Published by IASE

Annals of Electrical and Electronic Engineering

Journal homepage: www.aeeej.org

Trademark image recognition utilizing deep convolutional neural network

M. Haddad *, A. Saleh, Y. Jaber

Department of Electrical Engineering, School of Engineering, The University of Jordan, Amman, Jordan

ARTICLE INFO

Article history:
Received 12 October 2018
Received in revised form
2 February 2019
Accepted 15 February 2019

Keywords:
Content-based image retrieval
Deep convolutional neural networks
Particle swarm optimization
Self-organizing map

ABSTRACT

The purpose of this work is taking into account major challenges of trademark recognition i.e., reduction in semantic gap, attaining more accuracy, reduction in computation complexity by implementing trademark image retrieval through deep Convolutional Neural Networks (CNNs) integrated with relevance feedback mechanism. The dataset features are optimized through Particle swarm optimization (PSO); reducing the search space. These best/optimized features are given to the self-organizing map (SOM) for clustering at the preprocessing stage, The CNN model is trained on feature representations of relevant and irrelevant images, using the feedback information from the user bringing the marked relevant images closer to the query. Experimentation proves significant performance by using FlickrLogos-32 PLUS dataset, as illustrated within the performance results section.

1. Introduction

The automatic trademark retrieval system evaluates the query trademark among the database of trademark images for similarity, for approving the trademark of the corporation/organization. Majority of the CBIR applications based upon retrieval demand the image to search, to match and to retrieve. Conversion of the image's high level semantics inferred by individuals to low level semantics illustration is the key challenge of the work in this area; giving rise to the possibility of research in this work.

Feature vector's dataset is constructed by implementing feature extraction to deduce shape, color and texture representation for images of trademark. This database of features may include a few irrelevant and superfluous features, unnecessary for retrieval process. Hence, Particle Swarm Optimization (PSO) technique is employed to optimize this feature set; narrowing the search space of the retrieval process.

The optimization practice, Particle Swarm Optimization (PSO) exploits a parallel exploration of various points which are adjusted in the search room. PSO has the advantage of fast convergence while judged against other optimization practices, for example: global optimization algorithms and genetic algorithm (Kameyama et al., 2006). The self-organizing map (SOM) is trained upon this best dataset features. This training entails the input vectors plotting next to all others in close by plot elements (Laaksonen et al., 2001). The SOM arranges the optimized feature set in the form of clusters. The information obtained through the user's feedback i e. Relevance Feedback is used in the query modification process (Su et al., 2011).

The proposed work is intended to the improvement of RF performance by utilizing the deep CNNs for trademark image recognition and retrieval. The proposed scheme is to employ the capability of a deep CNN to revise its inner organization to provide the improved representations of the images applied for the improvement in retrieval process by using the user's feedback. The feature extraction is executed through deepest neural layers of the convolutional neural network, in order that the feature illustrations of the relevant images come nearer to the query and the not relevant images deviates as of the query image.

The suggested framework is a combination of the following two approaches, applied for trademark image retrieval: 1) The novel RF approach based on query modification and mining navigation patterns 2) The influence of deep CNN in the RF approach

The rest of the paper is organized in the following sections. Section 2 explains the novelty and contributions of the suggested framework. Literature regarding the related contributions in the area is described in Section 3. The proposed framework is depicted in section 4. Experimentation results are presented in section 5 and in section 6 conclusions are drawn.

2. Novelty and contributions of the suggested framework

Successful relevance feedback (RF) methodologies suggested earlier are combined with the learning approaches like Decision Trees, Support Vector Machines and boosting techniques. The proposed work is utilizing the power of deep CNN for improving the performance of RF; implemented through the following two phases:

1) Preprocessing Phase: Optimization of features dataset using PSO and clustering using SOM.
2) Retrieval Phase: Query modification and navigation pattern mining based RF is combined with Deep CNN, for trademark recognition and retrieval.

Feedback information records are retained to deduce the patterns of navigation for the subsequent query stages. This feedback information from the user is used to train the deep CNN model regarding relevant and irrelevant images. Due to the preprocessing phase there is reduction in the complexity of

* *Corresponding Author.*
Email Address: MHaddad@ju.edu.jo *(M. Haddad)*
https://doi.org/10.21833/AEEE.2019.03.003

computation of the suggested framework. The combination of deep CNN with RF proves improved retrieval results as shown in the results section.

