Expert Systems With Applications 170 (2021) 114545

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Additive deep feature optimization for semantic image retrieval

Saddam Hussain [a,*], Muhammad Ahmad Zia [b], Waqas Arshad [b]

[a] Department of Computer Science & IT, The University of Lahore, Sargodha 40101, Pakistan
[b] Department of Computer Science & IT, The University of Lahore, Lahore 54590, Pakistan

ARTICLE INFO

Keywords:
Image retrieval
Image search
Deep learning
Convolutional neural networks
MaxNet

ABSTRACT

Rapid increase in the distribution of multimedia content in recent times presents a challenging problem for content-based image retrieval systems. Image contents, such as position and shape of objects alongside contextual features such as background can be used to retrieve visually similar images. Variations in contrast, color, intensity and texture of contextually similar images make it an interesting research problem. A deep convolutional neural network-based model called MaxNet for content-based image retrieval is presented in this paper. The proposed system bypasses the reliance on handcrafted features and extracts deep features directly from the images, which are then used to retrieve contextually similar images from the database. The proposed MaxNet model is built by stacking the updated inception module in a hierarchical fashion. Features extracted from various pipelines in the inception module are aggregated after each inception maximizing the feature values. This novel aggregation step generates a model that is able to adapt to variety of datasets. Various types of aggregations are discussed in this study. Model overcomes the over-fitting problem by using a dropout layer after each inception block and just before the output layer. The system outputs softmax probabilities, which are stored in the feature database and are used to compute the similarity index to retrieve images similar to the query image. The MaxNet model is evaluated using four popular image retrieval datasets namely, Corel-1k, Corel-5k, Corel-10k and Caltech-101, where it outperforms state-of-the-art methods in key performance indicators.

1. Introduction

With increase in the use of internet, an even greater increase is noticed in digital data. A large quantity of information is generated everyday. The need to find relevant information from multimedia databases has great relevance in digital systems in recent times. Multimedia retrieval has applications in the fields of medicine, fingerprint identification systems, digital libraries, crime prevention, historical research, search engines, photograph archives etc (da Silva Torres & Falcao, 2006). To counter this need, efficient image retrieval techniques are required. The process of acquisition, transmission and storage of images is a significant problem. One of the most crucial factors affecting the performance of retrieval systems is the representation of features and similarity measurements. Research has been going on for several decades but recent developments in machine learning have revitalized research in this field.

Numerous techniques have been proposed so far but content-based image retrieval (CBIR) is still an on-going challenge. The major challenge in this domain is the issue of semantic gap that subsists between the low-level image features and high-level semantic concepts (Mehmood, Mahmood, & Javid, 2018). Low-level features reside at pixel level and are extracted using machine learning algorithms whereas, semantic features rely on human perception. This problem poses a challenge to artificial intelligence (AI) systems to build and train machines that are significantly intelligent and imitate human brain to deal with real world problems.

Several techniques have been developed to address the problem of image retrieval from multimedia databases (Saritha, Paul, & Kumar, 2019). Text-based Image Retrieval (TBIR) systems were introduced in 1970s (Sarwar et al., 2019). Text and annotated descriptors-based techniques were employed to retrieve particular set of images. The drawbacks to these legacy methods include: the need of manual annotation of images causing wastage of time, the accuracy becomes dependent on the perception of individuals, and the methods rely on the

* Corresponding author.
E-mail addresses: saddam4440@gmail.com (S. Hussain), muhammad.zia@cs.uol.edu.pk (M.A. Zia), waqas.arshad@cs.uol.edu.pk (W. Arshad).

https://doi.org/10.1016/j.eswa.2020.114545
Received 25 June 2020; Received in revised form 4 December 2020; Accepted 24 December 2020
Available online 1 January 2021

language used for annotation text.

