Virtual Stylist: A Recommendation Based Virtual Try-On System

Sukanya Saha

I. INTRODUCTION

Have you ever looked at an Instagram model and wondered - "This summer look is a so gorgeous! how would it look on me?"(see fig 1). This problem requires an application that would recommend similar products and let customers try on these items digitally. This paper proposes a novel computer vision-based technique called Virtual Stylist to tackle two emerging problems of fashion e-commerce platforms- 1. Recommend similar fashion products, 2. Virtual Try-On based on those products. This can be beneficial for e-commerce giants like Amazon, Macy's, Walmart, Gap, Nordstrom, etc. to provide customers with a platform to try out different fashion looks, outfits based on a customer image and a style image.

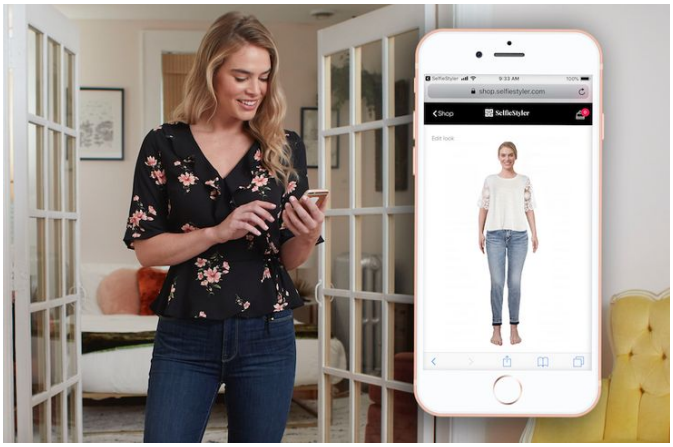

Fig. 1. Virtual Stylist [4]

Existing work on fashion products offerings focus either on recommending similar products or synthesizing images from a customer and product image. However, little has been done on an end-to-end system on recommending products as well as providing a virtual Try-on platform. Additionally, while an increasing number of studies have been conducted, the resolution of synthesized images is still limited to low. This acts as the critical barrier against satisfying online consumers. While recent works in virtual clothing try-on feature a plethora of possible architectural and data representation choices, they hardly improve the quality of generated images. This paper will combine a recommendation task and a virtual try-on task and generate high-resolution images to satisfy customer needs.

This paper introduces a baseline approach for building a virtual stylist system incorporating deep learning, generative adversarial network, and recommender systems. This architecture consists of two cascaded networks i) Fashion recommender network and ii) Image Synthesizer Network. The fashion product recommender network is based on human keypoint detection and object recognition. Whereas the image synthesizer network is a self-supervised model that given a reference image, I $\in R^{3\times H\times W}$ of a person and a clothing image, $c \in R^{3\times H\times W}$ (3, H and W denote the rgb channels, image height and width, respectively). The goal of this network is to generate a synthetic image, $\hat{I} \in R^{3\times H\times W}$ of the same person wearing the recommended target clothes c, where the pose and body shape of I and the details of c are preserved. In this model clothing-agnostic person representation is used to retrieve the pose map and the segmentation map of the person to eliminate the clothing information in I and finally generate $\hat{I}$). For experiments and evaluation of photo-realism, this paper utilizes the Deep Fashion dataset [10] and the High-Resolution Virtual Try-On (VITON-HD) dataset [9].

Fig. 2. Virtual Try On [5]

Finally, one application of this project can be social media advertisement eg. an Instagram fashion influencer want to

recommend a look for their followers to try and redirect the customers to the recommended product pages. This project can also open many doors for future research on recommendation based virtual try-on. For this project, both of these images should have overlapping body regions where the product needs to be fitted. Here, the challenge is synthesizing occluded body parts after fitting the recommended clothes on the customer image. Also, it is important to recognize a person's pose, body shape, and identity and deform the clothing product based on the person's posture without losing product details.

