] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros," Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", published in 2017 IEEE International Conference on Computer Vision (ICCV), Oct 2017 https://arxiv.org/abs/1703.10593

[13] Aladdin Masri; Muhannad Al-Jabi," Virtual Dressing Room Application", published in IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), April 2019 https://ieeexplore.ieee.org/document/8717410

[14] VITON Dataset, https://www.kaggle.com/datasets/rkuo2000/vitondataset

# 11. Workplan

| Month / Week → / Task / Module | Sign. of Internal Guide | Sign. of the co-ordinator | NOV w1 | NOV w2 | NOV w3 | NOV w4 | DEC w2 | DEC w3 | DEC w4 | JAN w1 | JAN w2 | FEB w1 | FEB w2 | FEB w3 | FEB w4 | MAR w1 | MAR w2 | MAR w3 | MAR w4 | APR w1 | APR w2 | APR w3 | APR w4 | MAY w1 | MAY w2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. Literature Survey & concept development | | | ■ | ■ | ■ | | | | | | | | | | | | | | | | | | | | |
| 2.Specifications of entire system | | | | | | ■ | | | | | | | | | | | | | | | | | | | |
| 3. System - level design (Block Schematic) | | | | | | | ■ | | | | | | | | | | | | | | | | | | |
| 4. Real-life Layout of the System | | | | | | | ■ | | | | | | | | | | | | | | | | | | |
| 5. Detail design (Software programming of each Block) | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5.1. Dataset Collection | | | | | | | | ■ | | | | | | | | | | | | | | | | | |
| 5.2. Data Visualization and study of libraries | | | | | | | | | ■ | | | | | | | | | | | | | | | | |
| 5.3. Pose Estimation & Skeleton | | | | | | | | | ■ | ■ | | | | | | | | | | | | | | | |
| 5.4 Parsing & Head removal | | | | | | | | | | | | ■ | ■ | | | | | | | | | | | | |
| 5.4. Cloth wrapping | | | | | | | | | | | | | | ■ | | | | | | | | | | | |

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **5.5. Model training** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **5.6. Model testing** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **5.7. Performance evaluation** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **6. Validation of each module's Design through simulation** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **7. Seminar Report (with plagiarism check)** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **8. Internal Seminar** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **9. Integration & Testing of the Complete System** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **10. Internal Project Demonstration** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **11. Project Report (with plagiarism check)** | | | | | | | | | | | | | | | | | | | | | | | | | |

# 12. User Manual

Steps for stimulation:

1. Open XAMPP Control Panel to start modules Apache and MySQL.
2. Run the manage.py file using the command “python manage.py runserver” in command prompt.
3. Run the localhost server URL mentioned on a Brower to open the user interface window.
4. Sign In for first users or Login for already registered users.
5. For signing in, enter details like name, username, email, password, etc.
6. After logging in, click on “DETECT” to open the detection window.
7. Choose the model and garment image and click on DETECT.
8. A new window will open showing the input model, pose estimated (skeleton), input garment, the shape and stitched stage output and the final refined output showing the input model wearing the input garment.
9. Close the output pop-up window and the browser.
10. Quit the server with CTRL-BREAK on command prompt.
11. Stop the Apache and MySQL modules on XAMPP Control Panel.