Generative Adversarial Network-based Visual Similarity Recommendation

P.A. Abdul Saleem¹*, L. Ravi Kumar², K.G.S. Venkatesan³

¹Department of Computer Science & Engineering (DS), CVR College of Engineering, Ibrahimpatnam, Telangana, India.
², ³Department of Computer Science & Engineering, MEGHA Institute of Engineering & Technology for Women, Edulabad, Hyderabad, Telangana, India.

Abstract: The purpose of a content-based recommendation system is to rank a set of results by how closely they match the user’s query. These days, virtually every online marketplace has some kind of recommendation system in place to steer customers to more goods and services they might enjoy. In this piece, we’ll go through how we used Generative Adversarial Networks to build an image retrieval system for e-commerce sites, with a focus on locating the best possible shoe photos to go along with a given product image. The writers of this paper conducted the study presented here. We demonstrate the outcomes of our study of the proposed deep learning network’s performance on a benchmark dataset, and we compare the network’s results to those of state-of-the-art systems.

Key words: Image Retrieval, Image Similarity, Deep Learning, Generative Adversarial Network.

1. INTRODUCTION

Content-based image retrieval (CBIR) [1] uses a query image as a starting point to scour a massive image database for similar images. By analyzing the photoscolours, forms, volumes, textures, and local geometries, among other visual features, CBIR systems help users find related images [2]. The steps
taken by the CBIR can be broken down into two categories: Similarity between image vectors can be calculated after feature vectors from each image have been retrieved. The quality of the feature vector used in the picture representation process is crucial to the effectiveness of such systems. One of the most difficult challenges is therefore to create a model for automatically extracting a feature vector from target photos that is quick, precise, and efficient. Another challenge is the labelling of a significant amount of training data. The generalizability of the learnt deep representations is compromised when supervised training is employed for all target images [3]. Interest in semi-supervised and unsupervised learning methods, which can be used to circumvent these limitations, has increased recently.

Generative adversarial networks (GANs), first proposed by Ian Goodfellow [4] in 2014, are an attempt to address the challenge of unsupervised learning. In the realm of semi-supervised and unsupervised learning, GANs are one of the most talked-about new methods due to their ability to learn deep representations from unlabeled training data. This makes them an exciting new direction in machine learning. The generator and the discriminator are both types of deep neural networks that make up a GAN. The generator network takes in unnatural-looking noise and outputs a picture that looks natural. To assess if an input is valid, the discriminator network applies the standard neural network classifier processes. Using GAN to create images with a realistic aesthetic aspect has yielded some promising results as of late.

2. RELATED WORKS

In recent years, AI’s creative capabilities have greatly beyond the reach of photorealism. In [5], we look at these and other uses. In order to retrieve deep features from GAN-generated images,[6] the core components of their system are a generator network, a VGGNet[7], and a discriminator network. They don’t just send the discriminator the real and fake photographs that they have in their system like that. The discriminator network receives the generated convolutional features from the pretrained model. Face images generated by their model appeared more lifelike than those generated by DCGAN[8] and DFC-VAE[9] when assessing clear facial components.

To function properly, e-commerce platforms require visual search and personalised product recommendations. Recommender systems that rely on collaborative filtering have struggled for a long time due to the computational complexity, scalability, and sparsity of large datasets. [10] Furthermore, the user’s click and buy history is given more weight than the actual content of the photographs when these systems make product recommendations. In this research, we provide a recommendation system that makes advantage of visual information processing to give the recipient the best possible guidance. To better serve our customers, we have developed and analysed numerous approaches to deep learning-based photo retrieval, as well as conducted comparative research, all in an effort to expand our selection of trendy shoe suggestions[11].

The rest of the paper follows the structure presented here. After a brief review of the existing literature (Section II), we will move on to discuss the Generative Adversarial Networks (GANs) that form the basis of our study (Section III). Section IV provides a comprehensive breakdown of the planned
network architecture and the train evaluations. The results of the experiment will be presented and analysed in Section V, and appropriate conclusions will be drawn in Section VI. Picture retrieval methods for various articles of apparel have swiftly become extremely popular as a direct result of the stratospheric expansion of e-commerce in recent years. Several studies [12]-[13] have retrieved and advised certain pieces of clothing for each buyer using methods like machine learning and predictive analysis. However, studies show that the complexity and subjectivity of the fashion idea for human visual appraisal might pose problems for users when trying to recover pictures. This is connected to how trends might influence how we interpret what we see.

