Content based Image Retrieval System using Combination of Color and Shape Features, and Siamese Neural Network

R Rajkumar, M V Sudhamani

Abstract: With an advent of technology a huge collection of digital images is formed as repositories on world wide web (WWW). The task of searching for similar images in the repository is difficult. In this paper, retrieval of similar images from www is demonstrated with the help of combination of image features as color and shape and then using Siamese neural network which is constructed to the requirement as a novel approach. Here, one-shot learning technique is used to test the Siamese Neural Network model for retrieval performance. Various experiments are conducted with both the methods and results obtained are tabulated. The performance of the system is evaluated with precision parameter and which is found to be high. Also, relative study is made with existing works.

Keywords: CBIR, Siamese Neural Network, One-shot learning, Color.

I. INTRODUCTION

The world wide web (WWW) is a large repository for enormous amount of resources. The digital images are also part of the resources and majorly contributes to the repository on web. This has led to difficulty for search engines to locate the images searched by users on web accurately. Today’s image searching techniques are using combinations of features extracted from the images and are found to be effective in retrieval of similar images. In order to improve the effectiveness in retrieval of similar images on web images with diversified classes [3], there are several deep learning algorithms that have shown steep improvement in retrieving of similar images. In this paper, we have proposed a combination of features and deep learning concept of Siamese Neural Network (SNN) and for evaluation of the retrieved images to be similar or not, One-Shot Learning system is used [4].

The rest of this paper is organized as given here: section II discusses the related works, III briefly explains the working of SNN and its architecture, IV describes the concept of one-shot learning used for evaluating the system, section V shows the experimental results, and VI is conclusions and future enhancements.

II. RELATED WORKS

The combination of features is found to be an effective way to find similar images from the repository with many Content Based Image Retrieval (CBIR) systems. In [6], extract the low-level features of image such as color, shape and texture.

The HSV color histogram, shape descriptors from Hu-moments, and the texture is obtained using Gray-Level Co-occurrence Matrix and combinations are made to retrieve the similar images and is found to be effective. In [7], the HSV color histogram is considered as the global feature and edges and along with Harris corner detector for corner points of an image are considered as combinations and retrieval of objects in an image is done. The experiments have shown effective CBIR system with the combination of features than using individual features. Many recent works [8, 9, 10, 11, 12] have used the convolutional neural network for extraction of features from the images and store them. These methods have shown superior performances than the earlier methods of feature extraction.

In this paper, a CBIR system implemented using both combination of features and deep learning technique. The experiment is carried out on 10,000 web images of 50 categories with 200 images in each and referred as CBIR-50 built using crawler. Fig. 1 shows sample images from each category in the repository.

Fig. 1: Sample images from CBIR-50 dataset

III. TECHNIQUES USED IN THE PAPER

A. Combination of features using color and shape descriptors

In our earlier work [5], the Content Based Image Retrieval (CBIR) system was built that used the combination of image features of color histogram in HSV color space for multi-
segments of an image and the shape features using Hu moment invariants. The first experiment was conducted for Wang’s ground truth dataset of 1000 images for 10 categories where average precision was 84.2%.

Later, the above technique is used with web images of CBIR-50. The features extracted for all images is stored in the features repository. The similarity is measured with chi-square similarity metric.

The similarity score reveals better performance and most similar images are displayed on the user interface.

B. Siamese Neural Network

The retrieval system using Siamese Neural Network with a K N-way one shot learning algorithm is proposed as a novel work using CBIR-50 dataset.

Siamese Neural Network is a similarity prediction model that uses convolution neural network (CNN) for feature extraction and makes prediction using these features as shown in Fig. 2. Here, image x1 and image x2 are given to CNN to compute the features. These features of image x1 and image x2 are referred to as h(x1) and h(x2) respectively. The features decide whether the images belong to the same category or not by finding the difference.

It is a difficult task to perform feature wise or element wise absolute difference, and hence we use a sigmoid activation function which computes similarity score. The sigmoid function generates similarity score between 0 and 1. The lesser the similarity score then both images belong to different category and higher the similarity score then, both images are same category.

