

Received April 24, 2020, accepted May 6, 2020, date of publication May 11, 2020, date of current version May 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2993574

FashionFit: Analysis of Mapping 3D Pose and Neural Body Fit for Custom Virtual Try-On

MOHAMMAD FARUKH HASHMI¹, (Member, IEEE), B. KIRAN KUMAR ASHISH²,
AVINASH G. KESKAR³, NEERAJ DHANRAJ BOKDE⁴,
AND ZONG WOO GEEM⁵, (Member, IEEE)

¹Department of Electronics and Communication, National Institute of Technology, Warangal 506004, India

²Hexvel Disruptor, Hyderabad 500084, India

³Department of Electronics and Communication, Visvesvaraya National Institute of Technology, Nagpur 440010, India

⁴Department of Engineering-Renewable Energy and Thermodynamics, Aarhus University, 8000 Aarhus, Denmark

⁵Department of Energy IT, Gachon University, Seongnam 13120, South Korea

Corresponding author: Zong Woo Geem (geem@gachon.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) under Grant 2020R1A2C1A01011131, in part by the Energy Cloud Research and Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT under Grant 2019M3F2A1073164.

ABSTRACT Visual compatibility and virtual feel are critical metrics for fashion analysis yet are missing in existing fashion designs and platforms. An explicit model is much needed for implanting visual compatibility through fashion image inpainting and virtual try-on. With rapid advancements in the Computer Vision realm, the increase in creating customer experience which leads to the great potential of interest to retailers and customers. The public datasets available are very much fit for generating outfits from Generative Adversarial Networks (GANs) but the custom outfits of the users themselves lead to low accuracy levels. This work is the first step in analyzing and experimenting with the fit of custom outfits and visualizing it to the users on them which creates the great customer experience. The work analyses the need for providing visualization of custom outfits on users in the large corpora of AI in Fashion. The authors propose a novel architecture which facilitates the combining outfits provided by the retailers and visualize it on the users themselves using Neural Body Fit. This work creates a benchmark in disentangling the custom generation of cloth outfits using GANs and virtually trying it on the users to ensure a virtual-photorealistic appearance and results to create a great customer experience by using AI. Extensive experiments show the high accuracy levels on custom outfits generated by GANs but not in customized levels. This experiment creates new state-of-art results by plotting users pose for calculating the lengths of each body-part segment (hand, leg, and so forth), segmentation + NBF for accurate fitting of the cloth outfit. This paper is different from all other competitors in terms of approach for the virtual try-on for creating a new customer experience.

INDEX TERMS Neural body fit, generative adversarial networks (GANs), pose, customer experience, segmentation.

I. INTRODUCTION

Recent developments and breakthroughs in Computer Vision realm in Fashion space such as the implementation of Variational Autoencoders (VAEs) [1], Generative Adversarial Networks (GANs) [2], and its variants gave a path to a myriad to fashion synthesis using computer vision [3]–[5]. These applications include visual-try on, AR/VR neural body fit, language-guided fashion synthesis. With daily uploading of photos in social media [4], [5] throughout the world in every-day scenarios, the datasets [4]–[6] and the environment

scenes understanding around has become much easier for the researchers to develop the visual recommendation systems based on given inputs by the user.

Most of the existing methods rely on the GANs [2], [7] which gives high-quality, fine, and realistic images [4], [8] as shown in Figure 1. Although GANs [2], [9] can generate high perceptual features that give realistic-feel [5], [10], the retailers are finding a narrow gap in gaining the customer experience due to the unavailability of custom testing on them of cloth outfits. GANs [2] generated fashion images [7] are mostly employed in today's virtual try-ons and fashion industry. These are mainly used for advertising and display purpose. The realistic images are propose in [11] are limited

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamad Forouzanfar¹.



FIGURE 1. Samples cloth outfits generated by GANs [14].

to selecting available garments for just normal shopping purpose. Often film stars, sports personalities, in movies, and so forth appear on television with mesmerizing cloth outfits. Most of them wouldn't reveal the source or are custom designed. Noa Garcia and Georgia Vogiatzis [12] proposed "Dress like a Star" method where outfits can be extracted from the video frames. The latest trend lies in the image generation which are realistic in vision. Han et.al [13] proposed FiNet network which generates realistic images. These are most often used in fashion industry to create new designs. Honda Shion [14] proposed VITON-GAN for a virtual try-on with GANs.

The existing methods couldn't achieve high-accuracy levels on custom visual try-on [3], [7], [12], [15], [16], where the user can upload himself/herself and try-on different cloth-image outfits for getting the reality-feel on screen. Many public datasets [6], [17]–[21] are available but the lag in gaining the power of complete access on users' fingertips isn't available. Therefore, this work proposes a novel architecture where the user can try out different outfits on himself/herself with gives the utmost reality on visual screens. This breakthrough can open many doors for the customers in getting the actual try-on feel as they usually do it in the shopping malls and bridges the gap between reality and reality-feel through Computer Vision.

In recent times, there has been extensive research in characterizing clothing using computer vision algorithms. The high rate of people uploading photos with different styles in social media has been quite spiked in recent years. The cloth outfits dataset is tremendous and is quite useful for the generation of new cloth garments and the perfect fit of new outfits using GANs [2] has been simple. With the availability of a humongous dataset of wide categories of cloth outfits, this work mostly focusses on the algorithm building for getting a photorealistic of garment fit onto bodies, i.e., virtual try-on of customized cloth outfits. This stylistic signature of virtual try-on mainly deals with GANs [2] computing of combining

the human body with custom selected cloth garments [13]. The perfect body fit [22] had been a challenge till recent past due to unavailability of data which lags in creating a photorealistic image, now as DeepFashion2 [17] and many such datasets have shown up, its become easier for virtual try-on, as it is being training on very large scale dataset bunch [23].

This work involves the construction of an advance framework to aggregate all the learned features from the 3D-pose plots [24] and corresponding cloth attributes for matching the nodes for visual perception [9]. The learned features from the CNN network involve the clothes categories, 3D-poses [24], and masks to solve the clothing image retrieval in an end-to-end virtual try-on manner.

II. RELATED WORK

A. VISUAL CLOTH OUTFITS AND GEOGRAPHY

The clothing style differs in different geographical locations [21]. Numerous styles in outfits throw some sort of challenging task because the outfits are complex in some scenarios such as thick styles, loose styles for shirts and many more. These latest styles are recent trends and have set a benchmark in the fashion industry. The lack of availability of these kinds of complex outfit datasets drew a line in moving forward for all available scenarios and forced the researchers to fit their architecture with limited outfits.