CBIR systems were introduced in 1990s to overcome the issues faced by the TBIR systems (Jain & Singh, 2011). CBIR is an automated system of indexing images for efficient retrieval by extracting low-level features from images, such as shape, color, and texture. Recently, deep learning-based techniques have come to fore as significant innovative solutions to image retrieval problems (Saritha et al., 2019). Deep convolutional neural network (DCNN) model works by feeding images in the input layer of the pre-trained model that extracts features either from activation values in a fully connected layer or from the spatial information present in convolutional layers. These features provide the basis for image retrieval task in CBIR systems. CNNs have proven to yield greater performance compared to machine learning algorithms relying on handcrafted features (Tolias, Sicre, & Jégou, 2015). Multiple layers of convolution kernels make the DCNN model a capable device possessing subjective learning capacity that is able to detect complicated representations in vision and recognition processes.

Numerous CNN-based techniques have been developed for image retrieval from multimedia datasets (Zhe, Chen, & Yan, 2019). Most of these methods make use of local features to produce a general image representation using pre-trained CNNs. However, various open queries and challenges require further investigation. The effects of fine-tuning the pre-trained CNN model using transfer learning to the CBIR task needs model trained previously on similar data. It also faces the challenge of over-fitting, as the model is fine-tuned using a small dataset. Secondly, the comparison of discriminatory power of features extracted through convolutional layers and the features quantized through the generic methodologies such as, Vector of Locally Aggregated Descriptors (VLAD) (Spyromitros-Xioufis, Papadopoulos, Kompatsiaris, Tsoumakas, & Vlahavas, 2014) uses additional classification and clustering algorithms. This makes it a very complex mechanism to use. Thirdly how the high-dimensional image features extracted through existing CNN-based architectures can be reduced. Finally, appropriate research is required to identify, how several aspects of CNN can be connected efficiently e.g. similarity or dissimilarity matching, query handling and performance of retrieval in terms of searching time and memory use. These challenges have led the way to this work.

A deep CNN architecture is proposed to address the problems related to model fine-tuning, features quantization, high-dimensionality features and effects on the performance of the system by the features characteristics such as, length and training procedure. The proposed architecture introduces an aggregation step that adds deep features after every inception module to reinforce deep features. Six types of aggregations namely addition, subtraction, multiplication, maximum, minimum and concatenation have been experimented on, where addition has generated superior results. Subtraction and multiplication produced meagre results and therefore, are not discussed in detail in the following sections.

The proposed CNN architecture named MaxNet draws inspiration from Residual Network (ResNet) (Zhe et al., 2019) and googleNet (Liu, Guo, Wu, & Cai, 2017), which are widely known deep learning architectures for computer vision tasks. Initially, the goal was to combine the residual block of ResNet with parallel stacking of googleNet. But it yielded poor results for this particular task. Therefore, MaxNet model has been developed that takes the concept of parallel stacking further to aggregate results of the updated inception module. The proposed network divides the feed-forward pass in three different streams, and then adds them forming an inception block. This makes the network deeper while retaining efficiency and thus increases its learning capacity. The CNN model is trained on four standard datasets for image retrieval problem. The proposed model classifies images in the dataset and outputs features, which are further used to retrieve images similar to the query image from the dataset. Following are the major contributions of this research article:

- A fully automated CNN-based algorithm using three prong parallel streams forming an updated inception block is proposed for the image retrieval problem.
- The proposed network takes into account both local and contextual information making it a suitable tool for reducing semantic gap between low-level features and high-level semantics in image retrieval task.
- The proposed CBIR system uses a very sparse feature space to retrieve images effectively, which is an unprecedented practice in image retrieval task.
- The proposed MaxNet model incorporates an aggregation step that adds features after each iteration to reinforce deep features. This reinforcement step alongside large max pool layers establish a model that is both effective and efficient in terms of performance metrics.
- This study presents results for a variety of popular datasets under different conditions and sets a benchmark for future research studies.

The remainder of this paper presents related work in Section 2, proposed methodology in Section 3, experimental setup in Section 4, achieved results in Section 5, and conclusion in 6.

2. Related work

Literature review on the subject of image retrieval can be divided into three major categories based on historical development of retrieval systems. These systems include generic CBIR systems, hybrid CBIR systems, and CNN-based image retrieval systems.

