International Journal of Science and Research (IJSR)
ISSN: 2319-7064
ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426

Novel Approach to Fashion Design using Artificial Intelligence

Ashwin Kherde[1], Prashant Jawade[2], Indrajeet Mane-Deshmukh[3], Pradnyoday Mirajkar[4]

[1, 2, 3,4]Department of Information Technology, Government College of Engineering Karad, Maharashtra, India

Abstract: ***The task of fashion designing now days is becoming more and more difficult due to increasing expectations of the consumers. For this, Artificial Intelligence is going to aid the growing need for variety of designs and their faster production which will also minimize the huge loss in production. Recent approaches include use of deep neural networks and here we are extending their use to generate the apparel designs. A class of deep neural networks known as Generative Adversarial Networks (GANs) which are comprised of two networks namely generative and discriminative. Specifically, we are using Deep Convolutional Generative Adversarial Networks (DCGANs) have certain constraints on their architecture which help them to be a good prospect for unsupervised learning. Additionally, we show that the proposed model functions generatively i.e., according to the user preferences, it can generate new images of clothing items similar to the user taste.***

Keywords: Fashion Designing,Generative Adversarial Networks, Artificial Intelligence, Deep Convolutional Generative Adversarial Networks

1. Introduction

The goal here is to make use of Artificial Intelligence as an aid to the fashion designers so that they can use the results produced with the help of AI as an initial product and apply their skills to produce even greater designs [9], [10]. This will be done with the help of large amount of historical user data collected from different sources (like online shopping stores). That is, training different algorithms which will learn the features from the available data. It can be challenging, particularly in domain like fashion for a number of reasons: new products are introduced continuously; preferences and style requirements of the user changes over time; in addition to that, the rules that define what is 'fashionable' are highly complex.

With the advancements in Computer vision and deep learning, Convolutional Neural Networks (CNNs) have been used to learn visual features from the image datasets.

Additionally, there is some work in the domain of generative networks; which involve developing new image representations by using convolutional neural networks.

Here, we try to combine these two fields of work to produce a system which will learn different features from image datasets and create new image representations with similar features. The idea is to use GANs [1] to develop multiple variations of fashion images, which will allow us to explore wide range of fashion items, modify the ones which already exist and generate new ones by applying human fashion designing skills on the results generated by GANs.

The generated representations are consistent but different from those in the training dataset; which allows us to use them in fashion designing.

Methodologically, our system builds upon Generative Adversarial Networks (GANs) which consists two types of neural networks: generative and discriminative. Generator replicates the data that is fed into the model, without ever interacting with the original data samples. Instead, this model learns to reproduce the data patterns by its interactions with the discriminator.

On the other hand, the role of the discriminator is to discriminate the original data samples from the ones created by the generator. On each run through the model, the generator adapts its output so that, it would be difficult for the discriminator to tell which is the original one and which one is artificial. Then, both generator and discriminator networks learn in parallel in order to perform their respective jobs perfectly. It can be seen from the Fig. 1., that some random variable is given as input to the generator (G); which in turn produces an output y which along with real data sample x is fed to the discriminator (D). The predictions made by discriminator are then supplied back to the generator and discriminator as a feedback.

Use of Artificial Intelligence in Fashion design can be further extended for artistic recommendation systems [11].

2. Related Work

Our work is related to various fields of research and we would like to focus them on the following:

2.1 Visually aware recommender systems

Recently, visually aware recommendation systems are introduced where user ratings are created in terms of visual signals as images of product. Such models successfully recommend alternative items (two similar to each other) and complementary items (trousers with matching t-shirts). Recommender systems helps in establishing connection between the user and items based on past purchases, views and interactions with the items. The method of Matrix Factorization helps in relating users with items that are highly rated by them. The most challenging task is to distinguish purchased products from non-purchased products. In recent time, point-wise and pairwise methods have been successful in adapting Matrix Factorization to address the challenges.

Point-wise methods considers non-observed (non-purchased) feedback to be negative and assign regression values to them either by associating 'confidence levels' to feedback, or by sampling non-observed feedback as negative instances.

Pairwise methods work on the assumption that positive feedbacks are to be preferred more than non-observed feedback.