B. IMAGE AND SEGMENT SYNTHESIS

Fashion is a huge ocean has an overflow of styles in it. The users are very smart in having to like different styles for each segment. For example, sleeveless, shorts, and many other styles [25]. Different segmented styles outfits throw a challenge accurately mapping it to the particular body segment. This problem arises when the shorts maps to hands which are false positive. In recent developments, the annotations

available and outfits generated [7], [9], [14] in fashion outfits wiped this challenge as shown in Figure 2.

C. VISUAL RECOMMENDATION SYSTEM

Visual Compatibility Modelling plays a key role in visual search, visual recommendation systems, and visual information retrieval systems [8], [14], [15], [26]–[29]. The recent trends in this space especially the image synthesis where similar cloth outfits appear on the screen. The simple NLP and CNN for classification groups similar objects into one query search. The latest outfits released often appear on the screen by users mapping onto a similar class/object. Furthermore, the existing models are all labeled and filtered by users to get different styles and outfits separately. The comparison is homogenous in the existing filter search. Different outfits and styles on a single catalog are done on different sorting and queries. This leads to losing potentials from exploring much deeper. Fashion experts recommend having all styles and outfits on a single plate to the users which are quick, smarter, and productive. This creates an imagination of users with a realistic feeling of comparison with different catalogs. The most recent usage of deep learning models is bidirectional LSTM and Siamese CNN [19], [30] for predicting the next item based on past and currently selected catalog by the user.

D. IMAGE SYNTHESIS

1) GENERATIVE ADVERSARIAL NETWORKS (GANs)

The latest advancements in creating customer experience are generating cloth outfits by GANs [2], [7], [13]. GANs [2] has opened a wide-angle in many domains, in fashion it proved to create a great path of creating a great customer experience. Retailers found this very potential way of attracting funnel of creating a great customer experience. Shion Honda [14] proposed a two-stage architecture of generating new clothes onto a person and transfer it to a different person. This new clothing visualization created many interests for the users trying out with different arbitrary poses [31], [32] and GANs generated outfits.

2) CONDITIONAL GANs

Conditional GANs has generative models of data on discrete labels and images, creates a new way of mapping scenes, edges for a bidirectional mapping between unpaired images [14]. The existing model consists of a semantic layout of images [33] generated from different inputs from the users.

E. HUMAN POSE + SEGMENTATION FOR GANs GENERATED OUTFITS

Another approach or getting accuracy in custom try-on outfits is the human parsing and pose estimation [34], [35] which contains pixel-level annotation of the body [4]. The work proposed by [14] covers the region of interest on the body

with specified key points where the cloth outfits are to be appended onto [11]. This approach made a wider path for GANs [7] to validate its generated outfits on the body which gave an essence of virtual try-on [14], [31], [32], [36].

F. DATASETS

This works leverages several datasets such as fashion datasets [25], DeepFashion2 [17] for custom cloth outfits. MVC dataset [19] provides invariant clothing retrieval and attribute prediction. It even provides cropped and dressed person images with varies views, multiple poses. MPII Pose dataset [35], DensePose [9] are used for human parsing and pose estimation. DeepFashion2 [17] contains a 391K training set with 13 classes, a validation set of 34K images, and 67K images of the testing set. LIP dataset [37] and few random images taken from google were tested for prediction. The annotations were largely done on the segmentation basis using the VIA annotation tool [38]. This paper experiments the combination and concatenation of both the architectures of Fashion and Pose for setting up a new benchmark in advancing the virtual try-on to create a new customer experience.

III. TECHNICAL APPROACH

This paper gives an essence of combining two-great esteems, i.e., Human Pose and Fashion's Virtual try-on. This experiment an initial step in the FashionAI by creating a fully custom virtual try-on at the fingertips of the users provided with the highest accuracies. The problem raised in the existing models is the absence of fully custom features like choosing custom outfits separately and virtually trying it on their catalogs. This created an inadequate customer experience in the fashion market. This paper summarizes a novel approach in which provides fully custom features with custom virtual try-on by combining two architectures.

A. VISUAL RECOMMENDATION AND QUERY

This step is crucial for selecting custom garments by the user. The similarity search based on Euclidean distance from selecting the outfits by the user gets the recommendation of similar garments available with the e-retailers.

The recommendation system queries [36], [39], [40] based on the past and current selection [3]. The image retrieval [3], [41]–[43] for the users is as follows. Let the E-retailers cloth outfit gallery be given by a set $G = y$, it computes the similarities s between the image query x and object y and ranks them $x = x^i$ and $y = y^j$, where $x^i \in R^{C \times I}$ and $y^j \in R^{C \times I}$ are locally mapped features of cloth image retrievals and the e-retailer respectively [21], [44], [45]. Mapping retailers and cloth images retrievals as global variables:

$$s_g = S_g(A(x), A(y)) \quad (1)$$

where, $A(\cdot)$ is the aggregate function consisting of a pooling operator (either average or maximum) in the convolutional layers and $S_g(\cdot, \cdot)$ is the similarity function. Most of the



FIGURE 2. A Snippet of DeepFashion 2 dataset.

times the similarity function is the Euclidean distance and sometimes cosine similarity can be used based on the datasets available. Most of the visual recommendation systems [36], [39], [46], [47] are built upon Euclidean distance operators. This function maps the input query to the clothing set available in the retailer's database.

Noise features were observed due to similar matching of pixels in backgrounds, scenes, other objects in the background which can be avoided only by cropping the target object is a challenging task in moving forward. To suppress this, [30] proposed a similarity correlation between the input query and the local pixels array as follows:

$$s_l = \sum_{i,j} w_l^{ij} S_l(x^i, y^j) \quad (2)$$

where, $S_l(.,.)$ is the scalar local similarity function, and w_l^{ij} is the scalar weight of local similarities $S_l(x^i, y^j)$ which is given by:

$$w_l^{ij} = \begin{cases} 1, & \text{if } j = \operatorname{argmax}_k (S_l(x^i, y^k)) \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

This mitigates the problem of correctly mapping the cloth outfits accurately to the body part segment. After the user

selecting the cloth outfit, he/she would then upload their full-length photo for a virtual-try on. The visual recommendation used here is to display the model works on garment classes that are trained on the DeepFashion2 dataset. Other than those classes, virtual try-on would not be possible for a complete virtual try-on since the results would go wrong.

B. HUMAN POSE ESTIMATION

The persons are detected by a person detection algorithm that used Region Proposed Network (RPN). No new data of a person is annotated nor trained, a pre-trained model of person detection from COCO dataset [48] is used to detect the person in a given frame. RPN works with performing a Boolean operation in the whole frame, by comparing pixel-to-pixel with respect to pixels trained in the person detection. The results return 'True' if the pixels correlate with each other, else 'False' if they are far from trained values.