2.1. Content-based image retrieval

CBIR techniques work by extracting feature representations from the images residing in large databases (Kumar & Gopal, 2014). Eakins in Eakins (2002) has divided image features into various levels. Primitive features like texture, color, spatial location or shape of image elements are categorized as level one features. Whereas, second level features are logical features comprising some level of implication about the individuality of an image. Local feature descriptors have also been examined, such as speeded up robust features (SURF) (Bay, Tuytelaars, & Van Gool, 2006) scale-invariant feature transform (SIFT) (Lowe, 1999) and bag of words (BoW) (Yang, Jiang, Hauptmann, & Ngo, 2007; Sivic, Russell, Efros, Zisserman, & Freeman, 2005; Wu & Hoi, 2011; Ahmed, Naqvi, Rehman, & Saba, 2019). Many studies targeting the image retrieval task use these features (Wangming, Jin, Xinhai, Lei, & Gang, 2008; Mansoori et al., 2009; Alfanindya et al., 2013), but the problem of the semantic gap is still unaddressed. SIFT features have been used in Wangming et al. (2008) to represent the training images and queries. Approximate nearest neighbor and KD-tree with Best-Bin-First (BBF) is applied to SIFT to classify and index images. A CBIR system (Mansoori et al., 2009) uses texture and color features interpreted as low-level features and binary tree data structure to capture high-level features to represent images. In an effort to improve results, a set of three features for image content descriptors including Gabor Wavelet, Color auto-Correlogram and Wavelet Transform is used in Anandh, Mala, and Suganya (2016). A hierarchical clustering technique to group together images into visually separable clusters within the dataset has been presented in Pandey and Khanna (2016). A more recent CBIR method based on local energy-oriented patterns (LEOP) has been proposed in Galshetwar, Waghmare, Gonde, and Murala (2019). It finds minute spatial features instead of relying on the relationship between neighboring pixels in contrast with traditional approaches.

In the past decade, various machine learning approaches have been

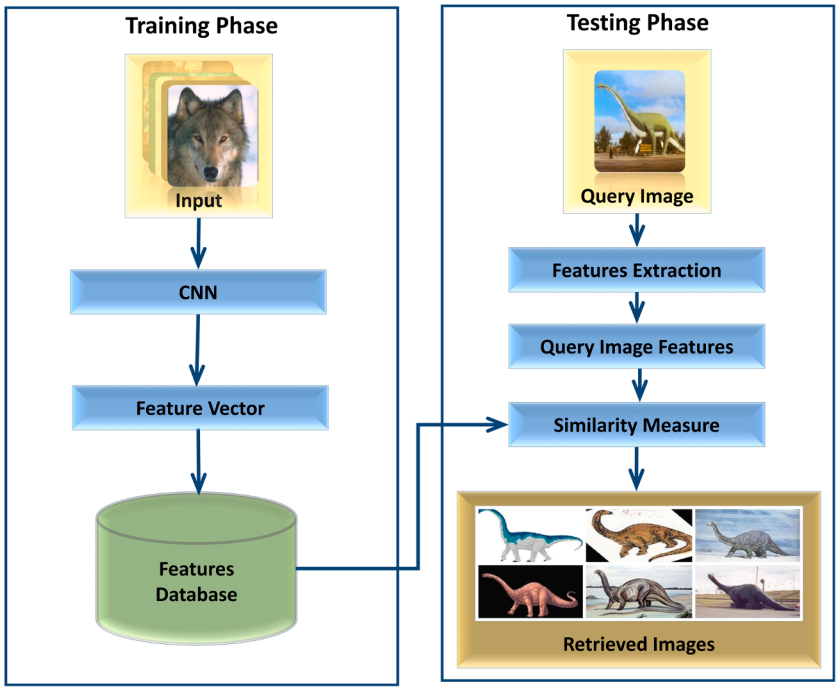

Fig. 1. Block diagram of the proposed methodology.

used to deal with the problem of the semantic gap through local features. Some of them have targeted learning compact or hash codes. In Zhang, Zhang, Gu, Tang, and Tian (2014), a hashing method called Topology Preserving Hashing (TPH) produces good results by ranking the pixel neighborhood. It also presents three additional TPH methods using kernel-based algorithms, supervised learning, and unsupervised learning. In Yu et al. (2018), a multi-trend binary coding descriptor (MTBCD) is proposed that extracts local features based on change in pixel trends. Recently, Benjamin Klein in Klein and Wolf (2019) proposed a hashing method that uses deep product quantization (DPQ). It learns both soft and hard representations of an image to accurately classify images into respective classes. Reference Jegou et al. (2011) presents a methodology that gathers local image descriptors into firm codes to boost the constraints of efficiency, searching accuracy and memory usage.