II. Related Work

A. Recommending Similar Fashion Products

Researchers over the years have proposed different methods that recommend similar fashion items given a user query or/and Product Display Image. This approach aims to recommend similar products for all fashion items and can boost customer engagement. I will talk about two major topics in recommending fashion products-

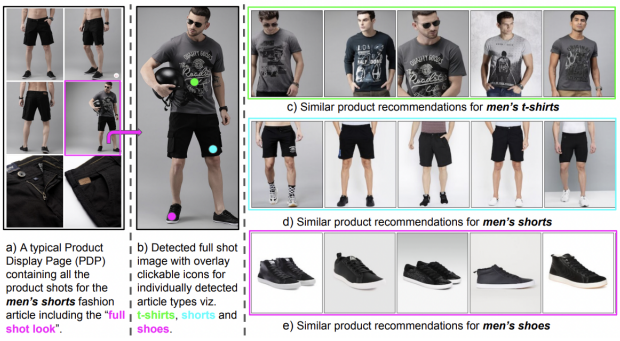

Fig. 3. Buy Me That Look: products recommendation [21]

1) Human Keypoint Estimation: Some of the noticeable works in keypoint estimation are Cascaded Pyramid Network (CPN) [8] , which has been dominant on the COCO 2017 key-point challenge. The Hourglass method [19] played a vital role in the MPII benchmark [6]. The CMU-Pose method [7] used a bottom-up approach that makes use of Part Affinity Fields. However, the Simple baseline pose estimation paper [27] outperformed the above methods by using optical flow-based pose propagation and similarity measurement. This method has been used in the Buy Me That Look [21] model whose architecture I am using in this paper.

2) Object Detection: The object detection task involves not only recognizing and classifying every object in an image but also localizing each one by determining the bounding box around it. Deep learning has been widely used for object detection. Researchers have been keen to use darknet architecture-based single-stage detectors like YOLO [22] for the task of real-time object detection. However, this project's goal of object detection does not require real-time output hence, this component of my pipeline can be done offline. So, the best choice would be to pick a model with a better mean Average Precision (mAP) score, while disregarding the run-time latency. The Mask RCNN [12] method has been chosen for this purpose as proposed by the Buy Me That Look [21] model.

3) Embeddings Learning: Embedding learning [29] [24] aims to learn representations of raw images so that similar images are grouped while moving away from dissimilar ones. My work follows the proposed method by the Buy Me That Look model that makes use of embedding learning to obtain representations, and compute image similarity.

My paper makes use of the proposed method mentioned above for more sophisticated application of product recommendation based on any Product model image given by the user as opposed to querying from existing database images. Additionally, this paper also synthesises the customer's look based on the recommended product which previous models did not take into account.

B. Virtual Try-On via Generative Adversarial Networks

There have been a large number of publications since the inception of virtual try-on methods in 2017. The focus areas of the methods are to parse human, segment products and body, estimate pose. Pose information has been embedded through DensePose model [23] and there exists impressive architectures for body and product segmentation.

In recent works of Conditional generative adversarial networks (cGANs) [18] researchers have utilized class labels [20] and attributes [25] to improve the image generation process. Some recent cGAN papers have tried to generate high resolution images. But, it causes blurry image generation when trying to handle large spatial deformation from input to output image. Moreover, methods using conditional GANs for try-on models were iteratively improved by removing adversarial loss, introducing perceptual loss, and experimenting with different extended network architectures to synthesize more detailed virtual try-on outputs. Most of the proposed approaches for these methods follow a two-stage architecture that involves cloth warping and person rendering. A two-stage approach has enough expressiveness for the model to learn how to do virtual try-on. In this paper, I propose a method that can address the spatial deformation of input images and properly generate customer images with the recommended products.