2.2 Fashion and clothing style

In addition to previous methods, modelling or creating fashion style and characteristics is becoming an important task example given for extraction of features from huge image datasets [3]. It mainly involves categorizing images relevant to a specific style as well as accessing items which are compatible. This means, recognition of not only the type of apparel, but also the material, patterns, etc. Therefore, it detects, classifies and describes how cloth would appear originally. There is a collaboration of multiple computer vision fields, a pipeline that approaches in several stages, from the detection of human face and upper body to classifying the clothes. Classification is done with the help of two classifiers, one is random forest to classify clothing types and Support Vector Machines to characterize the patterns of different apparels.

There are many applications for such systems in e-commerce, surveillance systems, online advertising up to a wide range of domains.

3. Methodology

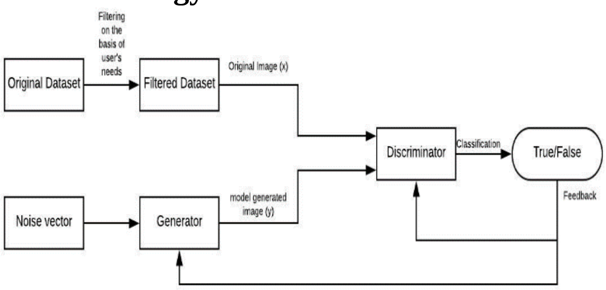

Figure 1: Proposed architecture of the model

At present – the fashion trend changes very rapidly due to (1) evolution of technology and (2) the availability of huge number of fashion retailers, which gives a wide exposure of current fashion trends to the consumers.

In the fashion industry, brands tend to block fabric anywhere between 18 months and two years in advance. Designs are frozen between 12 and 18 months ahead. And then starts the process of making the article (a T-shirt or a dress, and so on), six or eight months before it goes on sale. Since this involves mass production, within six months generally 30% of the batch is left unsold, which is typically cleared via discount sales. The global fashion business is pegged at USD2 trillion according to industry estimates. Of this at least 3%-4% is spent on design.

So, this is where our technology is likely to be useful, that is, in minimizing design cost while maximizing effectiveness, which will help fashion designers to produce faster results on the basis of our proposed architecture of the model given in Figure 1.

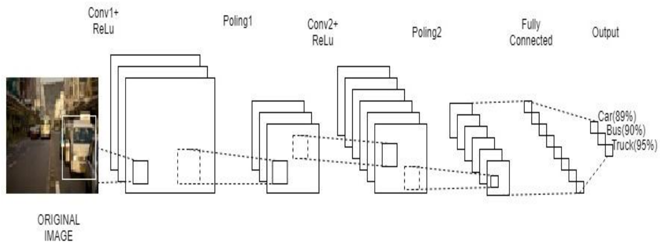

Figure 2: Convolutional Neural Network

3.1 Filtration of dataset

The dataset used here by us is the Amazon Fashion image dataset consisting of user reviews and product images, along with other product details such as title, product number, etc. It consists of thousands of images.

Following are the steps followed by us for pre-processing of the dataset:

1) Initially, images of different fashion products were mixed together (such as shoes, clothes, etc.). Whereas DCGAN required the images to be properly classified according to their types for generating authentic fashion image data. So, we manually classified approximately 20,000 images into different categories.
2) After classifying a part of dataset manually, we trained a Convolutional Neural Network on the classified dataset images, by partitioning the classified dataset into train and test dataset.
3) Using this trained classifier model we classified and downloaded the rest of the dataset images.
4) The images in the dataset were of random sizes (that is, different images were of different height and width), so we resized all of the images into images of size 64x64 as per the needs of DCGAN architecture.
5) After performing all these steps a well formed dataset was obtained.

CNN consists of convolution, pooling and fully connected layers which helps in extracting features from the image and then classifying them on the basis of the extracted features.
In Convolution layer particularly, a filter glides over the image that is provided as input to create a feature distribution. Convolution of another filter over the same image produces a different feature distribution. By repeating this process multiple times a complete feature map is produced. The size of the feature map depends on three features: depth of the filters used for convolution, strides (the shift in number of pixels when filter matrix moves over the input image matrix) and zero-padding (padding of input image matrix with zeroes around the border).

