IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

VIRTUAL CLOTH FITTING IN 2D USING DEEP LEARING

R.Palson Keneddy[1],Monishwar Roshan.B[2] ,Charan.M[3],Logesh[4], Jothi Krihnan.K[5]

Professor[1] ,Student[2,3,4,5]

Department of Computer Science and Engineering

PERI INSTITUTE OF TECHNOLOGY

ABSTRACT

In online shopping, People would like to know how they look in a particular dress they tend to buy. To bring closer between in-store (offline) and online shopping and provide a realistic shopping experience to the consumers, a system is required to predict how people look in a particular garment that they tend to buy. Traditional approaches to make such a system were dealing with computer graphics where a 3D avatar of the person is created and various clothes are visualized on him/her. The usage of such approaches limits practical applications due to high computation and hardware requirements. This Project aims to build a fitting system such that, it visualizes people wearing new clothes without any 3D information, only using Images, in the same pose as they are standing in. This Project is composed of two modules one for changing the shape of cloth given in the input according to the personality and another module to attach this cloth onto the individual body. Using Efficient CNN based networks, which learns to predict the parameters for changing shape via spatial transformation is used in the first module. The shape changed cloth is passed on to the next module which is based on U-net architecture that predicts the mask where to attach the cloth, using this information the cloth is attached onto the body for generating final output images. The system is able to generate natural and rich images without much loss.

Introduction

Online Shopping of Apparel has been in trend since the last decade and the ongoing years have seen the rise of demand for E-commerce fashion shopping, due to the convenience in shopping time and the wide availability of products at our disposal. Despite of such convenience online shopping provides there always exists an inconsistency that differentiates it from real-world shopping, which is the lack of physical trial of clothing apparel. The Lack of physical trial is one of top reasons for number of returns placed in fashion industry across E-commerce. The problem defined here can be solved with the help of developing a system that can visualize how the clothing realistically looks when we would wear in Real-world. Virtual fitting

systems are such that they predict/visualize new apparel on individuals. The presence of Such prediction systems brings a more realistic shopping experience to the users. To solve the problem of virtual fitting, initial approaches were in the 3D domain with the usage of computer graphics and simulation techniques. Various approaches were proposed in the 3D domain in which a deep image of the person is necessary, and the apparel is made to look at that person.

Since the cost of implementing such systems is high and applications are small-scale, there was an interest growing towards developing such fitting systems in the 2D domain, which deals only with images rather than depth. An approach has been provided by in the 2D domain, where they first predicted the blurry image of a person wearing a new dress and then passed onto another network to enhance the clothing area. The usage of shape-context-based shaping of dress results in poor transformation which fails to predict images for a wide variety of body shapes and poses. The problem with this approach is that it results in non-natural and poor /quality of dress details like logo, text, patterns on the dress. They focused mainly on the prediction of a person's image wearing a new dress without retaining proper details of clothing information. To solve the problem of virtual fitting in 2D to predict rich quality images where clothing information is visualized robustly. In this project, I propose to solve the problem using two separate networks. The first network is based on CNN for geometric matching for changing the shape of dress that entirely focuses on maintaining rich details in the dress and transforming it according to the individual's personality. The second network, which is based on Unet architecture that predicts a mask that is used for attaching the shape changed dress onto the individual. This predicts the output images where both the individual and also the cloth in him/her are in rich quality, which will be robust for predicting detailed apparels.

LITERATURE SURVEY

Enhance the clothing area. The usage of shape-context-based shaping of dress results in poor transformation which fails to predict images for wide variety of body shapes and poses. The problem with this approach is that it results in non-natural and poor quality of dress details like logo, text, pattern's on the dress. They focused mainly on the prediction of a person's image wearing a new dress without retaining proper details of clothing information. To solve the problem of virtual fitting in 2D to predict rich quality images where clothing information is visualized robustly. In this project, I propose to solve the problem using two separate networks. The first network is based on CNN for geometric matching [3] for changing the shape of dress that entirely focuses on maintaining rich details in the dress and transforming it according to the individual's personality. The second network, which is based on U-net architecture [8] that predicts a mask that is used for attaching the shape changed dress onto the individual. This predicts the output images where both the individual and also the cloth in him/her are in rich quality, which will be robust for predicting detailed apparels.