A lot of attention has been diverted towards the improvement of features in the image retrieval domain. It has been established that a single feature cannot represent an image completely. Therefore, local and global features have been combined to classify images in Xie, Qin, Xiang, Li, and Pan (2018). XOR patterns are used to achieve substantial improvement in features for the image retrieval task in Bala and Kaur (2016). A uniform partitioning scheme is applied in the HSV color space to extract dominant color descriptor (DCD) features in Fadaei, Amirfattahi, and Ahmadzadeh (2016). An approach to add the image spatial contents to the inverted index of the Bag of Words (BoW) model in order to reduce over-fitting has been discussed in Mehmood et al. (2018). It discusses the problem with a large sized dictionary of words as well as the semantic gap issue between high-level image semantic and low-level image features.

2.2. Hybrid CBIR systems

In the last few years, various approaches for combining and fusing different image descriptors have been suggested (Krizhevsky, Sutskever, & Hinton, 2012). Most of these approaches are tailored to the strategy of combining and selecting descriptors for a particular application area.

A hybrid CBIR system is proposed by combining parametric and non-parametric features in Rana, Dey, and Siarry (2019). It aggregates color moments, rankled transformation, and moment invariants to form a hybrid CRM model for content-based image retrieval. A CBIR variation is proposed in Ashraf et al. (2018) that uses the combination of texture and color features to establish the local feature vector. Various distance metrics and hybrid features for a CBIR system are studied in Mistry, Ingole, and Ingole (2017). The hybrid features combine spatial, edge and color descriptors with binary statistical image features (BSIF). A study on image feature information fusion for CBIR systems is presented in Ahmed, Ummesafi, and Iqbal (2019). They fuse the shape and spatial color features for the object recognition tasks. The fusion of visual words of binary robust invariant, SIFT and scalable descriptors has been achieved in Sharif et al. (2019) that relies on the visual BoW approach. A hybrid algorithm is developed by combining binary ant colony optimization algorithm with block truncation coding to retrieve images from Corel datasets in Chen, Chang, Lin, and Hsu (2019). A descriptor based on local color texture called local binary pattern for color images (LBPC) is discussed in Singh, Walia, and Kaur (2018). Furthermore, LBPC is combined with color histogram (CH) and local binary pattern of the hue (LBPH) to generate a fused model for effective image retrieval. An end to end image retrieval system based on deep learning and differential learning system on the Caltech dataset has been presented in Tian, Zheng, and Xing (2018). They have found the differential learning method superior to many conventional systems.

2.3. CNN-based image retrieval

The progressive deep learning and machine learning methodologies have given new means of narrowing the semantic gap by learning visual

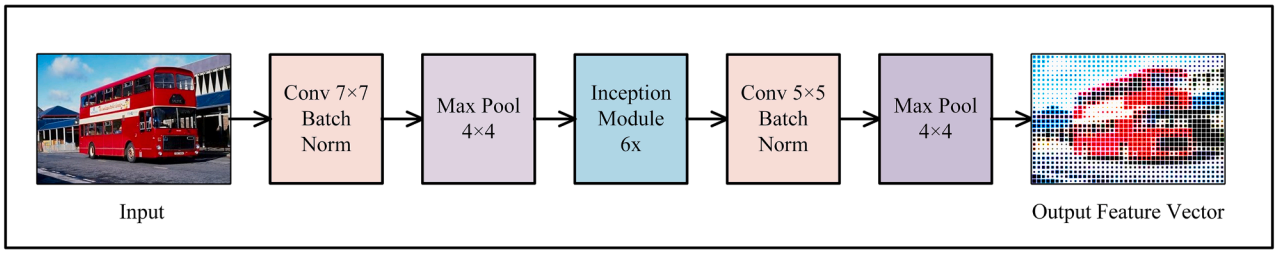

Fig. 2. Proposed MaxNet deep learning architecture.