3. INTRODUCTION TO GENERATIVE ADVERSAIRE NETWORKS
The Generator and the Discriminator are two independent networks that make up a Vanilla GAN, as can be seen in Figure 1. While the goal of is to estimate the likelihood that an image is real or false given a dataset that contains both real and fake images ($x = ..., x$), the purpose of is to produce inputs by mapping random noise. In contrast, the purpose of is to produce inputs by mapping random noise.

![Figure 1. Simple GAN structure](image)

Deep Convolutional GANs (DCGANs) consist of a generator network and a discriminator network in the same way as general adversarial networks (GANs) [14] do. The discriminator network works to determine whether or not a picture is genuine, while the generator network seeks to trick the discriminator network into believing that its images are genuine when in fact, they are phoney. In contrast to conventional GANs, the generator and discriminator in this architecture are both implemented as deep convolutional networks with the convolutional layers in the opposite order. Figure 2 depicts the architecture that was discussed in the initial DCGAN research paper[15].

A noise vector is used in both the input to the generator as well as the output to the network when utilising DCGAN. By transposing the layers of the convolutional network, new images are created. Both entirely linked layers and pooling layers are missing from the plan. In place of pooling layers such as max-pooling, the design now incorporates strided convolutions, which act as the discriminator, and fractional-strided convolutions, which act as the generator.
4. ARCHITECTURE FOR PROPOSED NETWORKS

The fundamental objective of this research is to develop a system for making suggestions that is predicated on the existence of visual similarities between items. The work that we carried out can be broken down into two main stages: During the first portion of this project, we look into a number of different methods for fast obtaining deep feature representations from a picture. We leverage the primary concept described in InfoGAN as a jumping off point for our work. This concept has previously been demonstrated to successfully utilise community knowledge in order to generate meaningful representations and realistic images. By skipping the first layer, we are able to take advantage of the deep representations learnt by these well-known pre-trained models for the task of shoe selection. As a result, we’ll be able to offer more helpful advice. Calculating the separation between the query image’s features and those of the images in the dataset is the next stage.

Discriminator model D takes as input a 128x128 colour image with a single input channel. The input to the Generator model G is a 108-dimensional vector made up of a 100-dimensional noise variable and an 8-dimensional latent coding that encodes class information. Our data is unlabeled (categorical code = 0), but we believe it naturally divides into eight distinct types. To keep the discriminator’s training consistent beyond the first layer, we employ Batch Normalisation. In order to avoid overfitting and memory issues. Third, after every convolution layer in the network, we employ a leaky ReLU layer. Fourth, a sigmoid activation function is utilised in the last layer to determine whether an image is authentic or not. The generator’s functionality includes the following methods: Each layer is then subjected to a Batch Normalisation process. For more photorealistic results, we dropout at a rate of 60% between each successive layer. Reason being, smaller dropout values (dropout 1) allow for the creation of more lifelike images. Tanh functions are used after the output layer, while ReLU functions are used after each transposed convolutional layer. See [8] for more information. Except for their respective terminal-layer output units, networks D and Q are structurally equivalent. Q’s activation function, tanh, is used at the top layer. Figure 3 displays the outcomes of the training that was performed on the network that was discussed. When deciding on a location, we give preference to those places that fall between 500 and 2000 points and have a gloss that is lower than the dreal. This
is due to the fact that the gloss values are significantly higher than the dreal values at these spots. To rephrase, generator has begun fabricating fake images that closely resemble those in the collection. This prevents the picture discriminator from telling the phoney photos from the actual ones. The difference between dfake and values is found to decrease after roughly 2500 iterations. This happens as the discriminator improves in its ability to separate similar objects. This is another proof that the generator has started making copies of real data. This resulted in a striking resemblance between the phoney and authentic images. This is why it is crucial to verify the information gathered at each stage.