3. Related contributions in the area

Kameyama et al. (2006) proposed a method by employing Particle Swarm Optimization (PSO) for optimization of the factors incorporated in the CBIR system's evaluation algorithm for relevance, by optimization in accordance to the results suitability. Laaksonen et al. (2001) used PicSOM; self-organizing maps (SOMs) like a relevance method used to Content-Based Image Retrieval (CBIR). Wang and Hong (2012) combined the image's global and local features for retrieval of trademark images. On the basis of similarity the Zernike Moments were sorted. The similarity metric used was Scale invariant feature transform (SIFT) features. Bagheri et al. (2013) suggested a logo recognition system by using shape features by describing them through Zernike moments and Fourier descriptors. Dempster-Shafer theory combination approach was employed for image recognition. Alaei and Delalandre (2014) designed document image's logo recognition framework by image's region of interest detection. For this region detection piece-wise painting algorithm (PPA) and probability features in conjunction through a decision tree method was employed. Suganthan (2002) executed indexing on shape scheme through self-organizing map. The geometric shape's structural information was extracted through pair wise relational attribute vectors. Two SOMs; SOM1 and SOM2 were implemented in the work. SOM1 having relational attribute vector's global histogram were fed as input to SOM2. The shape properties of the objects were defined by SOM1, SOM2 created topology preserving mapping for the structural shapes. Vesanto et al. (1999) employed the Self-Organizing Map (SOM Toolbox) as a quantization of vector by arranging the example vectors on top of a standard framework which was low-dimensional in an arranged way. Okayama et al. (2007) suggested two methods for retrieval optimization of content-based image retrieval (CBIR). The 1st one was a map generation by using information from feedback through supervised training. While in the second one the fine-matching relaxation action factors were optimized through Particle Swarm Optimization. Ma et al. (2012) designed a mixed scheme for image clustering and retrieval, utilizing Particle Swarm Optimization (PSO) and Support Vector Machine (SVM). Xue et al. (2013) presented two PSO-based multi-objective algorithms for selecting the features. The first was based upon the thought of non-controlled arrangement in PSO to tackle troubles while selecting the features. The proposals for crowding, mutation, and dominance to PSO for searching the Pareto front solutions were included within the 2nd algorithm. Romberg et al. (2011) proposed a structure for recognition of logos by quantizing illustration of the logo regions, determined through the spatial structures arrangement and local features. The experiments were assessed using Flickrlogos32 database. Revaud et al. (2012) learned a statistical form for the distribution of wrong detections produced by an image matching algorithm. The datasets used for testing were BelgaLogos and FlickrLogos dataset. In Romberg and Lienhart (2013); feature bundling was employed for logo recognition and retrieval. The bundles were formed by combing spatial neighborhood features and the local features. The experimentation was done using FlickrLogos-32 dataset. Recent works of Bianco et al. (2015), Eggert et al. (2015), Hoi et al. (2015), Iandola et al. (2015), and Bao et al. (2016) surveyed in the area have employed deep learning approach for trademark retrieval. Bianco et al. (2015) and Eggert et al. (2015) evaluated the structure for retrieval of logos through pre trained Convolutional Neural Networks (CNN) and the data which was synthetically produced. Hoi et al. (2015) utilized deep learning mechanism for recognition of logos. The object detecting method was experimented through deep region-based convolutional networks. Iandola et al. (2015) experimented logo recognition through deep convolutional neural networks (DCNNs). Various DCNN architectures were presented and examined for resulst using the FlickrLogos-32 database. Bao et al. (2016) explored the appropriate R-CNN's plan and situation for logo image recognition after examining the system using FlickrLogos-32 database. Tzelepi and Tefas (2016) utilized the influence of deep Convolutional Neural Networks (CNNs) with Relevance Feedback for retrieval of images. They utilized the feedback information from the user in RF to refine the deeper layer's feature representation of CNN. The pre trained model was retrained for the relevant and irrelevant features from the feedback, resulting in improved retrieval results by concerning the requirement of the user.

Taking into account the advantage of multiple features based retrieval system, the proposed work is developed by implementing the feature extraction for color, texture and shape. The work proposed in this area by many researchers is based upon the class of trademark images like merely an image, a text or both. The proposed work is handling all the classes of trademark images. The effective techniques found for optimization and learning is PSO and SOM respectively; employed at preprocessing phase. As discussed above, when only deep learning techniques employed for trademark image retrieval proves good retrieval results. The influence of Deep CNN is combined with RF, at the retrieval phase in the proposed framework.

4. Proposed framework

The proposed framework is implemented in the subsequent steps:

- Feature vectors database generation
- Feature set optimization
- Clusters creation through SOM
- Input query image
- Retrieve the relevant images based on similarity
- Take user's feedback (RF) upon relevant/ irrelevant images
- Use this feedback information to train Deep CNN
- Retrieve the refined results as final relevant images

4.1. Feature vector database generation

FlickrLogos-32 PLUS Dataset is used for implementing experiments. Details are depicted in Table 1. The openly accessible FlickrLogos-32 database includes images which are non-trademark images. The FlickrLogos-32 PLUS Dataset is generated by eliminating these non-trademark and including a number of trademark images.

Table 1
FlickrLogos-32 PLUS trademark database.