features straight from the images alleviating the need for hand-crafted features. Initially, a DCNN model for image classification problem on ImageNet dataset was proposed by Hinton in Krizhevsky et al. (2012). Several approaches followed and improved results employing advancements in the structure of traditional CNN. CNN's have recently been used for image retrieval owing to their built-in tendency to recognize patterns in images (Babenko, Slesarev, Chigorin, & Lempitsky, 2014; Babenko & Lempitsky, 2015). A region-based cross-matching algorithm to collect the maximum similarity per query region is proposed in Babenko et al. (2014). Sum-pooling is applied in Kalantidis, Mellina, and Osindero (2016) to whiten the region descriptors. The method presented in Kalantidis et al. (2016) is extended in LeCun, Bottou, Bengio, and Haffner (1998) by incorporating aggregation of neural codes and cross-dimensional weighting. A deep belief network (DBN) is used in Saritha et al. (2019) to analyze the potency of deep learning in extracting meaningful features for CBIR. Another deep learning framework for histopathology image classification and retrieval is presented in Peng, Boxberg, Weichert, Navab, and Marr (6614). It incorporates the concept of K-nearest neighbors in deep learning to retrieve visually similar images. A semi-supervised deep learning method called semi-supervised multi-concept retrieval to semantic image retrieval via deep learning (SMRDL) is used in Xu, Huang, and Wang (2019) to annotate web images. Another semi-supervised method based on CNN is presented in Li, Wan, and Xie (2018) for image retrieval tasks. It combines Conv-Deconv network with Robust-KSH and learns the hash function based on features extracted by convolution layers.

Deep learning is still a new prospect in image retrieval domain and has not been sufficiently explored on datasets used in this work. Therefore, a detailed study on the impact of deep learning for retrieving semantically similar images is presented in this paper.

3. Proposed methodology

The proposed methodology includes a pre-processing step followed by the MaxNet and a similarity index computation process as shown in Fig. 1. It comprises of two major phases i.e. The classification phase and the retrieval phase. Classification phase is more frequently used to train the MaxNet model. Whereas, to test the proposed CBIR system, retrieval phase is used. Images in the dataset are converted into grayscale which is added as the fourth channel alongside red, green and blue channels to form a four-dimensional input for the MaxNet model. Images are normalized to maintain zero mean value and variance near to one. A normalized image channel is generated by subtracting mean value from the pixel values and dividing the resultant pixel value by standard deviation. It can be expressed as follows,

$$I_n = \frac{(I-\mu)}{\sigma} \tag{1}$$

Where I_n, I, μ and σ represent the normalized channel, original channel, mean intensity value and the standard deviation respectively. To reduce the use of system resources, images are reshaped into a uniform resolution of 100×100 pixels. The use of low-resolution images for training the MaxNet model increases the efficiency of the system without comprising on the accuracy of image retrieval as fewer kernels are needed to convolve over the input feature maps. These images are passed through the MaxNet model for training purposes and finally feature vectors of all images are extracted to form the feature database. This database is used while retrieving images relevant to the query image. Detailed methodology is discussed in subsequent subsections.

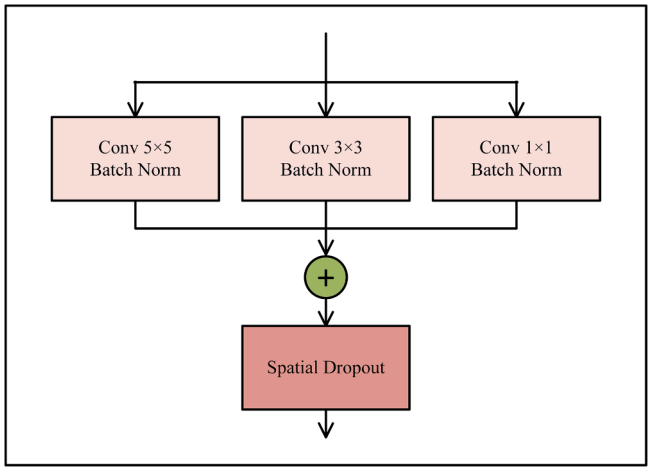

Fig. 3. Proposed inception module for MaxNet model.