![Figure 3. The Proposed Model Expenses](image)

5. RESULTS AND DISCUSSION
A server with 256 GB of RAM, a 24-core Intel Xeon E5-2628L CPU, and Ubuntu Server 16.04 is required to train the suggested deep neural network. The efficiency of the system is also evaluated with this tool. We only use the other eight NVidia GTX 1080-i GPUs on the server when we’re really training a model. The models are executed in an environment called Tensorflow. Mini-batches of one model are used for CPU testing. In the suggested model, the discriminator generates a feature vector of size [1, 1536]. The vector values are stored in the MongoDB database as arrays. Similarity scores between the query image and the retrieved images are calculated using the Euclidean distance. Test results for the proposed generative network as well as comparisons to other popular pre-trained models based on convolutional neural networks can be found in Table 1. We put the 10,000 shoe images available in the UT-Zap50K benchmark dataset through their paces by picking them at random and evaluating them on a number of different dimensions. The additional cost of inference time climbs, which may have a detrimental effect on the user experience. In contrast, the average execution time of
<table>
<thead>
<tr>
<th>Model</th>
<th>Inference Time (sec)</th>
<th>Size (MB)</th>
<th>Precision (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG16</td>
<td>0.32</td>
<td>518MB</td>
<td>0.74</td>
</tr>
<tr>
<td>VGG19</td>
<td>0.31</td>
<td>539MB</td>
<td>0.80</td>
</tr>
<tr>
<td>ResNet50</td>
<td>0.19</td>
<td>88MB</td>
<td>0.78</td>
</tr>
<tr>
<td>ResNet101</td>
<td>0.26</td>
<td>181MB</td>
<td>0.70</td>
</tr>
<tr>
<td>ResNet152</td>
<td>0.33</td>
<td>222MB</td>
<td>0.72</td>
</tr>
<tr>
<td>InceptionV3</td>
<td>0.21</td>
<td>82MB</td>
<td>0.74</td>
</tr>
<tr>
<td>InceptionResNetV2</td>
<td>0.16</td>
<td>205MB</td>
<td>0.61</td>
</tr>
<tr>
<td>MobileNet</td>
<td>0.13</td>
<td>26MB</td>
<td>0.34</td>
</tr>
<tr>
<td>MobileNetV2</td>
<td>0.14</td>
<td>24MB</td>
<td>0.83</td>
</tr>
<tr>
<td>DenseNet121</td>
<td>0.20</td>
<td>43MB</td>
<td>0.51</td>
</tr>
<tr>
<td>DenseNet169</td>
<td>0.22</td>
<td>67MB</td>
<td>0.66</td>
</tr>
<tr>
<td>DenseNet201</td>
<td>0.25</td>
<td>70MB</td>
<td>0.66</td>
</tr>
<tr>
<td>The Proposed Model</td>
<td>0.044</td>
<td>20.5MB</td>
<td>0.94</td>
</tr>
</tbody>
</table>

the recommended network we constructed is about 0.004 seconds per query image. When compared to the time-consuming and resource-heavy pre-trained models that were previously employed, this approach is much more efficient. Figure 4 shows the similarity results for a sample shoe image when the recommended model and the other models evaluated in this study were used.

6. CONCLUSION

This research aims to summaries our efforts to date in creating an online shopping image retrieval model powered by deep learning. In an effort to improve performance on the shoe image retrieval problem, this study expands on previous work by modifying the network design of InfoGAN and proposing a new network to achieve this goal. The study’s primary objective was productivity enhancement. We evaluate the proposed network’s efficacy by contrasting it to that of several industry-leading shoe image comparison platforms. The results show that when compared to the current state-of-the-art solution, the suggested model performs better and takes less time to execute. We conclude that the suggested model is suitable for use in the design of practical e-commerce systems due to its ability to offer correct and rapid inference conclusions. To better satisfy the needs of our clients, we intend to integrate the strategy with an online storefront.

References


Figure 4. Results comparing the proposed mode and pattern recognition, pages 8110–8119, 2020.