Early related works are based on traditional 3D computer graphics which render the human body-doubles/avatars using body scans [4, 5] and simulating different apparels on the avatars. A motion model was developed by [9] that shows promising results for real time environments.

Similar approaches [6] that infer the personality of humans using 3d body scans and depth sensors and adjusting 2D dress images on them were proposed. The practical implementations of such were developed by a fashion store [10] which have used AR technology to mark the cloth on individual but it is only limited to in-store application.

These above-mentioned approaches require high computation cost and need of special devices to capture the information required and also have limited applicability. The interest is growing towards more computationally efficient systems since the efficiency per cost and the device installation steps has hindered its application in large scale development like integration of such systems in E-commerce.

In image-based models [11] proposed a GAN based approach that swaps the dress onto the individuals by treating it as an image analogy problem, but it is not practical as it is not suitable for predicting images in any pose, requiring an image of humans wearing the cloth rather than only image of cloth. but in reality there may be no person who has worn that particular dress. In recent approaches [12] have used shape matching based hand-crafted networks to change the shape of dress, additionally using in-painting methods to further improve the final result. multiple spatial transformations are combined and a single image is formed for proper clothing deformation.

SYSTEM DESIGN

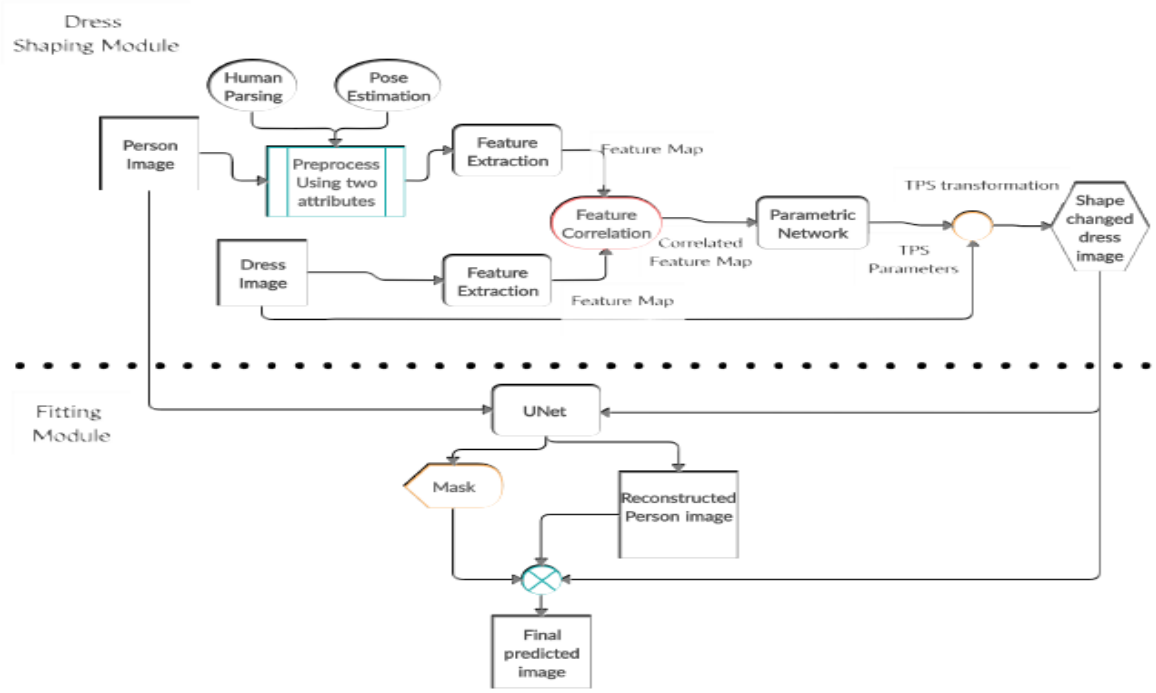

Fig 1: System design

System Architecture

The upper part of the figure is the Dress Shaping module. First the person image is preprocessed based on two attributes: pose estimation, human parsing which are explained. This preprocessed image along with the clothing image is given as input into first module which consists of three parts which are elaborated