1) PIPELINE

The Pose architecture [35] is designed with Symmetric STN, which consists of Spatial Transformer Network (STN) and Spatial De-Transformer network (SDTN) which are attached before and after the Single Person Pose Estimator (SPPE). The STN receives human posture as input from the

user and the SDTN generates pose proposals. The Parallel SPPE acts as an extra regularizer during the training phase. Finally, the Parametric Pose Non-Maximum Suppression Algorithm (p-Pose NMS) is computed to eliminate redundant pose estimations [22], [34], [35].

2) SYMMETRIC STN AND PARALLEL SPPE

A Symmetric STN + parallel SPPE was introduced to enhance SPPE when given imperfect human proposals [24], [34]. Generally, a perfect still posture is recommended to be given as input from the users. But this works experimented with the possible worst-case scenarios such as stylish poses, imperfect poses as input which can be mitigated by using the Spatial De-Transformer network (SDTN) and Spatial Transformer Network (STN) which selects a region of interests automatically.

The STN extracts high quality dominant human proposals. Mathematically, the STN performs as 2D affine transformation is expressed as:

$$\begin{pmatrix} x_i^s \\ y_i^s \\ 1 \end{pmatrix} = [\theta_1 \theta_2 \theta_3] \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix} \quad (4)$$

where, θ_1 , θ_2 and θ_3 are vectors in R^2 . x_i^s, y_i^s and x_i^t, y_i^t are the coordinates before and after transformation, respectively. After SPPE, the resulting pose is mapped into the original human proposal image. A Spatial De-Transformer network (SDTN) is required to remap the estimated human pose back to the original image coordinate [34], [35]. The SDTN computes the γ for de-transformation and generates grids based on γ :

$$\begin{pmatrix} x_i^s \\ y_i^s \\ 1 \end{pmatrix} = [\gamma_1 \gamma_2 \gamma_3] \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix} \quad (5)$$

where, γ functions are given as:

$$[\gamma_1 \gamma_2] = [\theta_1 \theta_2]^{-1} \quad (6)$$

$$\gamma_3 = -1 \times [\gamma_1 \gamma_2] \cdot \theta_3 \quad (7)$$

3) PARALLEL SPPE

To further help STN to extract good human dominant regions, a parallel SPPE branch was added in the training phase [34]. This branch shares the same STN with the original SPPE, but the spatial de-transformer (SDTN) is omitted. The human pose label of this branch is specified to be centered. All the layers of this parallel SPPE were frozen during the training phase. The weights of this branch are fixed, and its purpose is to back-propagate center-located pose errors to the STN module.

4) POSE NMS

Human detectors inevitably generate redundant detections, which in turn produce redundant pose estimations. Therefore, pose non-maximum suppression (NMS) eliminates the redundancies [34] to avoid false positives.

5) POSE DISTANCE

To plot the pose accurately from the center coordinates, a distance function is used given by $d_{pose}(P_i, P_j)$. Assuming the box for P is B_i , a soft matching function [24], [34], [35] is used given by:

$$K_{sim}(P_i, P_j | \sigma_1) = \begin{cases} \sum_n \tanh \frac{c_i^n}{\sigma_1} \cdot \tanh \frac{c_j^n}{\sigma_1}, & \text{if } k_j^n \text{ within } B(k_i^n) \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

where, $B(k_i^n)$ is a box center at k_i^n , and each dimension of $B(k_i^n)$ is 1/10 of the original box B_i . The \tanh operation filters out poses with low-confidence scores. When two corresponding joints both have high confidence scores, the output will be close to 1. This distance softly counts the number of joints matching between poses.

The spatial distance [34] between parts is also considered, which can be written as:

$$H_{sim}(P_i, P_j | \sigma_2) = \sum_n \exp \left[\frac{(k_i^n - k_j^n)^2}{\sigma_2} \right] \quad (9)$$

By combining both the equations, the final distance function can be written as:

$$d(P_i, P_j | \Lambda) = K_{sim}(P_i, P_j | \sigma_1) + \gamma \cdot H_{sim}(P_i, P_j | \sigma_2) \quad (10)$$

where, γ is a weight balancing the two distances, and $\Lambda = \{\sigma_1, \sigma_2, \gamma\}$.

C. HUMAN SEGMENTATION

Although pose would give enough accuracy while using Neural Body Fit [22], there are some false positives and false negatives while relying completely on the pose. Often due to local similarities in the local pixel-level grading, the pose plotting becomes inaccurate where the segmentation balances the architecture. To mitigate this, segmentation is used to assist the pipeline to accurately fit the cloth outfits on the body without any overlaps and mixing on existing outfits. Segmentation draws a border between each body segment which becomes a much easier task for mapping global similarities of clothes and the body.

The noise redundancies raised due to the local similarities are reduced by segmentation. the local pixel-to-pixel similarities make the pose plot incomplete and improper places deviating from the subject. The annotations of the segmentation are quite heavier than the pose and hence at the architecture level, deeper is the extraction of the features.

To make the model lightweight, the architecture is a combination of pose and segmentation. This assures the accuracy levels and reduces the usage of computational resources. Segmentation is first performed to reduce and eliminate the local noise. The pose is plotted based on the segmentation signals. The recommendation and coordinates given by both are then utilized by the Neural Body Fit [22] to map the outfits onto the body.

Base Algorithm:

```

1 Step 1: Detect Person using RPN
2 if Person == True:
3     extract( $x, y, w, h$ )
4     append(coordinate vector)
5 else
6     return Person
7
8 Step 2: Plot Pose plots around the body
9  $Q \leftarrow$  global pose array
10 Plot ( $x, y$ ) on each body part and plot pose plots, append then into  $Q$ 
11 Predict 3D points using Q-points; Q-2D points
12 NBF formula to plot ( $x + y, w + h$ )
13
14 Step 3: Cloth outfits selection and NBF integration for warping
15  $C \leftarrow$  query search word
16 for each in  $Q$ :
17     if  $C = \text{Person}(x, y)$ 
18         check the size with cloth garment
19     concatenate  $C(i) + Q = \text{master}$ 
20  $\text{master} * \text{NBF}$  {concatenate cloth garment size onto pose plotted and multiply it with person's ( $x, y$ )}
```

FIGURE 3. Pseudocode for the Base algorithm.**D. PSEUDO ALGORITHM**

The pseudo code for the Base algorithm is shown in Figure 3.