ReLU (Rectified Linear Unit) is the activation function. It is used to further improve the training of the neural networks. It works according to the relation $f(x) = \max(0, x)$. It produces 0 when $x < 0$ and a linear function when $x \geq 0$. Following is the graphical representation of the function:

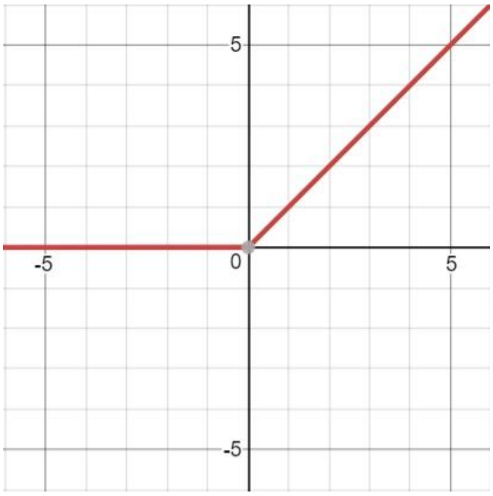

Figure 3: Activation function (Rectified Linear Unit) produces 0 as an output when x<0, and for x>0, it produces a linear graph with slope 1

Pooling (or down sampling) is used after every convolution for dimensionality reduction of feature map and retain the necessary information. In particular, pooling (1) reduces the input feature dimensions and makes it easy to manage, (2) minimizes the number of parameters and in turn the computations required, which helps in controlling the over fitting of data, (3) allows the network to resist small distortions and transformations in the input.

Multi-Layer Perceptron is a fully connected layer in which softmax is used as an activation function for the output layer. Results obtained from convolutional and pooling layers are the features extracted from the image which is given as the input. Fully connected layers will then use the extracted features to classify the input images.

The Convolutional Neural Network used here is based on CNN-F architecture from [5]. It is an efficient architecture. Better performance can be achieved by using some more powerful architectures (eg. ResNet [6]), but for now CNN-F is sufficient to present our method and can be easily trained on standard desktop hardware. Particularly, CNN-F consists of 8 layers in total, with 5 convolutional layers and 3 fully-connected layers.

Structural details of the architecture used by us are as shown below (see [5], Table 1 and Table 2 for details; st=stride; pad=spatial padding);

Table 1: Architecture of CNN(Layers 1-4)

Con1	*Con2*	*Con3*	*Con4*
64x11x11	256x5x5	256x3x3	256x3x3
st. 4, pad 0	st. 1, pad 2	st. 1, pad 1	st. 1, pad 1
x2 pool	x2 pool	-	-

Table 2: Architecture of CNN(Layers 5-8)

Con5	*Full6*	*Full7*	*Full8*
256x3x3	K1 drop-out	K2 drop-out	1000 soft-max
st. 1, pad 1			
x2 pool			

The main difference between the two architectures are the variable values of dropouts which allows the algorithm to train the algorithm in a better way.

3.2 Generative Adversarial Networks

As given in [7], deep convolutional generative adversarial networks (DCGANs) are very vital for unsupervised learning. DCGANs (see Fig. 4.) are produced by applying some rules on Convolutional GANs.

The architectural features of DCGAN include (1) strided and fractional-strided which are discriminator and generator respectively have taken place of pooling layers, (2) batchnorm is used in generator as well as in discriminator, (3) fully connected hidden layers are removed, (4) In all layers of generator ReLU activation is used but in output layer where Tanh is used, (5) Discriminator uses LeakyReLU activation for all layers.

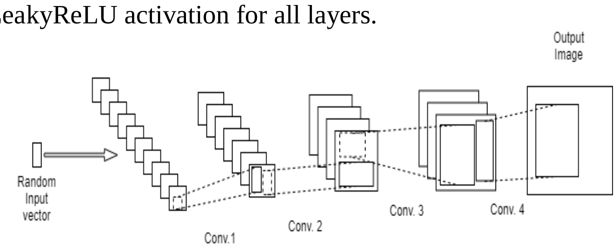

Figure 4: DCGAN Generator

As the generator generates the images, the discriminator (a CNN classifier) classifies the generated images as real (from the dataset) or fake (generated by the generator).