3.1. Convolutional neural networks

A deep learning-based image retrieval system is proposed in this work. The DCCN model focuses on multiple aspects present in an image to extract global as well as local information. DCNN models learn feature vectors directly from the images in contrast to traditional hand-crafted feature extraction practices (Sajid, Hussain, & Sarwar, 2019). This makes DCNN a powerful computer vision tool despite its relatively simple mathematical theory. DCNN's goal is to learn different kernels that produce the best abstract representation of features in the dataset. The convolution feature map F_{ci} is calculated as;

$$F_{ci} = b_c + \sum_i M_{ci} * I_i, \tag{2}$$

where bias, convolution kernels and input map are represented by b_c, M_{ci} and I_i respectively. $*$ operator is used to denote the convolution operation.

DCCN models are generally built by stacking three major layers one over the other. These layers include convolution layers, pooling layers

and fully connected layers (Hussain, Anwar, & Majid, 2017). These layers are stacked in between input and output layers. Other than these three layers there are many minor layers available such as dropout, batch normalization, adder and subtractor layers etc. Convolution layer forms the single most important unit of CNNs. It is the layer that contains various sized kernels that convolve over the input map to catch significant features present in the image. Pooling layers are most widely used for their utility in down-sampling the input feature map and in dealing with translational invariance. Max-pooling layer used in this setup selects the maximum feature value residing within rectangular subsection sized $p \times p$ of the feature map. Pooling layer down-samples the input feature map either by increasing the size of rectangular window called pool size or by increasing the pool stride. Pool stride determines how big a jump pooling window makes over the input feature map. Max-pool window of size $p \times p$ computes the output $P_{(i,j)}$ for activations in the feature map F as;

$$P_{(i,j)} = max\left(F_{(i+p,j+p)}\right) \tag{3}$$

Fully Connected layers are commonly used near the end of the CNN architecture to output the class distribution probabilities. The proposed MaxNet model uses these layers to form a generic framework for image retrieval task. A detailed description of the proposed framework is presented in the following section.

3.2. MaxNet classification

The most important stage in the proposed CBIR system is the classification stage, in which a DCNN model is trained using supervised learning practices to classify images belonging to various classes. The ultimate goal of this phase is to predict the class label of a particular image as well as to extract feature vector that will be used in the second phase to retrieve visually similar images. The proposed MaxNet model contains a total of twenty-one convolution layers that are iterated in a structured way to extract maximum information from the images. A pictorial representation of the proposed architecture is presented in Fig. 2.

The proposed network takes 4-dimensional input, where the three dimensions are the red, green and blue planes, while fourth dimension is the grayscale variant of the image. A three-prong structure is used that uses the parallel processing capacity of the system and then sums up the corresponding output. This parallel stacking of layers is called the inception module and is shown in Fig. 3. Inception module has originally been introduced in Szegedy et al. (2015) for ILSVRC 2014 classification challenge. It contains 8 convolution layers and concatenate the features to merge the pipelines. The updated inception module in contrast contains three convolution layers having kernels i.e. $5 \times 5, 3 \times 3$, and 1×1. Feature maps of these kernels are varied in each inception module to maximize variety in feature output. The inception module is repeated six times to produce a deep CNN architecture, and a convolution layer containing 7×7 kernels is included after every triplet of inception modules. Large kernels add more contextual information enhancing the quality of feature vector going into the inception module. Thus, this large kernel marks another difference between original inception module and the proposed updated module. All convolution layers are followed by a batch normalization layer (Ioffe & Szegedy, 2015). Batch normalization layer is used to normalize activations of preceding layer. It maintains mean activation value and standard deviation close to zero and one respectively. A deeper MaxNet model containing nine inception modules has also been tried but it generates lower results compared to the proposed MaxNet model. Therefore, it has been found that increasing the depth of CNN does not equate better results in every case. In between the inception modules, a max-pooling layer is added to summarize features in a rectangular fashion by selecting the maximum value in the vicinity.