The Siamese neural network uses the following convolution neural network (CNN) to perform the feature extraction process for images in the repository of CBIR-50.

![Fig.2: Block diagram for Siamese Neural Network](image)

The kernel is initialized and convolve over the image to transform it and obtain the feature map as given in following equation (1):

\[
G[m,n] = (f \times k)[m,n] = \sum_j \sum_k h[j,k]f[m-j,n-k]
\]

where

- \( f \) denotes the input image,
- \( h \) denotes the kernel or filter,
- \( m, n \) denotes the resultant image size,
- \( i, j \) denotes the filter size

The dimension of the output matrix depends on hyper parameters like stride, padding, number of filters and filter size and can be computed as given in equation (2):

\[
n_{out} = \left\lceil \frac{n_{in}+2p-f+s}{s} + 1 \right\rceil
\]

where

- \( n_{out} \) is the output image as feature map
- \( n_{in} \) is the input image
- \( p \) is the zero-padding size

From equation (2) we obtain the feature map from each stage of convolution layer. The next step is to apply a activation function to introduce non-linearity to the system. In the convolution networks, ReLU and sigmoid activation functions are used. ReLU is a Rectified Linear Units that is used to overcome the stepping function drawback of biasing the network linearly. The equation (3) shows the function used to compute the ReLU value for each pixel considered for input image:

\[
y = \max(0, x)
\]

The ReLU activation function slope for all positive value of x, and zero for other values of x =0 or <0. Another activation function used is sigmoid activation function as given in equation (4), that generates score between 0 and 1, mostly used during binary classification.

\[
S(x) = \frac{1}{1+e^{-x}}
\]
The feature map obtained from the convnets applied on the input image will be down sampled by using maxpooling and reduces the size of the tensor and thereby increases the speed in calculations. The max pool selects the maximum value from the region and put it into the corresponding output, and this max pool depends on filter size and stride used.

The Fig. 3 shows the convolution networks used in our proposed work with input layer of size 128x128x3, where 128x128 is the image size and 3 represents the number of channels used in image representation i.e., red, blue and green, as color images are considered.

The algorithm I shows the generating of training data with batch size being specified as parameter for the generate_training_data() function.

Algorithm I: To generate_training_data(batch_size) – SNN

Step 1: Let N = batch_size
Step 2: Let xtrain = [ ], ytrain = [ ]
    // (xtrain & ytrain – input & labels)
Step 3: Consider N/2 random classes as set1
Step 4: For every class C in set1:
    get 2 random images x and y from C
    Add (x,y) to xtrain, 1 to ytrain
    //similar image pairs
Step 5: Consider N/2 random classes as set2
Step 6: For every class C in set2:
    get 1 random image x from C
    get 1 random image y from random classes-{C}
    Add (x,y) to xtrain, 0 to ytrain
    //dissimilar image pairs
Step 7: Return xtrain, ytrain

In the above algorithm, we observe that the image pairs are considered and trained for 50% with similar images and 50% with dissimilar images, thereby it ensures that the network model doesn’t get biased towards either similar or dissimilar prediction only. The table III shows the use of hyper parameters for training the model on Kaggle cloud withGPU being used. The total number of epochs were 40,000, the batch size for training was 128, and the total time taken for training and extracting features is as shown in table I.

Table – I: Hyper Parameters used in Training of SNN

| Parameter               | Value                          |
|-------------------------|--------------------------------|
| Number of Training Images | 8,000                          |
| Epochs                  | 40,000                         |
| Batch Size              | 128                            |
| Optimizer               | Adam with LR = 0.000009         |
| Loss function           | Binary cross entropy           |
| Time taken on Kaggle cloud - Nvidia Tesla P100 GPU | 205.5 mins (3 Hours 25 mins 30 Secs) |
result with 4 categories of images.

- Similarly, for N category of images, we can have N-way one shot learning technique and obtain the scores to find similar images to the query image.