Database	Total No. Images	Trademark category
Publicly accessible: FlickrLogos-32 Database	8240	32
Generated: FlickrLogos-32 PLUS Database	8550	38

Fig. 1 shows example images of trademarks utilized in the implementation commencing FlickrLogos-32 PLUS database. The low level features of an image such as shape, color and texture, express the visual content of the image (Feng et al., 2013). Shape, color and texture feature extraction are implemented to retain the database feature's vector of the trademark images. Circularity and Fourier Descriptor methods are executed for shape, Color Moments, Color Histogram and Color Correlogram practices are employed to express color, Haar Wavelet and Gabor wavelet are used for texture.

Fig. 1: Example trademarks from FlickrLogos-32 PLUS database.

4.2. Feature set optimization

The optimized feature set is obtained by executing Particle swarm optimization (PSO) method; given to self-organizing map (SOM) for training. The retrieval of images schemes includes various features of database depiction comprising unnecessary features not functional for the retrieval process. Hence, optimization of features is needed to eliminate these unnecessary features from the database features. The search room contains the particle's present coordinate, in the structure of optimization search. The best solution is acknowledged by retaining the trace of every particle's coordinates within the crisis area. This value is referred as P_{best}. The top value attained by any particle amongst the entire particles in the area, known as Q_{best}. The k^{th} particle can be represented as $C_k = (C_{k1}, C_{k2, \dots, kn})$ and the velocity i e, particle's rate of progress is designated as $T_k = (T_{k1}, T_{k2, \dots,} T_{kn})$ particle alters its positions and velocity through subsequent equations:

$$T_k(v+1) = T_k(v) + d_1 * a_1\big(\mathrm{P_{best}} - C_k(v)\big) + d_2 * \ a_2(Q_{best} - C_k(v)) \quad (1)$$

$$C_k(v+1) = C_k(v) + C_k(v) \quad (2)$$

where T_k (v) and T_k (v+1) denotes velocity of k^{th} particle on iteration v and v+1 respectively; C_k (v) and C_k (v+1) denotes the k^{th} particle position on iteration v and v+1 respectively; d1 and d2 depicts the cognitive acceleration coefficient and social acceleration coefficient; a_1 and a_2 being random number ranges between (0,1).

PSO is transformative computational method showing swarm intelligence and optimizing feature set; however does not ensure optimization (Bai, 2010); consequently the clustering of these features is executed utilizing SOM. The relevance feedback is implemented by using some clusters near to the query identified by averaging their features.

4.3. Cluster creation through self-organizing map

The planning of data set as input, S^n on top of a typical two-dimensional nodes array is defined by SOM. The parametric reference vector $n_k \in S^n$ is linked with each node k in the plan.

The proposed work has projected the nodes array onto a rectangular pattern. The input is mapped by evaluating every input vector v against the n_k: P and the top correspondent is determined. Euclidean distance is used for comparing each input vector $v \in S^n$ with the n_k: P the winning node e is computed as follows:

$$\|v - n_e\| = \min\{\| v - n_k\|\} \quad (3)$$

v is mapped through e, comparative to the component values n_k. Topographically closer nodes within the array exhibit learning through the similar input. The updation formula is:

$$n_k\ (t+1) = n_k\ (t) + h_{e,k}(t)[v(t) - n_k\ (t)] \quad (4)$$

where the discrete-time coordinate is t and the region defining function $h_{e,k}$. The random values are used for the $n_k{:}\,P$ in the early phase.

4.4. Relevance feedback

The query modification based relevance feedback (RF) is used in the proposed work and integrated with deep CNN. Query modification is hybrid through Improved Query Point (IQP), Query Expansion (QE) and Query Modification (QM) method. The testing is done by executing 10 number of RF iterations. The positive images query points of every iteration are clustered by employing k-means clustering algorithm. Each cluster is allotted with a number. The feedback information from the user is maintained as log records in the following tables and the patterns found repeatedly are mined in the next query.

I. Initial log Table – Contains query image, relevant images, query point and iteration number
II. Query Point Table – Contains query image and query point
III. Navigation Action Table – Contains query image, cluster number and iteration number. This table is constructed as following:

a) Using the results of retrieval for each iteration, the query points clustering is done through k-means algorithm and allocated a number.
b) The cluster number and iteration number are stored in this table.
c) A pattern for every query is constructed using these clusters. For instance, for a query image, the retrieval results images are existing as, 1st iteration: cluster 3, 2nd iteration: cluster 2 and 3rd iteration: cluster 1, then the pattern is N31, N22, and N13. The mining is done using these sequential patterns through Apriori algorithm for mining.