- Feature Extraction
- Feature Correlation
- Parametric network and transformation

The output of this module is a shape changed dress image which is passed on to the next module along with the preprocessed person image into the fitting module. The lower part of figure is the Fitting module that predicts the reconstructed person image which highlights the regions where cloth should be fit and a mask that tells us regions in dress which are important for final predicted images of individual wearing new dress.

ALGORITHM

The following algorithms and optimization techniques are to be used:

- **Thin-plate-spline transformation [15, 16]** Thin-plate-splines Technique is used to provide a smooth interpolation between a set of control points. The surface is generated such that it is bent least (minimizing bend energy). The mapping function for any point (x,y) is given as:

$$F(x,y) = a_1 + a_2x + a_3y + \sum_{i=1}^{n} W_i U(|P_i - (x,y)|) \qquad (2.1)$$

The solution to this function has a closed form which can be solved using a system of linear equations as it is differentiable. TPS transformation is used widely in image translation applications where it is necessary to transform the shape of source image such that it matches into target image. TPS is an interpolation technique that provides a surface passing through all the points such that the bending energy U(jPi □ (x; y)j) is least possible.

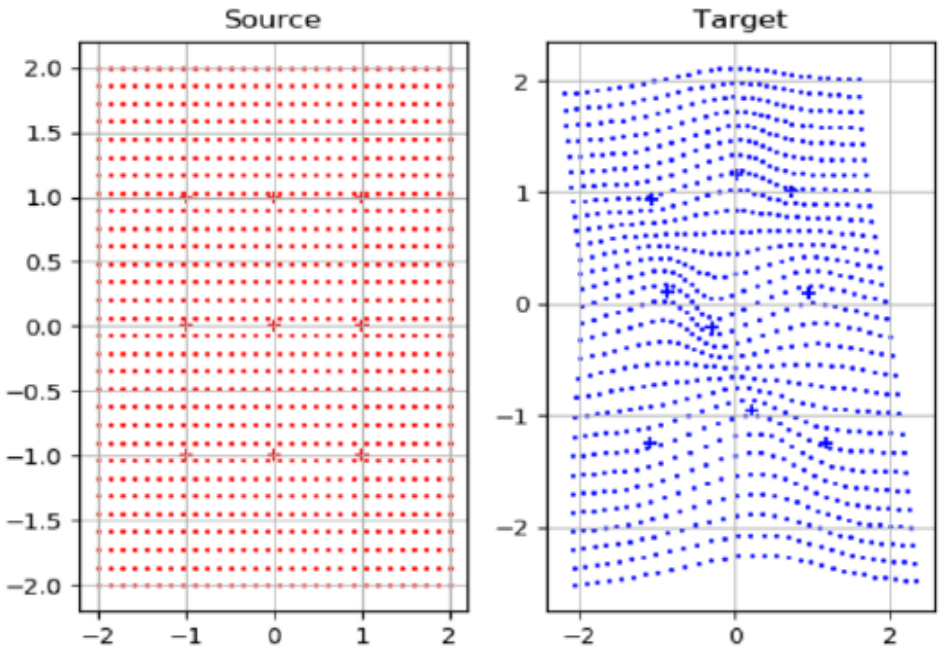

Fig 2: Thin Plate Spline Example

The fig 2.2 illustrates an example of such transformation. In the left part of the figure there are 9 control points in the source and based on the changes in co-ordinates of those 9 control points in the target the entire target surface is interpolated. If the change in coordinates of control points is in +ve x-direction then all the points are brought closer in x- direction, and if change is in -ve x-direction then the all points are little far compared to the source. In the figure due to various changes in directions of control points coordinates all the remaining points are displaced accordingly to the equation (2.1).