Training of GAN is done with the help of Least Square loss function in LSGAN[8], which allows the development of high quality images. The objective functions for our GAN are as follows:

$$\min_D V(D) = \mathbb{E}_{x\sim p_{data}(x)}[(D(x)-1)^2] + \mathbb{E}_{z\sim p_{data}(z)}[(D(z)-0)^2] \quad (1)$$

$$\min_D V(G) = \mathbb{E}_{x\sim p_{data}(x)}[(D(G(z))-1)^2] \quad (2)$$

By above equation, discriminator (D) predicts ‘1’ if images are real and ‘0’ if images are fake and the generator (G) tries to convince the discriminator that the images generated by it are real.

Generator structure in DCGAN:

It consists of five layers, each with a deconvolution layer (nn.ConvTranspose2d()), Batchnorm (nn.BatchNorm2d), and an activation function nn.ReLU() for each layer except the output layer which uses Tanh() as the activation function. There are five layers in total including output layer.

Discriminator structure in DCGAN:

It consists of five layers each with a convolution layer (nn.Conv2d()), Batchnorm (nn.BatchNorm2d()) and an activation function nn.LeakyReLU() except for the output layer which uses Sigmoid() as the activation function. There are five layers in total including output layer.

Pseudo code for DCGAN:

Generator ()

{

#Layer 1

```
nn.ConvTranspose2d(arg1, arg2, ……)
nn.BatchNorm2d(arg1, arg2, ……)
nn.ReLU(arg)
.
#Layer n
}

Discriminator ()
{
#Layer 1
nn.Conv2d(arg1, arg2, ……)
nn.BatchNorm2d(arg1, arg2, ……)
nn.LeakyReLU(arg)
.
#Layer n
}
```

4. Findings and Analysis

4.1 Filtration of dataset

We trained above stated CNN model to classify the dataset and obtain the images of purely one type in order to obtain clear and high resolution images.

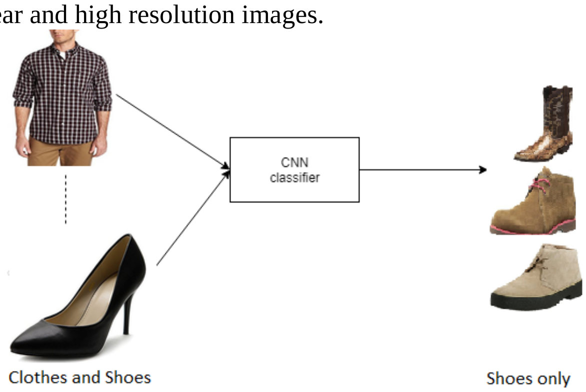

Figure 5: Image classification using CNN

The images classified using CNN were then resized and used for training the Generative Adversarial Network.

4.2 Fashion style generation

We trained our models on Google Colaboratory which is a research tool for machine learning education and research by Google with high compute power.

We used DCGAN to produce the fashion data with features similar to the filtered dataset by training the model for 30 epochs, which approximately took 6-7 hours to train.

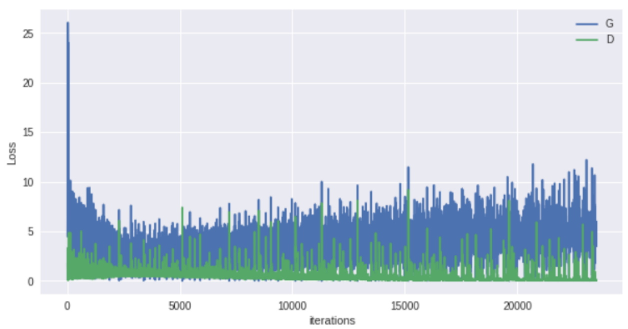

Figure 6: Generator and Discriminator loss during training

The loss graph shown in Figure 6 presents the loss of Discriminator (D) and Generator (G) over the iterations.