IV. Log division Table – Encloses Query point and relevant image

The Initial Log Table holds the information related to every feedback after each RF iteration. The navigation patterns are mined using Navigation Action Table. The relevant images are searched through Query Point Table and Log division Table. The sequential patterns obtained are used for constructing the navigation pattern tree. The sequential pattern is depicted through every branch of this tree. The query for the entire chronological patterns is exploited as seed of the tree called as query kernel. The following steps describe the search process:

1. The modified query point is obtained by averaging the features of images marked as positive.
2. The nearest query kernels are obtained to acquire the similar chronological patterns.
3. These similar trees of navigation pattern are utilized to get the neighboring leaf nodes.
4. The query points denoted by 't' which are relevant are obtained from the collection of the nearest leaf nodes.
5. The 'n' significant images are obtained.

The suggested method implements step 1 for generating Improved Query Point (IQP). Steps '2-5' are executed for Query Expansion (QE). The Query Modification (QM) process is implemented by obtaining new feature weights utilizing the positive images features got from each feedback.

4.4.1. Improved query point (IQP)

old_{qpt} denotes the query point of the images in the earlier feedback. The positive images features I are averaged to get the new query point new_{qpt}. If I ={i_1, i_2, ..., i_k } describes positive images and the j^{th} feature's dimensions, D_j ={ d_1^N, d_2^N, ..., d_m^N} got through the N^{th} positive image. Therefore the modified query point new_{qpt} designated by I is described as (Su et al., 2011):

$$new_{qpt} = \{\overline{D_1,D_2}, \dots, \overline{D_c}\} \quad \text{where } 1<=j<=c \tag{5}$$

$$\overline{D_j} = \{\overline{d_1,d_2}, \dots, \overline{d_m}\} \text{ and } \overline{d_t} = \frac{\sum 1\leq n\leq s, d_t^n \in D_j\, d_t^n}{s}$$

4.4.2. Query modification (QM)

Set of positive images: I= {i_1, i_2, ..., i_k} obtained through the previous query point old_{qpt} during the former feedback. The modified weight for the k^{th} feature D_k is calculated as (Su et al., 2011):

$$N_k = \frac{\frac{\sum_{x=1}^{a} \alpha_x}{\alpha_k}}{\sum_{y=1}^{a} \frac{\sum_{x=1}^{a} \alpha_x}{\alpha_y}} \tag{6}$$

where

$$\alpha = \sum_{z=1}^{m} \frac{\sqrt{\sum_{i=1}^{b} (d_i^z - d_i^{\text{oldqpt}})^2}}{b} \tag{7}$$

and

1<=k<=a

4.4.3. Query expansion (QE)

The QE method is used to perform the weighted KNN search. The nearest query kernel to each of I is deduced, named as positive query kernel, and the nearest query kernel for every negative image (NI), named as negative query kernel. Few query kernels may be common in query kernels for positive and negative collections. Each kernel is allotted with a token pn.chk to deal with such condition. pn.chk =0 if negative examples number is greater than the positive examples else will be equal to one.

The relevant query pits are deduced. Set of similar leaf nodes are obtained through the navigation pattern tree. The relevant images are searched using the modified weights of the features as computed in equation number 6. The search procedure is categorized into two stages: 1) Generation of the relevant query points 2) Determination of images which are relevant

The related query points are obtained through query point table. Step 1 determines the 't' similar query points. These 't' query points are searched in the log division table to acquire the relevant images. KNN search method is employed to retrieve the 'n' images which are nearer to new_{qpt}.

4.5. Deep CNN in relevance feedback

This paper proposes a CBIR framework for trademark images by integrating deep CNN with a relevance feedback mechanism. The relevant images from log records maintained in the above tables are given as input to the Deep CNN. The CaffeNet model is utilized; an execution of the AlexNet model for experimentation. The model includes 8 neural network layers which are trained; the initial 5 are convolutional and the rest 3 are fully connected. Max-pooling layers trail the 1st, 2nd and 5th convolutional layers, whereas the ReLU non-linearity (f(y) =max (0; y)) is employed to each fully connected and convolutional layer, excluding the last layer indicated by FC8. The seventh neural network layer representations indicated by FC7 are extracted through the CaffeNet model. The CaffeNet model parameters are used to initialize the entire layers of the latest model, excluding the FC8; substituted by a modified classification layer depicting the given dataset labels. There are no changes up to the convolutional layers which are initial fully connected layer indicated by FC6; FC7 and FC8 layers are updated using error back propagation. The feature representations are extracted through the FC7 layer's activations having features dimension of 4096.

In the first RF iteration, the user gives feedback as relevant or irrelevant images to the system.

This feedback information is utilized to train the deep CNN model regarding the feature representation of relevant and irrelevant images. Subsequently, the feedback information in the next RF iterations is utilized in the framework and the CNN model weights are updated, with the intention that relevant images move towards the query depiction whereas the irrelevant ones deviates from the same. The representations acquired through a CNN model for a collection of images as input are modifiable by changing the model weights. The FC7 parameters are trained on positive/relevant and negative/ irrelevant images. The regression task during this training is executed through euclidean loss. The query modification in RF is thus combined with Deep CNN for training upon relevant and irrelevant images.