IV. EXPERIMENTS**A. NETWORK CONSTRUCTION****1) HUMAN SEGMENTATION**

The first step would be the segmentation of humans. This would avoid the loss caused by the local similarities. Fully convolutional network architecture is used to obtain the semantic segmentation. The forward inference contracts for pixel-wise prediction. Starting with a 64-channel dimension with 2-convolution filters, adding on to 128-channel dimensions with 2-blocks and 2-convolutions. The outputs are fed as input to a 256-channel dimension with 3-convolution filters and then the model is added to two layers of 512-channel dimensions, one with 4-blocks and other with 5-blocks each with a 3×3 kernel filter size. The last two layers consist of a 4096-channel dimension, with 7×7 and 3×3 kernel filter sizes respectively. Each convolutional layer is fed with the Leaky ReLu activation function and the last layers are fed with ReLu activation filters. This is the De-Convolutional/Down Sampling funnel. This final layer is given as input to the Up-Sampling funnel where the decoding of the convolutional layers' computes. The De-Convolutional layer starts with computing the last layer of the 4096-channel dimension multiplying with the output size filter. After the sequential block, a 2D convolutional layer is initiated with a kernel size of 1×1 . This is summed up with the (preceding layer multiplied with the output size) with having a 4 Max-Pooling functions for each layer. Then, a 2D De-Convolutional layer with a kernel size of 32×32 with

21-channel dimensions and padding size 1. The total trainable parameters [49] at the end were ~ 13 million.

After the human is detected, the Boolean operation of setting up the segmented masks are computed pixel-wise on the object. The background is subtracted from the target object mitigating the local similarities in the image. The background subtraction shown in Figure 4 is computed based on the pixel-level grading and pixel-level intensity subtraction. Manual testing is done using the GrabCut method [27], where manual intervention is needed to select the target object.

This paper automated the automation of selecting the target object based on the intensity levels. The detected persons are sent for pixel-wise comparison and intensity calculation. The nearer object consists of heavier intensity than the rest. Then, the feed is fed to the subtraction operator where the low-pixel intensities are isolated from the high-pixel intensity object. This maps the target object isolated from the rest.

This would be the first step in the proposed pipeline, segmenting the human and applying mask on each segment. The annotations were done by VGG Image Annotator (VIA) tool, using the polygon geometry tool. The annotation was saved in .json format which was given for training. The dataset used is DeepFashion2 for training. This is fed to the next step pose estimation for each segment measurement.

2) HUMAN POSE ESTIMATION

The next step is the pose estimation of the target object. To maintain the accuracy levels in making the cloth outfits fit the human body, the pose is added onto the final layer of the segmentation pipeline. Segmentation may mix up the local similarities which degrade the Neural Body Fit [22]

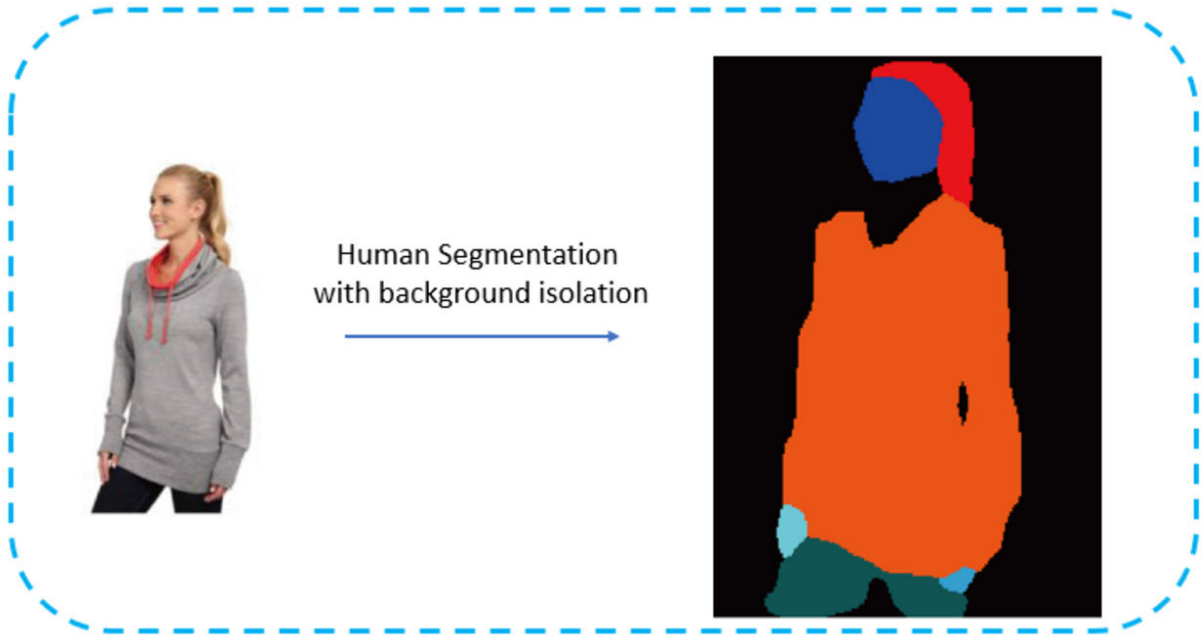


FIGURE 4. Human Segmentation alongside with background isolation. Each part is masked with different masks. Input image taken from LIP dataset.

in virtual try-on. To mitigate this, pose based on pixel-level grading is concatenated. The architecture starts from the output feed from the segmentation. Once the target object is identified, the coordinates are fed as inputs to the pose block where the pose lines are plotted. Reading these coordinates are input, the pose plotting architectures map the center-coordinates and plot the pose within the region of interest. Single Person Pose Estimator (SPPE) is used in this region of interest to estimate the human pose skeleton. This is based on the center-coordinates, where the joints are mapped from the center and a Spatial De-Transformer Network (SDTN) is used to remap the estimated human pose back to the original image coordinate system. Finally, a parametric pose Non-Maximum Suppression (NMS) is used for redundant pose deductions. 3D plots are visualized after applying affine 2D transform and then converting it into 3D plots by computing $([x_-, y_-, z_-]F[x, y, z]T = 0)$ matrix as mentioned in section 3.2.

This step utilizes the exact posture and size of the user which is used to select the cloth outfits of the measured size and for the Neural Body Fit algorithm to warp garments precisely. Post human pose extraction, based on the measurements and initial garment selection, similarity search would shoot recommendations of available target garment outfits which is the next step in the pipeline.

3) SIMILARITY QUERY REASONING

The query raised by the user by selecting the custom cloth outfits maps to the global similarity in recommending the objects. The similarity s_l^{ij} raised by conducting a sequence of similarity propagation, linear transformation, and non-linear

activation operator is mathematically propagated as:

$$s_{l_1}^{ij} = \sum_{l_2, m, n} w_P^{l_{1ij}l_{2mn}} \cdot S_{l_2}^{mn} = \sum_{l_2, m, n} w_P^{l_{1ij}l_{2mn}} \cdot S_{l_2}^{mn} \cdot S_P(X_{l_2}^m, y_{l_2}^n) \quad (11)$$

The linear transformation and non-linear activation function are defined by:

$$h_{l_1}^{ij} = \text{ReLu}(Ws_{l_1}^{ij}) \quad (12)$$

where, $W \in R^{C \times D}$ are the learnable parameters.