1. **Adam Optimizer [17]** Adam is a popular algorithm to generate results faster. In traditional machine learning algorithms in back-propagation step only learning rates are used which are modified in modern algorithms using RMSprop and gradient descent with momentum. Adam combines the advantages of both of them and helps in faster learning.
2. **Batch Normalization [18]** This technique helps in maintaining stability across the neural network pipeline and improves the speed of training the model. It helps in standardizing the input when several layers are used in a neural network. It helps in coping up with changes in distribution in input data by normalizing output of the previous layer by subtracting by mean and dividing by the standard deviation. The mean and standard deviation are learnable.
3. **Convolutional Neural Network (CNN) [19, 20]** This is most widely used deep neural network used in computer vision problems. Convolution is the simple application of a filter to an input that results in activation. Using convolutions it can extract features, in a multi-layered CNN network different layers extract different features such that selecting them minimize the loss function used. A loss function is generally used in CNN to make the model learn itself on what should the features(pixels) be selected based on activation values in kernels/filters at each layer.

Convolutional Neural Network is a deep learning algorithm that identifies and arranges the characteristics in images for computer vision [5]. This kind of model is used in our project that predicts the TPS parameters for establishing fine shape of dress and also generating final fitting images in later steps.

4. **U-Net Architecture [8]** U-Net was originally developed and used for segmentation tasks in biomedical images. Segmentation is a task where it labels the image into different classes. Since then, many segmentation tasks have used UNet architecture, where it is required to localize the target subject in an image like identify the region where a particular object is in the image. The original publication of U-Net was intended to solve the problem of localizing or making boundaries around the disease cells in biomedical images. The architecture consists of an encoder network followed by a decoder network, with connections between both of them thus forming a u-shape like appearance. The fitting module in our project is of U-Net architecture, which is required to predict the mask, that represents important regions of dress and person that is used for generating final output images.
5. **Perceptual Loss [1]** The traditional loss functions use the L1 or L2 losses which are the average of differences or mean squared errors. To compare the difference between two images using such losses for more complicated tasks like image super resolution and neural style transfer does produce blur images. The authors of perceptual loss have demonstrated that high quality images can be generated compared to using traditional L1 or L2 loss. In perceptual loss instead of calculating the difference between output images it calculates the difference in values of high-level representations like feature maps. We will use perceptual loss in our project to make high-quality output images.

The method of calculating such loss is as depicted in figure below:

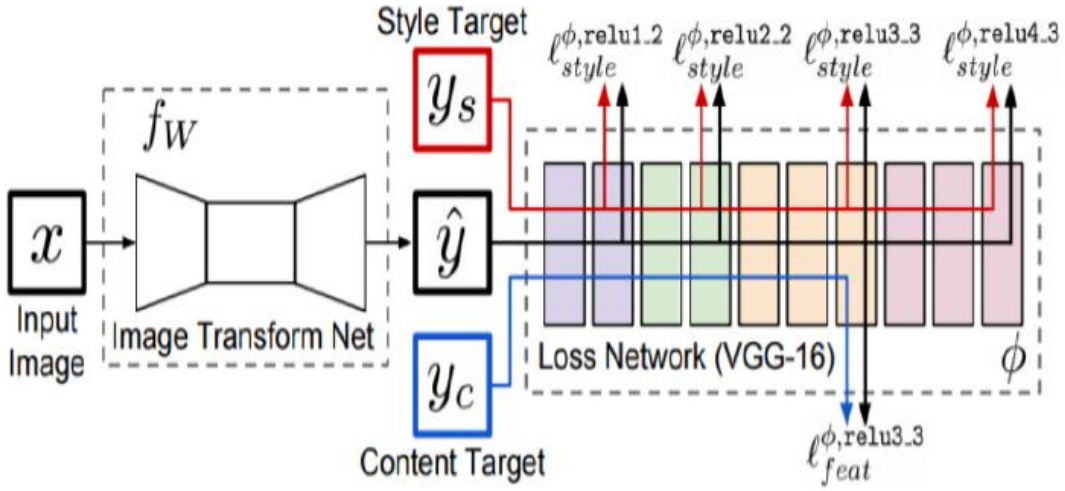

Fig 3: Method of calculating perceptual loss

In the fig 2.3 it represents the method for calculating perceptual loss. It consists of two networks. The left part is the network which is to be trained which is a model to solve our task and the right part is said to be a loss network. In our project the loss network used is pretrained VGG-16 model. The predicted output of our model ˆy for the input x is passed through loss network and the loss is calculated as below: We are not concerned about ys in the figure as it doesn't relate in the task of our project.

$$L^{j}_{perc}(y, \hat{y}) = \frac{\|\eta_j(\hat{y}) - \eta_j(y)\|_2}{V_j} \qquad (2.2)$$

ηj(x) is the feature map at the jth layer in the loss network which in our case is VGG-16, Vj is the volume of input at that particular layer. This difference in y with ˆy, forces ˆy to be similar to y in terms of values at every pixel. Rather than traditional L1,L2 losses this is more effective which has been demonstrated by the author [1].

- **VGG-16 network [2, 21]:** VGG is a convolutional Neural network which is initially developed to solve image classification tasks. VGG-16 is a very deep convolutional neural network which has 16 trainable layers. There are many versions of VGG with different numbers of neural layers. The authors demonstrated how using deeper layers affects the quality and accuracy of classification tasks. There are millions of parameters in VGG. The architecture of VGG-16 is depicted as below:

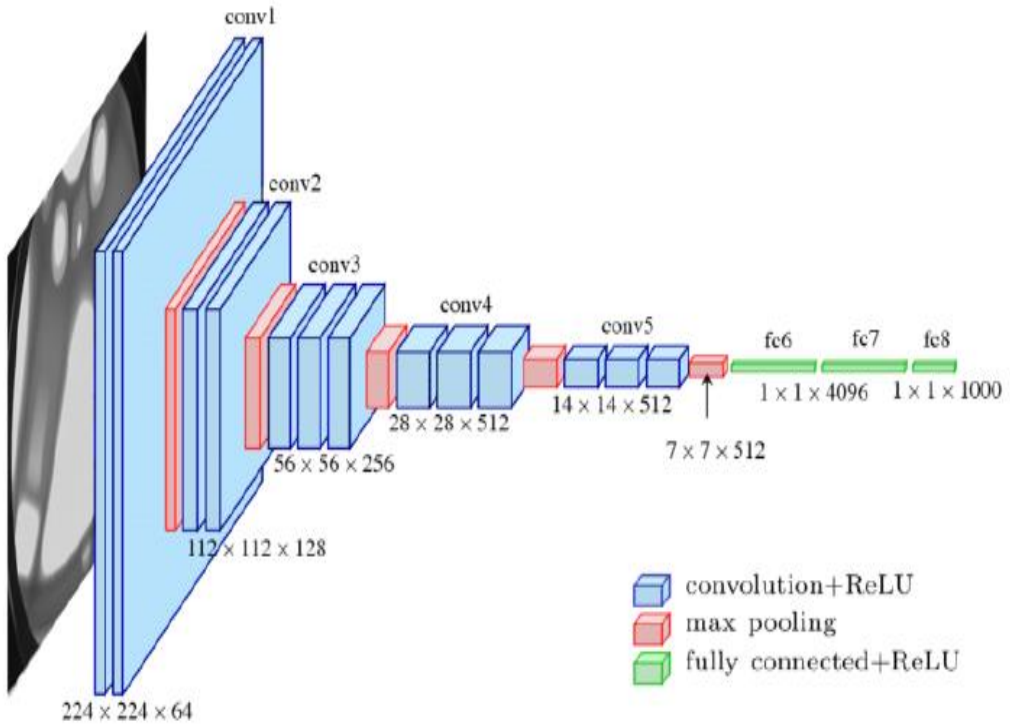

Fig 4: Convolutional process

VGG-16 architecture

The convolution and fully connected layers form the layers which have trainable weights and are 16 in number displayed as blue and green layers in figure 2.4. The number of kernels/filters used in subsequent layers are in incremental manner from 64 filters in the first layer to 512 filters in the last layer followed by fully connected layers. In our project we use the VGG-16 pretrained model to calculate the perceptual loss in the fitting module.