Let the feature illustration for query image appeared in FC7 layer is: $s \in P^{4096\times1}$

$Y^+ = \{Y_p \in P^{4096\times1}, p = 1, \dots, M\}$ is the feature representations set of M images marked as relevant by the user in the feedback. $Y^- = \{Y_q \in P^{4096\times1}, q = 1, \dots, N\}$ Denotes the feature representations set of N irrelevant images marked by the user.

These relevant and irrelevant illustrations of the images are modified by using retraining capabilities of the neural network. The following equations are utilized to deduce the modified target representations of the positive/relevant and negative/irrelevant images.

$$\min_{y_p \in y^+} \tau^+ = \min_{y_p \in y^+} \sum_{p=1}^{M} \|y_p - s\|_2^2 \tag{8}$$

$$\max_{y_q \in y^-} \tau^- = \max_{y_q \in y^-} \sum_{q=1}^{N} \|y_q - s\|_2^2 \tag{9}$$

The above optimization problems are solved through gradient descent. The objective function's first order gradients, τ^+ and τ^- are calculated as following:

$$\begin{aligned} \frac{\partial \tau^+}{\partial y_p} &= \frac{\partial}{\partial y_p}\left(\sum_{p=1}^{M} \|y_p - s\|_2^2\right) \\ &= \frac{\partial}{\partial y_p}\left((y_p - s)^\tau (y_p - s)\right) \\ &= 2(y_p - s) \end{aligned} \tag{10}$$

and

$$\begin{aligned} \frac{\partial \tau^-}{\partial y_q} &= \frac{\partial}{\partial y_q}\left(\sum_{q=1}^{N} \|y_q - s\|_2^2\right) \\ &= \frac{\partial}{\partial y_q}\left((y_q - s)^\tau (y_q - s)\right) \\ &= 2(y_q - s) \end{aligned} \tag{11}$$

As a result, the update rules for the n^{th} iteration are devised as:

$$y_p^{(m+1)} = y_p^{(m)} - 2\beta\left(y_p^{(m)} - s\right), y_p \in y^+ \tag{12}$$

$$y_q^{(m+1)} = y_q^{(m)} + 2\beta\left(y_q^{(m)} - s\right), y_q \in y^- \tag{13}$$

The preferred distance from the query representation is monitored by the parameter $\beta \in [0,0.5]$.

The above feature representations are used for the n+1 iteration as targets within FC7 layer. The neural network and the deep CNN regression task is devised and further given training through back propagation. Resulting in production for the positive/relevant and negative/irrelevant images representations nearer to the targets $y^{(m+1)}$. Hence the positive/relevant images come nearer to the query image within the FC7 layer and the negative/irrelevant ones move far as of the query. Accordingly, the RF feedback is incorporated by inputting the query and the feedback information of the images to the CNN input layer and getting the modified representations for FC7. This procedure is executed for every RF iteration; the CNN model is initialized with the preceding iteration's parameters and re-trained for the modified set of positive/relevant and negative/irrelevant images (Tzelepi and Tefas, 2016). The above process is executed by the system from the start for every new query.

Algorithm Algorithmic steps for the suggested framework can be given as following:

Step 1: *Feature Extraction implementation:*

i) For Color:

a) Color Correlogram
b) Color Histogram
c) Color Moments

ii) For Texture:

a) Haar Wavelet
b) Gabor Wavelet

iii) For Shape:

a) Circularity Feature
b) Fourier Descriptor

Step 2: *Applying optimization Algorithm: Particle swarm optimization (PSO)*

Determination of optimized features database through PSO

Step 3: *Implementing SOM*

SOM is trained on optimized features database

Step 4: *Selecting image as a query*
Step 5: *Extracting query image features and provide as Input*

Feature extraction of the query image is implemented
Query image is given as an input to framework

Step 6: *Retrieving images* using the similarity/distance

Recognize images relevant/similar to the query image from the database

Step 7: *Taking user feedback*

Relevant? Or not relevant Image?
If yes
Marked Image as relevant/positive
Else
Marked Image as irrelevant/ negative

Step 8: *Obtain modified query point (new_{qpt}) and retain the feedback information in the log*
Step 9: *Input the feedback information to Deep CNN for training*
Step 10: *Final retrieval results*

Final similar/relevant images are retrieved as output

5. Experimentation results

5.1. Implementation setup

The dataset used for testing the framework is FlickrLogos-32 PLUS Database. The optimization of the database features from overall 5034 number of features to the best 2519 number of features is done through particle swarm optimization algorithm. These optimized features are fed to SOM for learning. The classes/clusters are formed by SOM using these optimized features. The top three most relevant clusters are recognized by the system against a given query by averaging the images features within the cluster and the Euclidean distance between them. These three clusters are searched for relevant images.