This process pulls the recommendation of similar garments available with e-retailers. The visual search recommendation increases the business revenue of the retailers as well as provides a great customer experience by displaying an available wide range of outfits. The next step would be the virtual try-on in the pipeline.

4) NEURAL BODY FIT

Neural Body Fit (NBF) [22] predicts the body model and parts from the colored-segment mask map $I \in R^{224 \times 224 \times 3}$ using CNN-estimator having Up-Sampler and Down-Sampler parameterized by weight w . The pose-estimators and shape are defined by $\theta(w, I)$ and $\beta(w, I)$ with respect to independent variables weights and segment maps [22].

Mathematically, the function $N_{3D}(w, I)$ that maps from semantic images to 3D-meshes are given by:

$$N_{3D}(w, I) = \begin{cases} M(\theta(w, I), \beta(w, I)) \\ W(T(\theta(w, I), \beta(w, I))) \\ J(\beta(w, I), \theta(w, I), W) \end{cases} \quad (13)$$

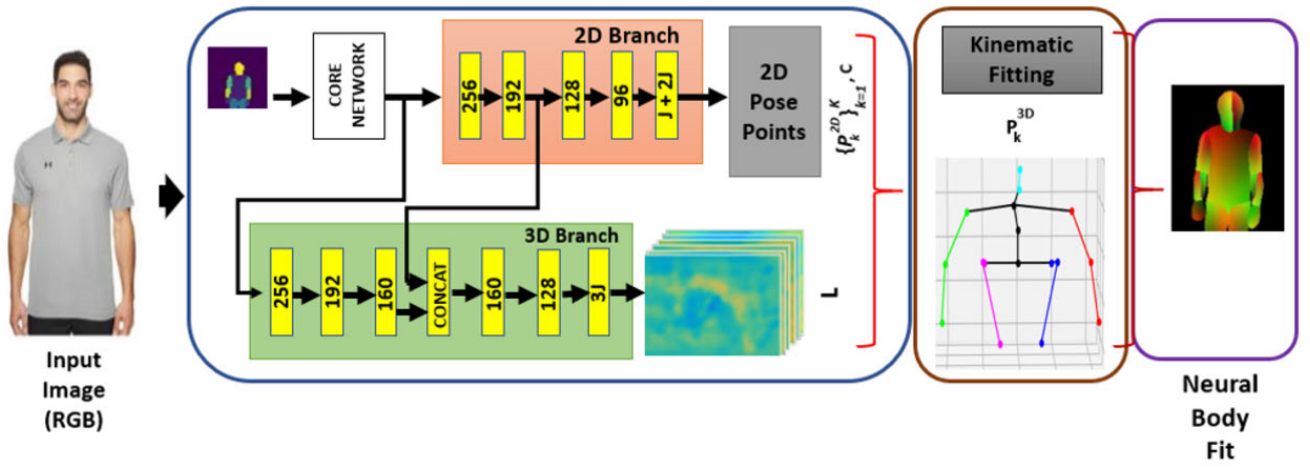


FIGURE 5. Architecture of mapping Neural Body Fit from 3D points. Input Image taken from LIP dataset.

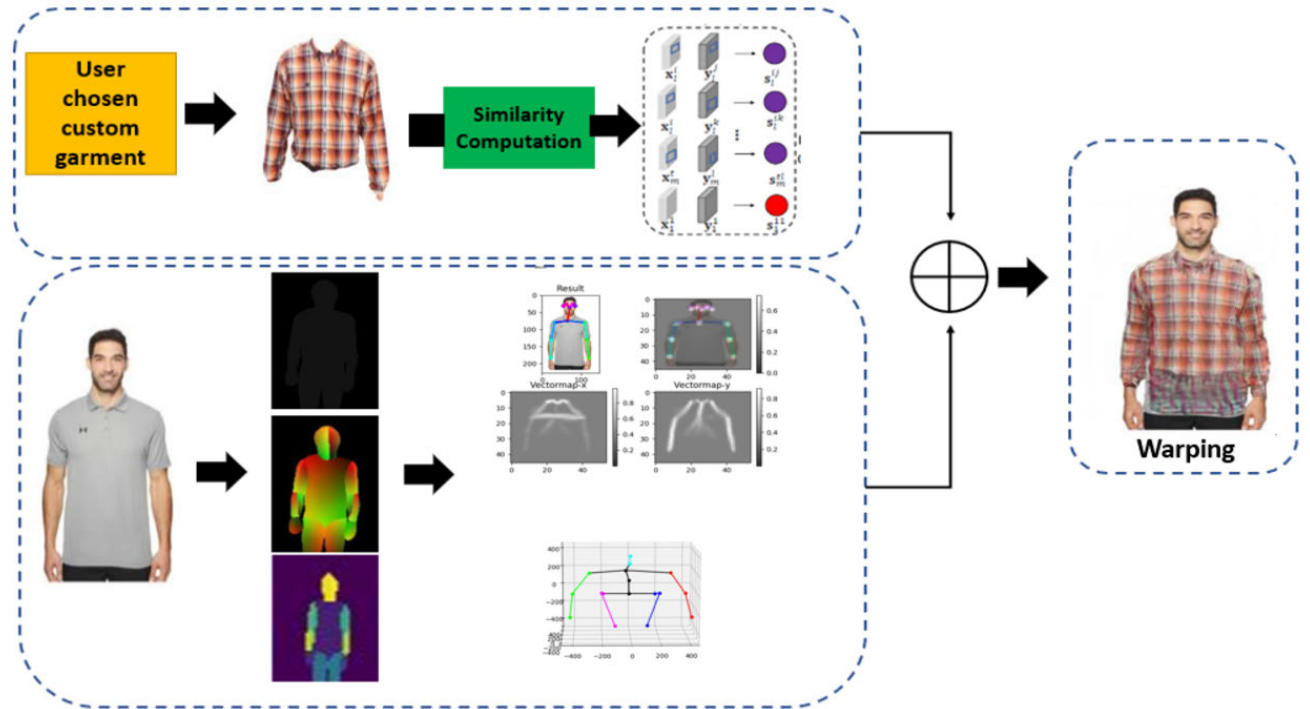


FIGURE 6. Network of pointing points for virtual try-on the custom garment from the search query. Testing done on LIP dataset.

These are parameterized by network parameters w . NBF estimates the 3D maps and 3D joints onto the coordinates of the 2D points, $N_J(w, I) = J(\beta(w, I))$. Using a projection operation $\pi(\cdot)$ 3D joints are projected onto a single image plane:

$$N_{2D}(w, I) = \pi(J(w, I)) \quad (14)$$

where, $N_{2D}(w, I)$ is the NBF function that maps 2D joint coordinates. These operations are differentiable and gradient-based optimization is used to update the model with an

average-mean loss function. Figure 5 describes the working architecture of NBF.