1) Image-to-Image Translation and StyleGAN: Deep Generative Adversarial Networks and StyleGANs [14] (Figure 4) have shown great potential for synthesizing high-quality photo-realistic images. It is very efficient and accurate in generating high resolution realistic images and can be done using the pre-trained model. My work focuses on designing a pose-conditioned GAN with precise control on the localized appearance (for virtual try-on) and pose (for reposing). Image-to-Image Translation [13] provides a general framework for mapping an image from one visual domain to another. Recent advances include learning from unpaired dataset [30], extension to videos. Similar to many existing human reposing methods, my work can be thought of as an image-to-image translation problem that maps an input target pose to an RGB image with the appearance from a source

image. This paper uses spatial modulation in StyleGAN for detail transfer.

Picture: *These people are not real – they were produced by our generator that allows control over different aspects of the image.*

Fig. 4. StyleGAN [14]

2) Conditional Image Synthesis: In recent works of Conditional generative adversarial networks (cGANs) [18] researchers have utilized class labels [20] and attributes [25] to improve the image generation process. Some recent cGAN papers have tried to generate high resolution images. But, it causes blurry image generation when trying to handle large spatial deformation from input to output image. Moreover, methods using conditional GANs for try-on models were iteratively improved by removing adversarial loss, introducing perceptual loss, and experimenting with different extended network architectures to synthesize more detailed virtual try-on outputs. Most of the proposed approaches for these methods follow a two-stage architecture that involves cloth warping and person rendering. A two-stage approach has enough expressiveness for the model to learn how to do virtual try-on. In this paper, I propose a method that can address the spatial deformation of input images and properly generate customer images with the recommended products.

3) Attention and Activations Functions: Recent works in Attention and transformers [28] have been thriving in outperforming state-of-the-arts with extended and improved architectures especially in the machine translation task. Even though self attention has been evolving in image processing and computer vision application, only a handful of papers have been published in the try-on spectrum utilizing self-attention techniques. The first paper that looked into it is called ShineOn [15] which verifies its power to model the longer-range dependencies when transferring the garment to the model person. In my paper, I am using a self attention model as proposed by them to utilize its ability to attend to visual and spatial regions of importance and it has been placed where the feature map depth is greatest. This paper also investigated that ReLU networks might have a bias towards learning low-frequency information and showed that smoother activation functions are more effective at achieving robust results for representing and reconstructing media.

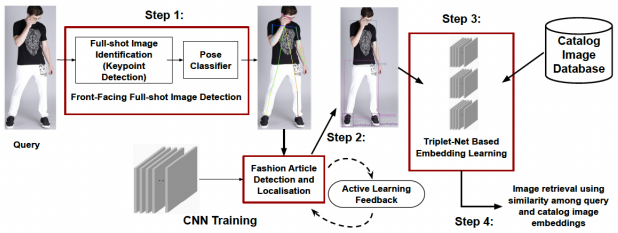

Fig. 5. Recommender Network Architechture [21]

III. Methods

This architecture consists of two cascaded networks i) Fashion Image recommender network and ii) Image Synthesizer Network. First I will talk about the Fashion image recommender network.

A. Fashion Image recommender network-

This network architecture can be derived in four modules as stated below. In this part of the architecture, I followed the architecture that is mentioned in the Buy me that look [21] paper(Figure 5).

Front-facing Full-shot Image Detection- Product Display Page (PDP) image contains images of the product in different views or angles, including the full-shot look image. This step identifies full-shot images using human keypoint detection such as head, ankle, etc. For this, they chose a ResNet [17] base architecture for feature extraction. They added three deconvolution layers after the final layer of the Resnet backbone with batch normalization and ReLU. See fig 6 They used 256 filters in each layer. The predictions generate a k-key-points heatmap. These images can be front, back, side facing and cause occlusion-related problems. Hence they added another sub-component for performing pose classification of the image.

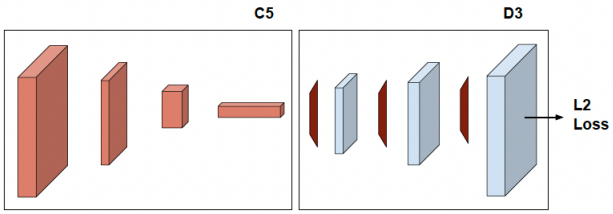

Fig. 6. Key Point Detection [21]

Fashion article Detection and Localisation- In this stage, they used the front-facing full shot look images from stage-1 and passed those in stage-2 a CNN network with active learning detects the fashion objects in the image and also does localization. The idea is to recommend similar product images with cropped regions of interest on the front-facing full shot look images. They trained the article type detection and localization module using the bounding box tags for the fashion articles present in these images. For the fashion

article detection and localization task, they trained the Mask RCNN model. This provides bounding box locations and classification for around 20 apparel types.