Firstly 20 relevant/similar images are returned by the system based on similarity. For every query, 10 RF iterations are executed. For every RF iteration, user feedback is taken upon these 20 retrieved images as relevant or irrelevant. The proposed work has used 10 as relevant and 10 as irrelevant images to refine the model. The value of parameter β is set as 0.2 and the model training is done for 1 epoch for every RF iteration. Final output images retrieved are the most relevant 9 images.

5.2. Analysis of time complexity

The PSO algorithm's time complexity is O $(2nd+n^2+2nd)$ here n represents the particles number and search space's dimension is d. For SOM algorithm it is given by O (s^2); where s denotes the input sample size. The RF algorithm possesses the time and memory cost linear by the dimension of the total features (Su et al., 2003). Therefore it's time complexity is calculated as:

$$\sum_{a=1}^{m} \mathrm{O}(\text{ number of total images } * \mathrm{p_a}) = \mathrm{O}(\text{number of total images}) * \mathrm{P}$$

Here P $=\sum_{a=1}^{m} p_a$; where p_a represents the a^{th} feature's dimension.

All the convolutional layers in deep CNN have time complexity as:

$$O(\sum_{i=1}^{n} m_{i-1}\ p_i^2 m_i n_i^2\ ).$$

Where i represents convolutional layer's index, and n describes the depth i e. convolutional layers number. The filters number as well recognized as 'width' in the i^{th} layer is m_i. Whereas m_{i-1} represents the input channels number of the i^{th} layer. The spatial size i e. the filter's length is p_i. The spatial size of the output feature map is denoted by n_i. This time complexity is applicable at both time i.e., training and testing time, but through a different range. The training time required per image is approximately three times (one for forward and two for backward propagation respectively) of the time needed for testing per image. The above formula does not account the time cost of fc layers and pooling layers; which acquires 5-10% computational time.

The overall time complexity of the suggested algorithm can be given as: T = T1+T2+T3+T4

Where T1, T2, T3 and T4 are time complexity of PSO, SOM, RF and deep CNN algorithm respectively. T1 and T2 cost incurs at one time in the preprocessing stage where as T3 and T4 costs at each query stage.

5.3. Performance results in terms of retrieval

The final retrieved images against a given query using the influence of deep CNN are depicted in Table 2, Table 3 and Table 4 respectively on basis of the similarities of color, texture and shape.

Table 2
Color similarity based output retrieved images.

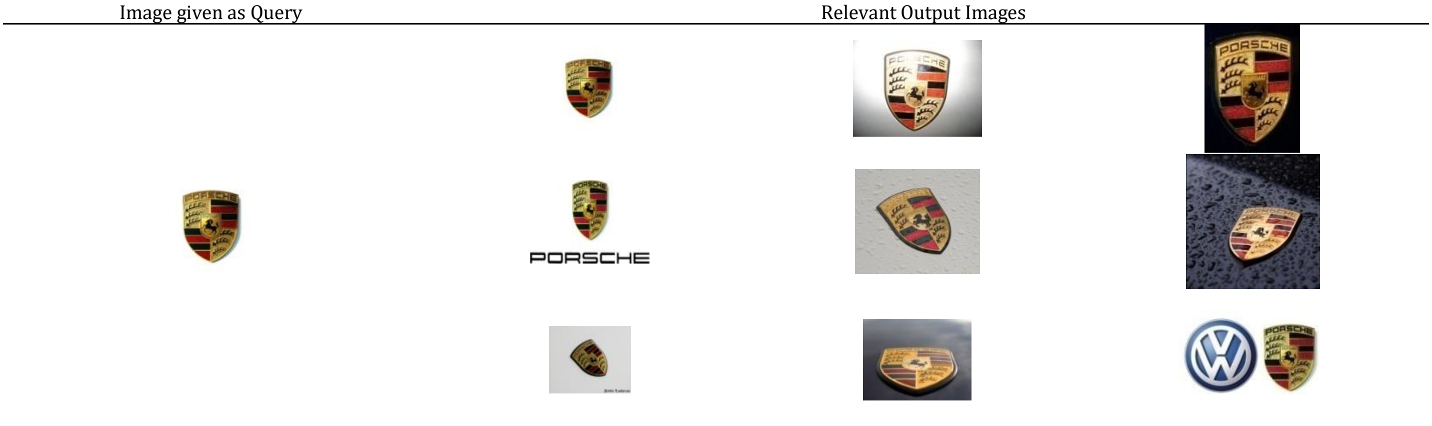
Image given as Query	Relevant Output Images

Table 3
Texture similarity based output retrieved images.

Image given as Query	Relevant Output Images

Table 4
Shape similarity based output retrieved images.