This work provides an end-to-end virtual try-on based on above technical approach steps. The first step includes sorting out persons from the frames, i.e., segmentation by isolating it foreground with background. Post isolating the subject from the background for precise warping of garments, pose is plotted for accurate garment warped onto the subject based on the pose measurements. Now, after selecting the target garment, based on the selected garment, similarity search



FIGURE 7. The warping process of custom garments onto user's garment.

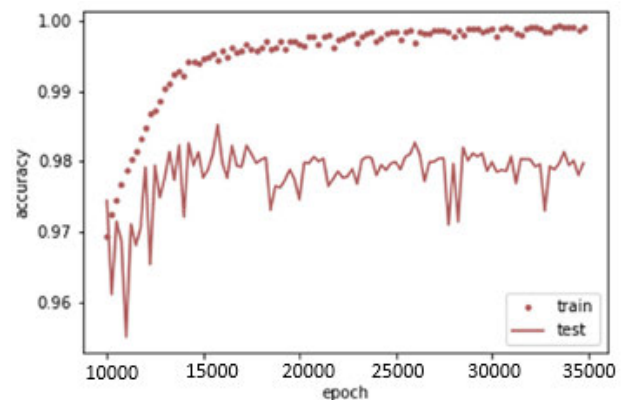
query recommends a similar type of garment as a recommendation feature. Selected garments/outfits are warped on the subject using the Neural Body Fit algorithm. The warping of garments onto the subject by matching the joints and points.

The results of the trained model on the DeepFashion2 dataset identifying the cloth garments on the body. This is useful since most of the cloth outfits in the retailer platform would be dressed in some fashion models. Hence, isolating the garments selected by the user is crucial to perform a proper warping. If the cloth outfits are individually displayed as a raw piece, this step can be skipped and directly perform virtual warping on the user.

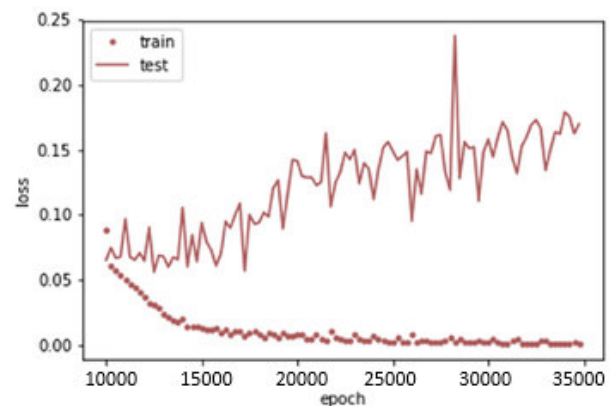
The backbone of warping, i.e., neural body fit is then computed post selecting the garments and uploading the user. The NBF computation mostly includes the extraction of points from the human. The last step would be the warping process of garments on the body based on the points extracted. These two steps are crucial in virtual try-on with NBF acting as a backbone to the architecture. Figure 6 describes the warping of the garment chosen by the user on the user. The similarity search shown in Figure 6 is the visual recommendation of similar garments that can be tried out by the user. Figure 7 gives a glimpse of garments warping outputs in the mid-way computation. It clearly states the first step of isolating the garment. From Figure 6, the then step would be NBF of extracting points from the user. The garment then sets to warp on the user. This is the over the pipeline of a virtual try-on.

B. PERFORMANCE METRICS

Figure 8 resumes the performance of the network trained on the DeepFashion2 dataset after over 35K epochs. With over



(a) Training and Validation accuracy.



(b) Training and Validation Loss.

FIGURE 8. Training, Validation accuracy and loss.

341K of the training set and 67K of testing set, the overall training patch accuracy was noted 99.78% and testing patch at 97.73%. With building the network with Adam optimizer,



FIGURE 9. Results of Object Segmentation of cloth outfits of trained model on DeepFashion2 dataset. The input image is a randomly picked test sample from a fashion retail 'LC WAIKIKI'. (Source: <https://www.lcw.com/en-IE/p/trousers-9WM055Z8-FKR>).

both training and validating showed decreasing behavior. The network took 1024×1024 sized image patches to train the model. At the RPN stage, the bounding box scales were defined as 0.25, 0.5, 1.0. Then 2.0 with 0.5, 1.0, 2.0 aspect ratios, which the network generated 12 anchor boxes for each location of the feature maps. The inner lining of segmentation points would be throughout the edges of the object. For a better picture, Garment Prediction shown in Figure 9 is the best example. The Batch Size was defined as 1 to prevent memory allocation errors, although this consumes a lot of time for training, the authors chose this way to prevent memory constraint errors. The training process continued till 35K epochs which took 72 hours. The learning rate started at 0.003 at the beginning and reduced to half of it in each further 15K step. The training loss did not decrease more post 30K steps.

C. EVALUATION METRICS

In the object detection tasks (technically segmentation tasks), the two widely used performance metrics are average precision and F1 score. The detector compares the predicting boundary boxes and segment points according to the intersection over union (IOU) at each epoch to update its parameters.

The F1 score evaluation metric is used to calculate the success rate by computing precision and recall rate respectively. Precision is the ratio of the actual matches of all objects that are detected as matches and recall is defines the ratio of the number of objects that are detected correctly to the number of all ground truths. Neither of them is individually sufficient to measure the performance of the network. Therefore, the F1 score, i.e., harmonic mean of them in mathematical terms is too calculated. By defining true positives (TP) as truly detected objects, false negatives (FN)

as non-detected objects, false positives (FP) as falsely detected objects, the above terms precision, recall and F1 is mathematically represented as:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (15)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (16)$$

$$F1 \text{ score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (17)$$

Overall crucial evaluation metrics have been plotted on the custom selected outfits as shown in Table 1. More and more dataset is needed to improve the F1 score on all garment. The scores are plotted on selected garments in Table 1 which are considered as most purchased from the retailer stores.

TABLE 1. Precision, recall, and F1 score on custom-selected outfits on custom-chosen/uploaded humans.

Type of Garment	Precision	Recall	F1 Score
Top	0.89	0.73	0.80
Shirts	0.97	0.87	0.91
Pants	0.97	0.92	0.95
Skirts	0.86	0.68	0.76
Dresses	0.87	0.65	0.74

D. RESULTS ON LIP DATASET (LOOK INTO PERSON) [37]

After passing the input human body to the network, the user raises a query for visualizing custom-selected outfits on the human body. Using NBF architecture, the custom-selected outfits are parsed onto the human.

This paperwork is different from other methods since this experiment mainly deals with custom-selected outfits onto custom-chosen humans (mostly users themselves).

TABLE 2. Accuracy results on custom-selected outfits on custom-chosen/uploaded humans.