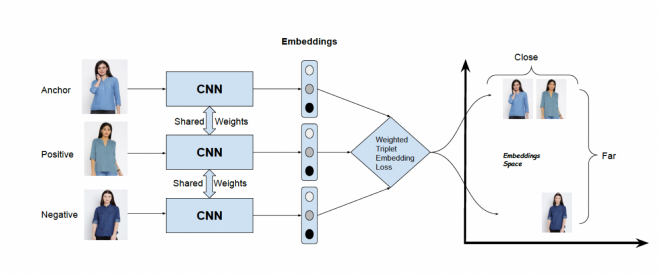

Fig. 7. Embeddings Learning [21]

Embedding generation for article types- In this module, embeddings are made of all images available in the catalog and stored in the database. Then, I am retrieving similar images from the database to the query image. To achive this authors, used the extracted relevant fashion articles from the full-shot look image from step 1 of this module. They needed to represent these article types and the products in the database, in a common embedding space that groups together similar articles while moving away dissimilar ones. Here they used a triplet based architecture to learn the embeddings which consists of three identical Convolutional Neural Networks (CNN) with shared weights (each of which may be regarded as a branch). they also tested different configurations of ResNet. See fig 7.

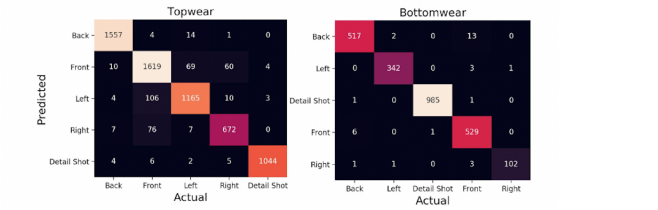

Fig. 8. Embeddings Learning Heatmap [21]

The triplet images are used in this architecture such that the first two are similar images whereas the last one is a dissimilar image. The objective here is to bring the similar of positive images closed in terms of embeddings and move away the embeddings of the dissimilar image from the other two.

Since the steps/modules are loosely connected so there is no need not to stick to algorithms that are used in the paper. Hence, these steps can be easily reproduced by experimenting with a new set of algorithms for the new dataset. One can use this as a base architecture and try various sets of algorithms that fit this or a new dataset.

B. Image Synthesizer Network: Pose-conditioned StyleGAN2-

I am using the model architecture proposed by the TryOnGAN [16] paper. Generative Adversarial Networks

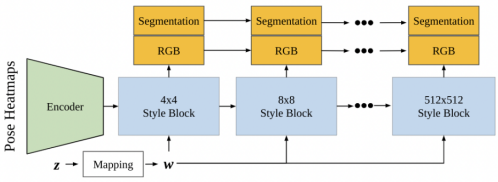

Fig. 9. Modified StyleGAN2 Architechture [16]

(GANs)[11] have been shown to synthesize impressive images from latent codes. As discussed in related work section, the idea to combine progressive growing and adaptive instance normalization (AdaIN) with a novel mapping network between the latent space, Z, and an intermediate latent space, W, encouraged disentanglement of the latent space. Transforming intermediate latent vectors wW into style vectors s further allowed different styles at different resolutions. Furthermore, recently developed StyleGAN2 inversion methods enable the projection of real images into the extended StyleGAN2 latent spaces, Z+ and W+ [26]. Motivated by those advances they choose StyleGAN2 as the base architecture. Hence, I am using this pre-trained StyleGAN2 model on fashion images, with key modifications on pre-condition(Figure 6).