Fig. 4: Sample of 50 way One – shot learning technique

The Fig. 4 shows a sample for 50 way one shot learning technique and similarly the model is evaluated using the K N-way One-shot learning technique, where the query image is evaluated against set of images consisting of N-random images from different classes, K-times. The algorithm – II shows the steps in implementing CBIR system using SNN.

In our proposed work, we have considered the value of K to be number of images in each category, i.e. 200 images in each category. N – way represents number of the categories in the repository, in our case it is 50. The One-shot algorithm was introduced to evaluate the system for similar or dissimilar images being retrieved.

Algorithm II: CBIR using Siamese Neural Network

Step 1: Input query image X
Step 2: Input K value for number of similar images retrieval
Step 3: Let N = number of categories in repository
Step 4: Call K N-way one shot learning on X
Step 5: For each 1th iteration, save best matched image into result
Step 6: Display images in the result

IV. PERFORMANCE EVALUATION METRIC

The Precision and mean average precision (MAP) are computed using the following equations (5) and (6):

\[
\text{Precision} = \frac{\text{Number of Relevant Images Retrieved}}{\text{Total Number of Images Retrieved}} \tag{5}
\]

\[
\text{MAP} = \frac{\sum_{q=1}^{Q} \text{Precision}(q)}{Q} \tag{6}
\]

where q is the value of precision for a query Q is number of query images

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

A. Experiments conducted using combination of color and shape features

The table II shows the average precision obtained for the experiment conducted on random category of images with random number of samples used as query image in each category.

The table also displays time taken to retrieve top 100 similar images using a machine with Intel® Core™ i5-3230 CPU @ 2.6 GHz and 8 GB RAM on Windows 10 platform and developed using python 3.6.

Table - II: Average Precision and Time with Color + Hu moment

| Sample Categories | Precision (%) | Time taken (secs) |
|-------------------|--------------|------------------|
| Apple             | 88.75        | 18.45            |
| Camera            | 67.50        | 18.50            |
| Chocolate         | 68.75        | 17.97            |
| Desert            | 84.38        | 18.00            |
| Jeans             | 98.75        | 17.82            |
| Lotus             | 71.25        | 17.75            |
| Shark             | 100.00       | 17.95            |
| Sheep             | 81.25        | 17.75            |
| Tabla             | 88.13        | 18.05            |
| Taj Mahal         | 72.50        | 18.15            |
| **Average**       | **82.13**    | **18.04**        |

Experiment is carried out on CBIR-50 dataset and results obtained shows that the performance is found to be good with 82.13% of precision over the random testing of images.
The resultant set are shown in the below Fig. 5 for various query images.

**Fig. 5: Sample retrieved images for various queries**

The performance obtained using the color and shape features can be further improved with the help of Siamese Neural Network as discussed in subsequent section.

The similar retrieved images along with two sample query input images is as shown in Fig. 6.

**Fig. 6: Display of input query image and the resultant set**

In table – III, the precision for all categories of images is shown for randomly chosen images from each category. The average precision obtained is 98.62% and is found to be very good.

The table – IV shows a comparison of the average precision obtained using various experiments and with existing works. It is observed that the proposed work using SNN has performed better by 0.21% with average precision than the existing work [12].
Content based Image Retrieval System using Combination of Color and Shape Features, and Siamese Neural Network