Type of Garment	Accuracy (%)
Top	81.6
Shirts	83.4
Pants	91.06
Skirts	71.56
Dresses	70.5
Overall Accuracy	79.624

Table 2 describes the accuracy comparison of 5-cloth garments onto the proposed architecture. Since, this is the first fully custom-selection and custom-chosen humans for the virtual try-on method, the comparison with other state-of-art methods is not performed. The sole intention was to fully achieve high accuracy levels with all kinds of garments available. However, the architecture was also validated with GANs generated outfits.

TABLE 3. Accuracy table of virtual try-on of GANs generated outfits.

Type of Garment	Accuracy (%)
Top	53.5
Shirts	62.7
Pants	58.9
Overall Accuracy	58.36

An interesting result raised as shown in Table 3 above when the architecture is validated with GANs generated outfits. Since the model was not trained with any of the GANs data, the accuracy was quite low when compared to the real-clicked data. Due to the variations of pixel values, the precise fitting of outfits on the human body was not good enough to go further.

V. CONCLUSIONS AND FUTURE SCOPE

This work defines the complete automation of virtual try-on on custom chosen objects and humans. The accuracy levels of DeepFashion [17] trained models were almost ~80 % on the LIP dataset. The idea and approach are novel in implemented with a series of the pipeline. The other state-of-art methods are completely different approaches in terms of approach. This paper sets a new benchmark for a complete custom virtual try-on of garments on custom loaded human bodies. Therefore, the further scope of extending this work is phenomenal in achieving the highest accuracies on all possible datasets in noisy conditions and all possible cloth outfits. This experiment is the initiation of fully automated custom virtual try-on, there is more to go, and this paper initiated the first step in achieving it. However, the limitations of this work are heavyweight model which is computationally more complex. The model is limited to classes present in DeepFashion2 dataset. The future scope would be availability of large datasets, as already mentioned in section II-A,

the geography is large and diverse, continental dataset can emerge and altogether can train the model to make it into universal garment virtual try-on. This would be very useful for any retailers especially in the tourist's spots. Apart from fashion realm, implementation in entertainment and gaming would boost the gaming industry. The algorithm can be impeded with GANs and GANs generated outfits can be tried onto the custom chosen persons and vice-versa for marketing and advertisements.

REFERENCES

- [1] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*. [Online]. Available: <http://arxiv.org/abs/1312.6114>
- [2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [3] W. Di, C. Wah, A. Bhardwaj, R. Piramuthu, and N. Sundaresan, "Style finder: Fine-grained clothing style detection and retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 8–13.
- [4] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
- [5] J. Huang, R. Feris, Q. Chen, and S. Yan, "Cross-domain image retrieval with a dual attribute-aware ranking network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1062–1070.
- [6] *Fashion Ai Dataset*. Accessed: Dec. 31, 2019. [Online]. Available: <http://fashionai.alibaba.com/datasets>
- [7] K. Ak, A. Kassim, J.-H. Lim, and J. Y. Tham, "Attribute manipulation generative adversarial networks for fashion images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10541–10550.
- [8] Z.-Q. Cheng, X. Wu, Y. Liu, and X.-S. Hua, "Video2Shop: Exact matching clothes in videos to online shopping images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4048–4056.
- [9] R. A. Guler, N. Neverova, and I. Kokkinos, "DensePose: Dense human pose estimation in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7297–7306.
- [10] Z. Wang, Y. Gu, Y. Zhang, J. Zhou, and X. Gu, "Clothing retrieval with visual attention model," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.
- [11] H. Chen, A. Gallagher, and B. Girod, "Describing clothing by semantic attributes," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 609–623.
- [12] N. Garcia and G. Vogiatzis, "Dress like a star: Retrieving fashion products from videos," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2293–2299.
- [13] X. Han, Z. Wu, W. Huang, M. Scott, and L. Davis, "FiNet: Compatible and diverse fashion image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4481–4491.
- [14] S. Honda, "VITON-GAN: Virtual try-on image generator trained with adversarial loss," in *Proc. 40th Annu. Conf. Eur. Assoc. Comput. Graphics (Eurographics)*, Genoa, Italy, May 2019, pp. 1–2.
- [15] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 241–257.
- [16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [17] Y. Ge, R. Zhang, X. Wang, X. Tang, and P. Luo, "DeepFashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5337–5345.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
- [19] K.-H. Liu, T.-Y. Chen, and C.-S. Chen, "MVC: A dataset for view-invariant clothing retrieval and attribute prediction," in *Proc. ACM Int. Conf. Multimedia Retr. (ICMR)*, 2016, pp. 313–316.