C. Try-On Optimization

After training the model based on StyleGAN2, the authors generated a variety of images in both RGB and segmentation mask formats. But the other way is to project the pair of images to the latent space of the generator using an optimizer to compute the latent codes that minimizes the perpetual distance between input and the image from the generator. Linear combinations of these latent codes will produce images that combine various characteristics of the pair of input images. The desired try-on image is what the garment from the second image is transferred to the person from the first image lies somewhere within this space of combinations.

This generates the final output using an interpolation coefficient which proposes a greedy approach to select binary query vectors that reduces changes within the region of interest while minimizing changes outside of the region of interest.

The researchers also optimized the query vectors using manual clustering technique to define the semantic regions. Hence, they implemented an optimization loss term that preserves the the identity of the person while the style is based of garment.

The loss function is -

$L = \alpha_1 L_{localization} + \alpha_2 L_{garment} + \alpha_3 L_{identity}$

Editing-localization Loss- They also used an editing localisation loss term to the network to keep the interpolation only to the region of interest. More specifically the region of interest is the overlapping space between semantic regions in the image and the activation tensors. Then they used the segmentation output from the network to define semantic

cluster memberships. Those segmentation are converted to binary cluster heatmaps. For each layer, the heatmaps are down sampled. This is is to correct the resolution. The activation tensors are normalized per channel by subtracting the mean of each channel and dividing by the standard derivation of the channel. M is then calculated as-

$M_{k\times c} = \frac{1}{NHW}\sum_{n,h,w} A^2 \odot U$

This value is calculated for every pair of garment and person image. Say the segmentation of interest is i, then the authors calculated the the least relevant activation channels by subtracting the ith row of each M matrix from the k rows in that matrix and max over each channel. Finally, the localization loss is calculated as

$L_{localization} = \sum M_c^i \odot q_c$

Garment Loss- This loss defines the shape, texture, area of interest of the garment. For this researchers used VGG embeddings to compute the perpetual distance between the garment areas of two images- style image and the generated image. Hence, they calculated the binary mask for the garment in both images. On the RGB image they used element wise multiplication for mask calculation, blurred it with a Gaussian filter and down-sampled it. Then calculated the distance between two masked images-

$L_{garment} = d\left(I^g_{GarmentMasked}, I^t_{GarmentMasked}\right)$

Identity Loss- This term helps the network to preserve the identity of the person. For this loss the paper used hair and face as main identity of the person image. Then they calculated the loss person image and the generated try-on image as -

$L_{identity} = d\left(I^g_{identityMasked}, I^t_{identityMasked}\right)$

Projection- For testing this algorithm on any random image, first the authors projected the real image into the extended latent space Z+. Then they used the optimization algorithm proposed in the paper that learns a latent vector z per layer and results in the final image with identity of the person image and garment of the style image. This optimization also uses a perpetual loss to identify the latent vectors. Then using the pose-conditioned network and condition on the pre-computed pose of the image they project the output image.

IV. Experimental Design-

A. Dataset

I am using the DeepFashion2 [10] for training the recommendation system and the VITON-HD [9] dataset for training the try-on module. Both the dataset has high-resolution images with VITON-HD having 1024×768 virtual try-on data. DeepFashion2 is a benchmark dataset curated for clothes detection, pose estimation, segmentation, and retrieval. It has 801K clothing items where each item has rich annotations such as style, scale, viewpoint, occlusion, bounding box, dense landmarks. There are also 873K Commercial-Consumer clothes pairs. The VITON-HD dataset has been generated by crawling 13,679 frontal-view women and top clothing image pairs on an online shopping mall website. Since this datasets are huge and it will take weeks to train models on these, I am using 10% sample of these datasets.

B. Fashion article Detection and Localisation

Inside the Fashion image recommender network, I am training the Mask RCNN on a custom training on the DeepFashion2 dataset to generate the bounding box location and classification from across different apparel types. I am able to obtain around an average mAP(Mean Average Precision) of 65% for all the classes while reaching as high as 80% for some of the top-wear classes (shirts and t-shirts). These values are in line with this model's performance on the Microsoft Common Objects in Context (MS COCO) data set [17], which is around 60.3% at IOU 0.5. This is justified because the COCO dataset has more classes with natural real-world images, whereas images in my case are from a lesser number of article categories.