Table-III: Precision for 50 – category of images using Siamese Neural Network

| Category | Precision (%) | Category | Precision (%) | Category | Precision (%) | Category | Precision (%) | Category | Precision (%) |
|----------|---------------|----------|---------------|----------|---------------|----------|---------------|----------|---------------|
| Apple    | 99.18         | Chocolate | 98.85         | India    | 99.33         | Sachin   | 98.18         | Taj Mahal | 99.28        |
| Barack Obama | 99.28       | Clock    | 99.20         | Jeans    | 99.13         | Salman   | 99.30         | Television | 99.28        |
| Bikes    | 99.28         | Coca cola | 98.83         | Kangaroo | 97.88         | Shark    | 97.04         | Tiger     | 98.58        |
| Bottle   | 95.70         | Dairy Milk | 99.40         | Lotus    | 99.18         | Sheep    | 99.18         | Traffic Light | 97.61        |
| Buddha   | 98.23         | Dalmatian | 99.15         | Maggi    | 98.98         | Ship     | 98.80         | Vase      | 96.99        |
| Butterfly | 99.35        | Desert   | 99.28         | Mango    | 97.31         | Shirt    | 98.11         | Vulture   | 98.31        |
| Camera   | 97.24         | Eiffel Tower | 98.78        | Mobile   | 98.36         | Shoes    | 99.30         | Waterfall | 99.20        |
| Car      | 99.20         | Elephant | 98.73         | Narendra Modi | 99.18    | Soccer Ball | 98.03        | Watermelon | 99.05        |
| Cat      | 98.95         | Helmet   | 99.25         | Peacock  | 99.28         | Sunflower | 99.10         | Wine      | 95.10        |
| Chair    | 98.90         | Horse    | 97.91         | Rose     | 98.93         | Tabla    | 99.40         | Zebra     | 99.03        |

Mean Average Precision = 98.62%

Table -IV: Precisions compared with existing works

| Experiment conducted with Features | Dataset | Precision (%) |
|------------------------------------|---------|---------------|
| Color in YCtBR, Color Moments [13]  | Wang’s Dataset with 1K images | 82.70 |
| HSV Color histogram and Hu moment invariants [5] | Wang’s Dataset with 1K images | 84.20 |
| HSV Color histogram and Hu moment invariants [Proposed] | CBIR-50 with web images | 82.13 |
| Color, Shape and Texture [12] | Dataset with 10K web images | 98.41 |
| Siamese Neural Network + One shot Learning Technique – [Proposed] | CBIR-50 with 10K web images | 98.62 |

The table - V shows the time taken for retrieving top 100 images for random query images given as input on various machines as mentioned in the table. It is found that the Kaggle machine on cloud with GPU takes about 5.49 seconds to retrieve top 100 images from the CBIR-50 dataset using SNN and machine with 3rd generation Intel i5 CPU processor with 2.6 GHz speed needs about 883.15 seconds.

Table - V: Time taken for top 100 images retrieval on different machines

| System Specifications | Time Taken (seconds) |
|-----------------------|----------------------|
| Kaggle Cloud – Nvidia Tesla P100 GPU | 5.49 |
| Nvidia GeForce GTX 1060 max-Q (6GB – GDDR5) GPU (CUDA 10, Tensorflow 1.12, Keras 2.23) | 18.45 |
| Intel Core ™ i7 – 7700 HQ CPU 2.8 GHz (7th Gen) | 396.11 |
| Intel Core ™ i5 – 4570 CPU 3.2 GHz (4th Gen) | 434.19 |
| Intel Core ™ i5 – 3230M CPU 2.6 GHz (3rd Gen) | 883.15 |

VI. CONCLUSIONS AND SCOPE FOR FUTURE WORK

This paper proposed a retrieval based on combination of features and SNN. The experiments were carried out on the CBIR-50 dataset. The average precision value was 82.12%. In order to enhance the precision value from the earlier method, a deep learning approach, SNN was adopted. The result obtained for average precision for all the categories is 98.62% which is more than the existing works. The time taken for the top 100 similar images on Kaggle is also recorded. The Siamese neural network is seen to give high precision results. But in case of new category of images gets added, then the model needs to be retrained along with new category and this consumes time for retraining. There is a scope to improve the network such that it can retrieve the new category of image by extracting features and also avoid the retraining phase.

REFERENCES

1. Bromley, Jane Guyon, Isabelle LeCun, Yann Säckinger, Eduard Shah, Roopak, Signature verification using a "siamese" time delay neural network, Advances in Neural Information Processing Systems 6: 737–744, 1994.
2. Bharath Hariharan, Ross Girshick, Low-Shot Visual Recognition by Shrinking and Hallucinating Features, The IEEE International Conference on Computer Vision (ICCV), pp. 3018-3027, 2017.
3. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, Li Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database, 2009 IEEE Conference on Computer Vision and Pattern Recognition, https://doi.org/10.1109/CVPR.2009.5206848, 2009.
4. Gregory Koch, Richard Zeme, Ruslan Salakhutdinov, Siamese Neural Networks for One-shot Image Recognition, Proceedings of the 32nd International Conference on Machine Learning, Lille, France, JMLR: W&CP volume 37, 2015.