SYSTEM IMPLMENTATION

SOFTWARE DESCRIPTION

The purpose of the Software Requirement Specification is to produce the specification of the analysis task and also to establish complete information about the requirement, behavior and also the other constraint like functional performance and so on. The main aim of the Software Requirement Specification is to completely specify the technical requirements for the software product in a concise and in unambiguous manner.

PYTHON

In technical terms, Python is an object-oriented, high-level programming language with integrated dynamic semantics primarily for web and app development. It is extremely attractive in the field of Rapid Application Development because it offers dynamic typing and dynamic binding options.

Python is relatively simple, so it's easy to learn since it requires a unique syntax that focuses on readability. Developers can read and translate Python code much easier than other languages. In turn, this reduces the cost of program maintenance and development because it allows teams to work collaboratively without significant language and experience barriers.

Additionally, Python supports the use of modules and packages, which mean that programs can be designed in a modular style and code, can be reused across a variety of projects. Once we've developed a module or package we need, it can be scaled for use in other projects, and it's easy to import or export these modules.

One of the most promising benefits of Python are that both the standard library and the interpreter are available free of charge, in both binary and source form. There is no exclusivity either, as Python and all the necessary tools are available on all major platforms. Therefore, it is an enticing option for developers who don't want to worry about paying high development costs.

Python Usage

Python is a general-purpose programming language, which is another way to say that it can be used for nearly everything. Most importantly, it is an interpreted language, which means that the written code is not actually translated to a computer-readable format at runtime. Whereas, most programming languages do this conversion before the program is even run. This type of language is also referred to as a "scripting language" because it was initially meant to be used for trivial projects.

The concept of a "scripting language" has changed considerably since its inception, because Python is now used to write large, commercial style applications, instead of just banal ones. This reliance on Python has grown even more so as the internet gained popularity. A large majority of web applications and platforms rely on Python, including Google's search engine, YouTube, and the web-oriented transaction system of the New York Stock Exchange (NYSE).

Python can also be used to process text, display numbers or images, solve scientific equations, and save data. In short, it is used behind the scenes to process a lot of elements we might need or encounter on our device(s) - mobile included.

GOOGLE COLAB

In this project, google colab is used as an open-source IDE.

Google Collaboratory is a free online cloud-based Jupyter notebook environment that allows us to train our machine learning and deep learning models on CPUs, GPUs, and TPUs.

It gives us a decent GPU for free, which we can continuously run for 12 hours. For most data science folks, this is sufficient to meet their computation needs.

Google Colab gives us three types of runtimes for our notebooks:

- CPUs,
- GPUs, and
- TPUs

Colab gives us 12 hours of continuous execution time. After that, the whole virtual machine is cleared and we have to start again. We can run multiple CPU, GPU, and TPU instances simultaneously, but their resources are shared between these instances.