C. Pose Conditioning-

Pose conditioning is crucial to omit entanglement of pose image and style image's latent space. Otherwise, pose and and style of the respective images may not be preserved. Hence, the authors conditioned the StyleGAN2 on pose. This results in replacement of constant input at the beginning of the generator with an encoder that takes an input of 64 X 64 resolution of pose representation. Then they also used PoseNet base architecture for key point detection of the images. Then they also created a 17 pose heatmap corresponding to these key points which serves as input to the encoder.

Fig. 10. Segmentation and Image Generation [16]

D. Segmentation Branch

To improve disentanglement authors also used segmentation mask of the actual image. This removes dependencies on existing segmentation models during optimization. It generates an alternative output to the StyleGAN2 RGB branch's output. See fig 10

E. Discriminator-

In the Image Synthesizer Network, I have two discriminators, one for pose and one for segmentation. The pose discriminator receives as input either a real RGB image/pose heatmap pair or a conditional pose heatmap with the corresponding generated RGB image. The segmentation discriminator receives real/generated RGB image/segmentation

pairs. The two discriminators are weighted equally during training. To prevent overfitting of pose to style, I am using the following data augmentations on the pose input to the discriminator: 1) add gaussian noise to the normalized keypoint locations before creating the heatmap, and 2) drop keypoints with probability less than 0.4.

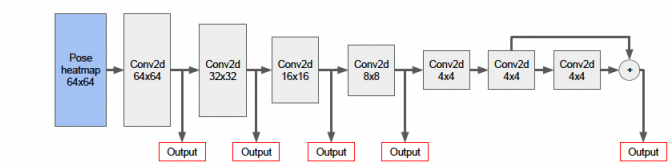

Fig. 11. Pose Encoder [16]

F. Retraining StyleGAN2-

TryOnGAN paper's conditional StyleGAN2 network was implemented in TensorFlow with 25 million iterations, on 8 Tesla v100 GPUs, for 12 days. Once the network was trained, they performed a hyperparameter search for the optimization loss weights. Since I am using a smaller sample of the dataset, I experimented with these parameters- batch size, gamma, augmentation. After trying different values for these parameters I am choosing these values which gave the best results- batch size = 32, gamma=5, augmentation= ada [14].

G. Evaluation Metrics-

In the Fashion image recommender network, for Front-facing Full-shot Image Detection Results, I make use of a ResNet18 network as the pose classifier to classify an image into one of the following categories: front, back, left, right, or detailed shot. For this part, I am using precision, recall for each of the categories and mAP(Mean Average Precision) since these work best for classification model evaluation.

For the Fashion article Detection and Localisation results, I am treating it as a classification task of bounding box with evaluation metrics as IOU of threshold 0.5 for measuring ground truth as IOU works best for object recognition related tasks. Results on embedding generation for article types are judged using cosine similarity among embeddings.

For Image Synthesizer Network, I am using two metrics to compare the methods and types of images: FID(Frechet Inception Distance) [1] to evaluate photorealism and ES (Embedding similarity) to evaluate the quality of try-on or how similar is the result to the input in the garment part. These two metrics efficiently evaluate the photorealism and garment quality of the Image Synthesizer Network as proposed in the TyOnGan paper.

H. Experimental Result-

1) Qualitative Result: : I experimented with a few sample images [3] and I will talk about two such samples. In fig 7, the image on the first row first column is a look image based on which the next image on the right is generated. Here, you can see the recommendation is based on color and sleeve length. The future scope of this work could be to recognize the materials of the garment for product suggestions. The identity of the person image is preserved which is a bit easier here since there is very little need for skin or garment material generation. Another scope for improvement would be to preserve the texture of the garment in the generated image as well since here the velvety material of the garment image becomes silky in the resulting image.