- [20] W.-L. Hsiao, I. Katsman, C.-Y. Wu, D. Parikh, and K. Grauman, "Fashion++: Minimal edits for outfit improvement," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5047–5056.
- [21] W. H. Lin, K.-T. Chen, H. Y. Chiang, and W. Hsu, "Netizen-style commenting on fashion photos: Dataset and diversity measures," in *Proc. Companion The Web Conf. Web Conf. (WWW)*, 2018, pp. 395–402.
- [22] M. Omran, C. Lassner, G. Pons-Moll, P. Gehler, and B. Schiele, "Neural body fitting: Unifying deep learning and model based human pose and shape estimation," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2018, pp. 484–494.
- [23] N. Inoue, E. Simo-Serra, T. Yamasaki, and H. Ishikawa, "Multi-label fashion image classification with minimal human supervision," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2261–2267.
- [24] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis, "Learning to estimate 3D human pose and shape from a single color image," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 459–468.
- [25] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "DeepFashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1096–1104.
- [26] C. Corbiere, H. Ben-Younes, A. Rame, and C. Ollion, "Leveraging weakly annotated data for fashion image retrieval and label prediction," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2268–2274.
- [27] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut" interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.
- [28] Y. Shen, H. Li, S. Yi, D. Chen, and X. Wang, "Person re-identification with deep similarity-guided graph neural network," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 486–504.
- [29] G. Tzias, R. Sircu, and H. Jégou, "Particular object retrieval with integral max-pooling of CNN activations," 2015, *arXiv:1511.05879*. [Online]. Available: <http://arxiv.org/abs/1511.05879>
- [30] Z. Kuang, Y. Gao, G. Li, P. Luo, Y. Chen, L. Lin, and W. Zhang, "Fashion retrieval via graph reasoning networks on a similarity pyramid," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3066–3075.
- [31] C.-W. Hsieh, C.-Y. Chen, C.-L. Chou, H.-H. Shuai, and W.-H. Cheng, "Fit-me: Image-based virtual try-on with arbitrary poses," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 4694–4698.
- [32] M. Yuan, I. R. Khan, F. Farbiz, S. Yao, A. Niswar, and M.-H. Foo, "A mixed reality virtual clothes try-on system," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1958–1968, Dec. 2013.
- [33] J. Fu, J. Wang, Z. Li, M. Xu, and H. Lu, "Efficient clothing retrieval with semantic-preserving visual phrases," in *Proc. Asian Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 420–431.
- [34] R. Girdhar, G. Gkioxari, L. Torresani, M. Paluri, and D. Tran, "Detect-and-track: Efficient pose estimation in videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 350–359.
- [35] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3686–3693.
- [36] A. Sanchez-Ferrer, H. Perez-Mendoza, and P. Shiguihara-Juarez, "Data visualization in dashboards through virtual try-on technology in fashion industry," in *Proc. IEEE Colombian Conf. Appl. Comput. Intell. (ColCACI)*, Jun. 2019, pp. 1–6.
- [37] X. Liang, K. Gong, X. Shen, and L. Lin, "Look into person: Joint body parsing & pose estimation network and a new benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 871–885, Apr. 2019.
- [38] A. Dutta and A. Zisserman, "The VIA annotation software for images, audio and video," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2276–2279.
- [39] M. H. Kiapour, X. Han, S. Lazebnik, A. C. Berg, and T. L. Berg, "Where to buy it: Matching street clothing photos in online shops," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3343–3351.
- [40] X. Wang and T. Zhang, "Clothes search in consumer photos via color matching and attribute learning," in *Proc. 19th ACM Int. Conf. Multimedia (MM)*, 2011, pp. 1353–1356.
- [41] X. Guo, H. Wu, Y. Cheng, S. Rennie, G. Tesauero, and R. Feris, "Dialog-based interactive image retrieval," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 678–688.
- [42] X. Ji, W. Wang, M. Zhang, and Y. Yang, "Cross-domain image retrieval with attention modeling," in *Proc. ACM Multimedia Conf. (MM)*, 2017, pp. 1654–1662.
- [43] B. Zhao, J. Feng, X. Wu, and S. Yan, "Memory-augmented attribute manipulation networks for interactive fashion search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1520–1528.
- [44] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*. [Online]. Available: <http://arxiv.org/abs/1609.02907>
- [45] Y. Li, D. Tarlow, M. Brockschmidt, and R. Zemel, "Gated graph sequence neural networks," 2015, *arXiv:1511.05493*. [Online]. Available: <http://arxiv.org/abs/1511.05493>
- [46] I. Tautkute, T. Trzcinski, A. P. Skorupa, K. Lukasz, and K. Marasek, "DeepStyle: Multimodal search engine for fashion and interior design," *IEEE Access*, vol. 7, pp. 84613–84628, 2019.
- [47] C. Yan, Y. Chen, and L. Zhou, "Differentiated fashion recommendation using knowledge graph and data augmentation," *IEEE Access*, vol. 7, pp. 102239–102248, 2019.
- [48] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
- [49] W. Kim, B. Goyal, K. Chawla, J. Lee, and K. Kwon, "Attention-based ensemble for deep metric learning," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 736–751.



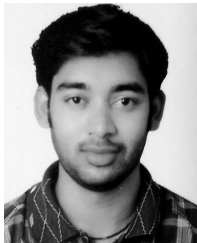
MOHAMMAD FARUKH HASHMI (Member, IEEE) received the B.E. degree in electronics and communication engineering from R.G.P.V Bhopal University, the M.E. degree in digital techniques and instrumentation from SGSITS Indore/R.G.P.V Bhopal University, in 2010, and the Ph.D. degree from VNIT Nagpur under the supervision of Dr. Avinash G. Keskar. He has a teaching and research experience of ten years. He is currently an Assistant Professor with the Department of Electronics and Communication Engineering, NIT Warangal. He has published up to 60 articles in International/National Journals/Conferences. His current research interests are computer vision, machine vision, machine learning, embedded systems, digital signal processing, image processing, and digital IC design etc. Dr. Hashmi is a member of LMIETE, ISTE, and LMIAENG.



B. KIRAN KUMAR ASHISH received the B.Tech. degree in electronics and communication engineering from the Anurag Group of Institutions, Hyderabad, in 2019. He has a year experience as an AI Engineer. He is currently the Tech Lead with Hexvel Disruptor, an AI-IoT based startup in India which deals with leveraging IoT and AI in the Sports realm. His primary areas of interest are computer vision and pattern recognition. He has completed and awarded a certificate in Artificial Intelligence and Machine Learning as an Excellent Performer grade from IIIT-Hyderabad and Talentsprint combined AI/ML Certification Programme. He has particular expertise in computer vision and been an Investigator in facial recognition, facial anti-spoofing, facial emotion and micro-expressions, GAIT analysis, and deep fakes. He recently published a conference paper in International Conference on Machine Vision on GAIT analysis. In addition to this work, he currently collaborates with his mentor Dr. Farukh Hashmi and other researchers to build momentum in computer vision space.



AVINASH G. KESKAR received the B.E. degree (Hons.) from VNIT, Nagpur, in 1979, and the M.E. degree (Hons.) from IISc, Bangalore, in 1983. He has 26 years of teaching experience and 7 years of industrial experience. He is currently a Professor with the Department of Electronics and Communication Engineering, VNIT Nagpur. His current research interests include computer vision, soft computing, and fuzzy logic etc. Dr. Keskar is a member of IAENG and a Senior Member of FIETE, LMISTE, and FIE.



NEERAJ DHANRAJ BOKDE received the M.E. degree in embedded systems from the EEE Department, BITS Pilani, Pilani Campus, India, and the Ph.D. degree from the Electronics and Communication Department, Visvesvaraya National Institute of Technology, Nagpur, India. He is currently a Postdoctoral Researcher with the Department of Engineering, Renewable Energy and Thermodynamics, Aarhus University, Denmark. His research interests include time series analysis, data mining, and prediction methodologies. He is serving as the Guest Editor in *Energies* journal.



ZONG WOO GEEM (Member, IEEE) received the B.Eng. degree from Chung-Ang University, the Ph.D. from Korea University, and the M.Sc. degree from Johns Hopkins University. He researched at Virginia Tech, University of Maryland, College Park, and Johns Hopkins University. He invented a music-inspired optimization algorithm, Harmony Search, which has been applied to various scientific and engineering problems. He is currently an Associate Professor with the Department of Energy IT, Gachon University, South Korea. His research interest includes phenomenon-mimicking algorithms and their applications to energy, environment and water fields. He has served for various journals as an Editor (Associate Editor for *Engineering Optimization*, a Guest Editor for *Swarm and Evolutionary Computation*, the *International Journal of Bio-Inspired Computation*, and the *Journal of Applied Mathematics, Applied Sciences*, and *Sustainability*).

• • •