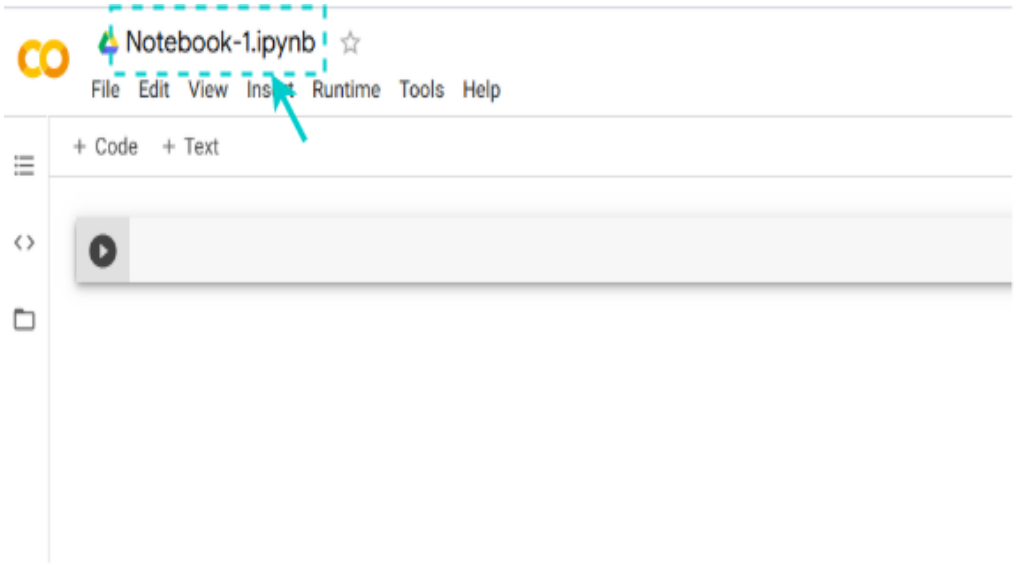

Fig 5: Google Colab Notebook

Colab notebooks allow us to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more. When we create our own Colab notebooks, they are stored in our Google Drive account. We can easily share our Colab notebooks with co-workers or friends, allowing them to comment on our notebooks or even edit them. To learn more, see Overview of Colab. To create a new Colab notebook we can use the File menu above, or use the following link: create a new Colab notebook. Colab notebooks are Jupyter notebooks that are hosted by Colab.

As a developer, we can perform the following using Google Colab;

- Write and execute code in Python
- Create/Upload/Share notebooks
- Import/Save notebooks from/to Google Drive
- Import/Publish notebooks from GitHub
- Import external datasets
- Integrate PyTorch, TensorFlow, Kera's, OpenCV.