In the second set of samples [2], the recommended image again resembles the color, length, and collar type of the garment. The generated image tries to capture the pose and body shape of the person image. Here, it requires generating some garment material on the person's left arm. Though it does a good job in suppressing some garment material on the right hand it struggles a bit on the generating materials. This is a major challenge of this project to generate skin and garment material and can be improved in future work.

Fig. 12. Experimental Result [2], [3], [16]

2) Quantitative Result: : For both recommender and image synthesizer modules I used the pre-traimed models from the papers ShopLook and TryOnGAN. For testing the performance of those models, I used test set and evaluation metrics suggested in those papers. For the recommender module, I used their test set which yielded precision and recall of 27% (P@3) and 14.3% (R@3). Then I tested the image generator module on their test set and achieved Fréchet Inception Distance of 35.9% and embedding similarity (ES) score of 0.31. Authors also ran a study with crowd source workers to check if the style and pose is preserved according to human judgement. I refrain from this last part of testing due to limited resources.

I. Conclusion-

In this project, I have proposed an end-to-end virtual stylist platform to be used in fashion e-commerce to suggest customers products based on a look image and let them try those products virtually. For recommendation, I have used a neural network-based architecture with the addition of pose classification and embedding generation. For try-on,

a StyleGAN2 based architecture with pose conditioning is used.

For future work, the recommendation model can be improved to suggest garments based on similar materials of clothing and the try-on module needs to disentangle body pose and skin generation in a better way. The try-on module can use more latent variables to help interpolation blocks better preserve the person's identity and garment texture.

REFERENCES

[1] frechet-inception-distance. https://machinelearningmastery.com/how-to-implement-the-frechet-inception-distance-fid-from-scratch/. Accessed: 2021-11-17.

[2] pinterest. https://www.pinterest.com. Accessed: 2021-11-17.

[3] pinterest2. https://www.google.com/imgres?imgurl=https Accessed: 2021-11-17.

[4] Virtual fitting room. https://in.pinterest.com/pin/345369865172616862/. Accessed: 2021-11-09.

[5] Virtual try-on. https://venturebeat.com/2020/06/05/amazons-new-ai-technique-lets-users-virtually-try-on-outfits/. Accessed: 2021-11-09.

[6] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[7] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields, 2017.

[8] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation, 2018.

[9] Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization, 2021.

[10] Yuying Ge, Ruimao Zhang, Lingyun Wu, Xiaogang Wang, Xiaoou Tang, and Ping Luo. A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images. *CVPR*, 2019.

[11] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018.

[13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks, 2018.

[14] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks, 2019.

[15] Gaurav Kuppa, Andrew Jong, Vera Liu, Ziwei Liu, and Teng-Sheng Moh. Shineon: Illuminating design choices for practical video-based virtual clothing try-on, 2021.

[16] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. Tryongan: Body-aware try-on via layered interpolation, 2021.

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

[18] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014.

[19] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation, 2016.

[20] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier GANs. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2642–2651. PMLR, 06–11 Aug 2017.

[21] Abhinav Ravi, Sandeep Repakula, Ujjal Kr Dutta, and Maulik Parmar. Buy me that look: An approach for recommending similar fashion products, 2021.

[22] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection, 2016.

[23] Iasonas Kokkinos R{iza Alp Güler, Natalia Neverova. Densepose: Dense human pose estimation in the wild. 2018.

[24] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015.

[25] Wei Shen and Rujie Liu. Learning residual images for face attribute manipulation, 2017.

[26] Jonas Wulff and Antonio Torralba. Improving inversion and generation diversity in stylegan using a gaussianized latent space, 2020.

[27] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking, 2018.

[28] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France, 07–09 Jul 2015. PMLR.

[29] Chin-Chia Michael Yeh, Dhruv Gelda, Zhongfang Zhuang, Yan Zheng, Liang Gou, and Wei Zhang. Towards a flexible embedding learning framework, 2020.

[30] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020.