Image given as Query	Relevant Output Images



The metrics used for evaluation of the framework are precision, recall and accuracy, calculated as below:

1.Precision = TP / (TP+FP)
2.Recall = TP / (TP+FN)
3.Accuracy = (TP + TN) / (TP + TN + FP + FN)

Here, TP is True Positive, TN is True Negative, FP is False Positive and FN is False Negative.

The retrieval results for 10 example images as query are reported in Table 5. Table 6 describes the similarity (%) for color, texture and shape for the topmost output images retrieved. In Table 7 and Table 8 the suggested framework is compared with Su et al. (2011) and recent deep learning based approaches proposed by Iandola et al. (2015) and Bao et al. (2015).

Table 7 depicts the improved retrieval results of the method when the Deep CNN is integrated with RF. The optimization (PSO) and clustering techniques (SOM) applied earlier in the preprocessing phase has reduced the space for searching and therefore the complexity.

The comparison in Table 8 proves the improved retrieval results with the integration of deep CNN in RF when compared with the other two, only deep learning based models.

6. Conclusion

In the proposed trademark retrieval system the Deep CNN model is combined with query modification based RF technique at the retrieval phase. In the first iteration of RF, the feedback information from the user is given as input to the deep CNN and trained upon the relevant and irrelevant features representations. In the successive RF iterations these feature representations are refined in the CNN model. This has improved the retrieval performance because of the knowledge of user's information need and bringing the relevant images nearer to the query and moving irrelevant far from it. The preprocessing phase has reduced the search space thereby the time needed for execution significantly due to the optimization (PSO) and clustering (SOM) techniques employed. Future research work direction is replacing the SOM algorithm used for clustering at preprocessing phase by deep CNN.

Table 5
Retrieval results for sample 10 query images.

SNO	QUERY IMAGE	PRECISION	RECALL	ACCURACY (%)
1		0.9751	0.9221	93.58
2		0.9745	0.9481	93.82
3		0.9674	0.9187	93.86
4		0.98	0.9401	94.59
5		0.9643	0.9463	94.77
6		0.981	0.9429	93.71
7		0.945	0.9178	92.97
8		0.979	0.9321	94.81
9		0.982	0.9486	94.76
10		0.988	0.9795	94.54

Table 6
Similarity (%) of color, texture and shape for the topmost nine relevant output images.

S. No	Relevant output Image	Similarity (%) of Color	Similarity (%) of Texture	Similarity (%) of Shape
1		89.91	87.98	92.11
2		92.284	91.033	93.654
3		94.605	93.126	95.12
4		98.769	97.012	98.561
5		95.721	96.117	87.274
6		78.214	85.813	88.891
7		87.661	82.929	85.721
8		78.939	84.862	74.632
9		75.127	84.321	68.519

Table 7
Suggested framework's comparison with Su et al. (2011).

S No.	Methodology	Database	Parameter	
			Precision	Recall
1	Su et al. (2011)	7 classes of different categories of 200 images each (1400 Images)	0.910	----
2	Framework by employing just RF	FlickrLogos-32 PLUS Database	0.935	0.898
3	Suggested Framework (By employing PSO+SOM + Deep CNN in RF)	FlickrLogos-32 PLUS Database	0.973	0.939

Table 8
Comparison of the approaches of Iandola et al. (2015) and Bao et al. (2016) with the suggested framework.

S No.	Methodology	Database	Parameter	
			Mean average Precision	Accuracy
1.	Iandola et al. (2015) a)FRCN+AlexNet b)FRCN+VGG16	FlickrLogos-32 Database	a) 73.5% b) 74.4%	89.6% (maximum with GoogLeNet-GP architecture)
2.	Bao et al. (2016)	FlickrLogos-32 Database	84.2%	----
3.	Suggested Framework (employing PSO+SOM + Deep CNN in RF)	FlickrLogos-32 PLUS Database	97.36 %	94.14 %

References

Alaei A and Delalandre M (2014). A complete logo detection/recognition system for document images. In the 11th IAPR International Workshop on Document Analysis Systems, IEEE, Loire Valley, France: 324-328. https://doi.org/10.1109/DAS.2014.79

Bagheri MA, Gao Q, and Escalera S (2013). Logo recognition based on the dempster-shafer fusion of multiple classifiers. In the Canadian Conference on Artificial Intelligence. Springer, Berlin, Heidelberg, Germany: 1-12. https://doi.org/10.1007/978-3-642-38457-8_1

Bai Q (2010). Analysis of particle swarm optimization algorithm. Computer and Information Science, 3(1): 180-184. https://doi.org/10.5539/cis.v3n1p180

Bao Y, Li H, Fan X, Liu R, and Jia Q (2016). Region-based CNN for logo detection. In the International Conference on Internet Multimedia Computing and Service, ACM, Xian, China: 319-322. https://doi.org/10.1145/3007669.3007728