Published By: Blue Eyes Intelligence Engineering & Sciences Publication
5. R Rajkumar, Dr. M V Sudhamani, Development of Retrieval System using Color from Multi-Segments and Shape features from Images as Combination, Accepted to International Journal of Engineering and Technology (IJEAT) – Open Access Journal, DOI: 10.35940/ijeat.A1291.109111, ISSN: 2249 – 8958, Volume-9 Issue-1, October 2019.

6. Ramesh K Lingadalli, N.Ramesh, Content Based Image Retrieval using Color, Shape and Texture, International Advanced Research Journal in Science, Engineering and Technology, Vol. 2, Issue 6, June 2015.

7. H. Kavitha, M.V. Sudhamani, Object Based Image Retrieval from Database Using Combined Features, IEEE Fifth International Conference on Signal and Image Processing,DOI: https://10.1109/ICSIP.2014.31, 2014.

8. Alexey Potapov, Innokentii Zhdanov, Oleg Scherbakov, Nikolai Skorobogatko, Hugo Lapatie, Enzo Fenoglio, Semantic Image Retrieval by Uniting Deep Neural Networks and Cognitive Architectures,11th conference on Artificial General Intelligence, AGI’18, 2018.

9. Retrieval, Ong EJ, Husain S, Bober M. Siamese network of deep fisher-vector descriptors for image retrieval. arXiv preprint arXiv:1702.00338, 2017.

10. Wang, H., Cai, Y., Zhang, Y., Pan, H., Lv, W., Han, H.: Deep learning for image retrieval: what works and what doesn’t. In: IEEE International Conference on Data Mining Workshop (ICDMW), pp. 1576–1583, 2015.

11. Ahmad Alzu’bi, Abbes Amira, Naeem Ramzan, Content-Based Image Retrieval with Compact Deep Convolutional Features, Neurocomputing (2017), doi: 10.1016/j.neucom.2017.03.072, 2017.

12. Oudida Mohamed, El Asnaoui Khalid, Ouanan Mohammed, and AksasseBrahim, Content-Based Image Retrieval Using Convolutional Neural Networks, RTIS 2017, AISC 756, pp. 463–476, https://doi.org/10.1007/978-3-319-91337-7_41, 2019.

13. YatiDandotiya, Anshul Atre, “Image Retrieval using Edge Detection, RLBP, Color Moment Method for YCbCr and HSV Color Space” IEEE International Conference on Electronics, Communication and Aerospace Technology, pp. 662 – 668, 2017.

14. Shrinivasacharya P., Sudhamani M.V. (2013), “WWW based Image Retrieval using Color, Texture and Shape Features”, International Journal of Applied Engineering Research, ISSN 0973-4562 Vol. 10 No.86, 2015.

AUTHORS PROFILE

Mr. R Rajkumar is presently working as Assistant Professor, Dept. of ISE, RNSIT, Bengaluru. has completed B.E and M.Tech degrees in 2001 and 2007 respectively. He has 16 years in teaching, 7 years in research and 2 years industrial experiences. His research area is Content based image retrieval using features and annotations.

Dr. M V Sudhamani, currently working as Dean-R&D, Professor and HoD, Dept. of ISE, RNSIT. She is having Teaching, Research and Industrial experience of 25 years. She has specialization in Image Processing, Content-based Image Retrieval, Advanced Algorithms and Databases. Guided and guiding candidates for Ph. D degree. She has carried out two research projects from VTU and AICTE. She has served as member of Board of Examiners (BOE) and Board of Study (BOS) member in VTU and other autonomous institutions across India. She has organized two international conferences ICDECS 2011 and 2015, and one more in December 2019.