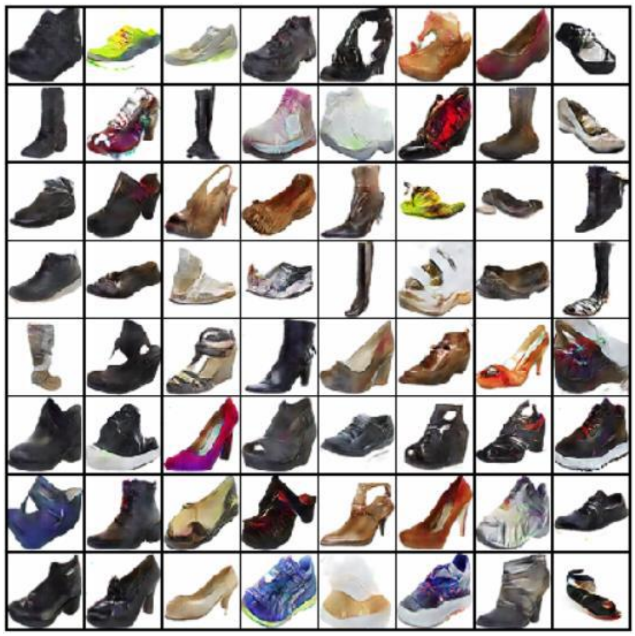

Figure 7: Result images generated using DCGAN from the original fashion dataset

Figure 7 shows the sample of resultant fashion images generated by DCGAN model.

It can be seen that most of the images can be recognized as a footwear. Visibility and resolution of the images can be improved further by increasing the number of layers in the Generator and Discriminator networks and training the model for a longer duration.

5. Conclusion

To sum up, our work has led to the conclusion that we can produce new fashion data using Artificial Intelligence. In particular, deep neural networks are of great importance in this field and also for unsupervised learning. Deep convolutional Generative Adversarial Networks and Least Squares Generative Adversarial Networks, produce significant results in the generation of image data which, in future can be of great help to all the people in fashion industry as well as the consumers. Our methods can be improved in future to produce even more realistic and high resolution images.

References

[1] Kunfeng Wang, Chao Gou, Yanjie Duan, Yilun Lin, Xinhu Zheng, and Fei-Yue Wang," Generative Adversarial Networks: Introduction and Outlook," IEEE/CAA Journal of Automatica Sinica, Vol. 4, No. 4, October 2017.

[2] J. McAuley, C. Targett, Q. Shi, and A. van den Hengel, "Image-based recommendations on style and substitutes," in SIGIR, 2015.

[3] L. Bossard, M. Dantone, C. Leistner, C. Wengert, T. Quack, and L. Van Gool, “Apparel classification with style,” in ACCV, 2013.
[4] Tienmei Guo, Jiwen Dong, Henjian Li, Yunxing Gao,”Simple Convolutional Neural Network on Image Classification,” in 2017 IEEE 2nd International Conference on Big Data Analysis.
[5] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, “Return of the devil in the details: Delving deep into convolutional nets,” in BMVC, 2014.
[6] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in CVPR, 2016.
[7] Alec Radford, Luke Metz, Soumith Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks,” arXiv:1511.06434v2 [cs.LG] 7 Jan 2016.
[8] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, “Least squares generative adversarial networks,” arXiv preprint ArXiv:1611.04076, 2016.
[9] Natsumi Kato, Hiroyuki Osone, Daitetsu Sato, Naoya Muramatsu, Yoichi Ochiai,” DeepWear: a Case Study of Collaborative Design between Human and Artificial Intelligence,” 12th International Conference on Tangible, Embedded, and Embodied Interaction.
[10] Haosha Wang, Khaled Rasheed,” Artificial Intelligence in Clothing Fashion”.
[11] R. He and J. McAuley, “Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering,” in WWW, 2016.

Author Profile

Ashwin Kherde is pursuing his B. Tech. degree in Information Technology from Government College of Engineering Karad, Shivaji University, Kolhapur.

Pradnyoday Mirajkar is pursuing his B. Tech. degree in Information Technology from Government College of Engineering Karad, Shivaji University, Kolhapur.

Indrajeet Mane-Deshmukh is pursuing his B. Tech. degree in Information Technology from Government College of Engineering Karad, Shivaji University, Kolhapur.