Bianco S, Buzzelli M, Mazzini D, and Schettini R (2015). Logo recognition using cnn features. In the International Conference on Image Analysis and Processing, Springer, Berlin, Heidelberg, Germany: 438-448. https://doi.org/10.1007/978-3-319-23234-8_41

Eggert C, Winschel A, and Lienhart R (2015). On the benefit of synthetic data for company logo detection. In Proceedings of the 23rd ACM International Conference on Multimedia, ACM, Brisbane, Australia: 1283-1286. https://doi.org/10.1145/2733373.2806407

Feng D, Siu WC, and Zhang HJ (2013). Multimedia information retrieval and management: Technological fundamentals and applications. In: Long F, Zhang H, and Feng DD (Eds.), Fundamentals of content-based image retrieval: 1-26. Springer Science and Business Media, Berlin, Heidelberg, Germany.

Hoi SC, Wu X, Liu H, Wu Y, Wang H, Xue H, and Wu Q (2015). Logo-net: Large-scale deep logo detection and brand recognition with deep region-based convolutional networks. arXiv preprint, arXiv:1511.02462.

Iandola FN, Shen A, Gao P, and Keutzer K (2015). Deeplogo: Hitting logo recognition with the deep neural network hammer. arXiv preprint, arXiv:1510.02131.

Kameyama K, Oka N, and Toraichi K (2006). Optimal parameter selection in image similarity evaluation algorithms using particle swarm optimization. In the IEEE International Conference on Evolutionary Computation, IEEE, Vancouver, Canada: 1079-1086. https://doi.org/10.1109/CEC.2006.1688429

Laaksonen J, Koskela M, Laakso S, and Oja E (2001). Self-organising maps as a relevance feedback technique in content-based image retrieval. Pattern Analysis and Applications, 4(2-3): 140-152. https://doi.org/10.1007/PL00014575

Ma L, Lin L, and Gen M (2012). A PSO-SVM approach for image retrieval and clustering. In the 41st International Conference on Computers and Industrial Engineering, Los Angeles, California, USA: 629-634.

Okayama M, Oka N, and Kameyama K (2007). Relevance optimization in image database using feature space preference mapping and particle swarm optimization. In the International Conference on Neural Information Processing. Springer, Berlin, Heidelberg, Germany: 608-617. https://doi.org/10.1007/978-3-540-69162-4_63

Revaud J, Douze M, and Schmid C (2012). Correlation-based burstiness for logo retrieval. In the 20th ACM International Conference on Multimedia, ACM, Nara, Japan: 965-968. https://doi.org/10.1145/2393347.2396358

Romberg S and Lienhart R (2013). Bundle min-hashing for logo recognition. In the 3rd ACM Conference on International Conference on Multimedia Retrieval, ACM, Dallas, Texas, USA: 113-120. https://doi.org/10.1145/2461466.2461486

Romberg S, Pueyo LG, Lienhart R, and Van Zwol R (2011). Scalable logo recognition in real-world images. In the 1st ACM International Conference on Multimedia Retrieval, ACM, Trento, Italy: 25-32. https://doi.org/10.1145/1991996.1992021

Su JH, Huang WJ, Philip SY, and Tseng VS (2011). Efficient relevance feedback for content-based image retrieval by mining user navigation patterns. IEEE Transactions on Knowledge and Data Engineering, 23(3): 360-372. https://doi.org/10.1109/TKDE.2010.124

Su Z, Zhang H, Li S, and Ma S (2003). Relevance feedback in content-based image retrieval: Bayesian framework, feature subspaces, and progressive learning. IEEE Transactions on Image Processing, 12(8): 924-937. https://doi.org/10.1109/TIP.2003.815254

Suganthan PN (2002). Shape indexing using self-organizing maps. IEEE Transactions on Neural Networks, 13(4): 835-840. https://doi.org/10.1109/TNN.2002.1021884

Tzelepi M and Tefas A (2016). Relevance feedback in deep convolutional neural networks for content based image retrieval. In the 9th Hellenic Conference on Artificial Intelligence, ACM, Thessaloniki, Greece. https://doi.org/10.1145/2903220.2903240

Vesanto J, Himberg J, Alhoniemi E, and Parhankangas J (1999). Self-organizing map in Matlab: The SOM Toolbox. In the Matlab DSP Conference, 99: 16-17.

Wang Z and Hong K (2012). A novel approach for trademark image retrieval by combining global features and local features. Journal of Computational Information Systems, 8(4): 1633-1640.

Xue B, Zhang M, and Browne WN (2013). Particle swarm optimization for feature selection in classification: A multi-objective approach. IEEE Transactions on Cybernetics, 43(6): 1656-1671. https://doi.org/10.1109/TSMCB.2012.2227469