RESULT AND PERFORMANCE USING GRAPH

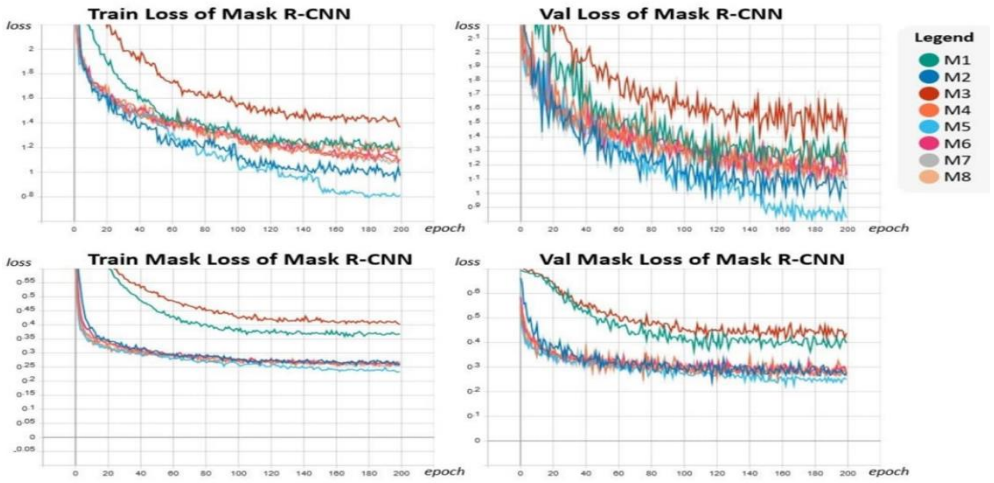

Fig 6: Performance Graph

CONCLUSION

The virtual cloth fitting in 2D, a system that relies on using images, has been successfully developed. The main aim of the system is achieved which is fitting a dress onto the individual and visualizing the results as a generated image. The development of this project is based on efficient deep learning neural networks CNN, U-net are the main networks used. The dress shaping module is successfully able to change the shape of the new dress by estimating the necessary parameters. It is able to not only change the shape but also in maintaining the details of the dress's logo, patterns, text associated on the dress without much loss. The fitting module successfully fits the new dress on the individual without overfitting and occlusion with other body parts. The generated images have near natural appearance. This system will be helpful to bring a more realistic shopping experience to the users and is computationally efficient, and effective results can be generated. Robust predictions can be made with this system, which is necessary for large-scale integration like in E-commerce. critical parameters of the model are the features extracted and proper estimation of the TPS transformation parameters. The activation function in CNN are the important parameters and they are ideally chosen such that they give optimal values of loss function, from various experiments tried on changing neural network architecture.

REFERENCES

[1] J. Johnson, A. Alahi, and F. Li, "Perceptual losses for real-time style transfer and super-resolution," CoRR, vol. abs/1603.08155, 2016.

[2] V. khandelwal, "Architecture and implemetnation of vgg-16," 2018. [Online]. Available: https://medium.com/towards-artificial-intelligence/ the-architecture-and-implementation-of-vgg-16-b050e5a5920b

[3] I. Rocco, R. Arandjelovic, and J. Sivic, "Convolutional neural network architecture for geometric matching," CoRR, vol. abs/1703.05593, 2017.

[4] P. Guan, L. Reiss, D. Hirshberg, A. Weiss, and M. J. Black, "DRAPE: DRessing Any PErson," ACM Trans. on Graphics (Proc. SIGGRAPH), vol. 31, no. 4, pp. 35:1–35:10, Jul. 2012.

[5] M. Sekine, K. Sugita, F. Perbet, B. Stenger, and M. Nishiyama, "Virtual fitting by single-shot body shape estimation," 10 2014, pp. 406–413.

[6] G. Pons-Moll, S. Pujades, S. Hu, and M. Black, "Clothcap: Seamless 4d clothing capture and retargeting," ACM Transactions on Graphics, vol. 36, pp. 1–15, 07 2017.

[7] X. Han, Z. Wu, Z. Wu, R. Yu, and L. S. Davis, "VITON: an image-based virtual try-on network," CoRR, vol. abs/1711.08447, 2017.

[8] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," vol. 9351, 10 2015, pp. 234–241.

[9] A. Hilsmann and P. Eisert, "Tracking and retexturing cloth for realtime virtual clothing applications," in In Computer Vision/Computer Graphics Collaboration Techniques (MIRAGE, 2009, p. 94.

[10] trimirror, 2018. [Online]. Available: https://www.trimirror.com/

[11] N. Jetchev and U. Bergmann, "The conditional analogy gan: Swapping fashion articles on people images," 10 2017, pp. 2287–2292

[12] K. Li, M. J. Chong, J. Liu, and D. Forsyth, "Toward accurate and realistic virtual try-on through shape matching and multiple warps," 2020.

[13] K. Gong, X. Liang, D. Zhang, X. Shen, and L. Lin, "Look into person: Self supervised structure-sensitive learning and a new benchmark for human parsing," 07 2017, pp. 6757–6765.

[14] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," 12 2018.

[15] R. Sprengel, K. Rohr, and H. S. Stiehl, "Thin-plate spline approximation for image registration," vol. 3, 1996, pp. 1190–1191 vol.3.

[16] H. Lombaert, 2006. [Online]. Available: https://profs.etsmtl.ca/hlombaert/ thinplates/

[17] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," International Conference on Learning Representations, 12 2014.

[18] F. D, 2017. [Online]. Available: https://towardsdatascience.com/ batch-normalization-in-neural-networks-1ac91516821c

[19] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in 2017 International Conference on Engineering and Technology (ICET), 2017.

[20] S. saha, 2018. [Online]. Available: https://towardsdatascience.com/ a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[21] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," 2015, pp. 730–734.

[22] [Online]. Available: https://www.zalando.co.uk/