The deep features extracted from inception module are aggregated after each iteration. Additive aggregation is proposed, as it yields the best experimental results in terms of evaluation measures. The additive aggregation is denoted by a plus sign in Fig. 3. It has been noted that summing the deep features supplement them in comparison to subtraction, multiplication, maximization, minimization and concatenation.

Concatenation of features after inception module has been proposed in Szegedy et al. (2015) as well and it performs sufficiently well but additive aggregation outperforms even concatenative aggregation. Four types of aggregative steps are discussed in the following subsections.

3.2.1. Maximization

In this type of aggregation, a maximum function is used at the end of inception module to select deep features that have maximum values between three inception pipelines. In this way features with low or insignificant values are discarded reducing complexity that is found in original inception module. Maximization layer computes feature maps as,

$$O_f = max\left(IP_1, IP_2, IP_3\right), \tag{4}$$

where O_f, IP_1, IP_2, and IP_3 denote the output deep feature map of inception module, input stream one, input stream two and input stream three of the inception module.

3.2.2. Minimization

In contrast to maximization, a minimum function is used after each inception module to select deep features with minimum values. This type of aggregation has been experimented on to demonstrate that low feature values do not produce significant results for image retrieval task. Minimized deep feature map is output as,

$$O_f = min\left(IP_1, IP_2, IP_3\right), \tag{5}$$

3.2.3. Concatenation

Concatenation has been proposed in the original inception module as well (Szegedy et al., 2015). In this step, deep features extracted from inception pipelines are joined together creating an array of features equal in size to three arrays extracted from inception pipelines. This way no feature is discarded as in the case of maximization and minimization. Concatenation does produce good results but at the cost of complexity and efficiency of the system. This is due to the fact that concatenation retains significant as well as insignificant features and thus has a high number of parameters. Therefore, it also contains the risk of overfitting the data. The large concatenated deep feature map is generated as,

$$O_f = concat\left(IP_1, IP_2, IP_3\right), \tag{6}$$

3.2.4. Additive aggregation

Additive aggregation produced the best results without compromising the efficiency or simplicity of the model. This step adds the deep features extracted from different pipelines increasing the feature values significantly. This action leads to increase in the significant feature values and thus reports better regularities for model to map on. Since features are added, it contains significantly lower number of features compared to concatenation step and therefore, is both efficient and simple. Additive aggregation of deep feature maps is computed as,

$$O_f = add\left(IP_1, IP_2, IP_3\right), \tag{7}$$

Output of the last inception module is fed into a convolution layer

Fig. 4. Sample images from three Corel Datasets i.e. Corel-1k, Corel-5k, and Corel-10k and Caltech-101 dataset.

containing 5×5 kernels followed by max pool of 4×4 to downsample feature vector. It is further connected to output layer to produce class distribution probabilities using softmax activation function. These softmax probabilities are used as feature vectors to form a feature database. This database is used in the retrieval phase for query matching and image retrieval. Softmax non-linear features *SF* for every class label *l* are computed as,

$$SF(c = l|F) = \frac{e^{FW_j}}{\sum_{l=1}^{C} e^{XW_l}}, \tag{8}$$

where F, W, and C denote the feature vector, weight parameters and the total number of classes, whereas, c is the class probability of an image.

3.3. Image retrieval

After training the MaxNet model, the dataset is fed into the network, and the feature vectors are extracted from the last layer of the proposed MaxNet model. These feature vectors represent all the images in the dataset. These features are stored in the feature database as shown in Fig. 1. The query image is also feed-forwarded through the network and the extracted features are compared with feature vectors of the entire dataset. The similarity measure used for comparison of features between the query image and rest of the dataset is the Euclidean distance measure (Kokare, Chatterji, & Biswas, 2003). In addition to feature comparison, the predicted class label is used to limit the area of interest, thus ignoring the irrelevant images. It is also a useful way to reduce computational time for retrieval systems. Images that are closer to the query image in terms of similarity index are displayed as retrieved results.

4. Experimental setup

4.1. Dataset

The proposed system is validated on Corel-1k (Mehmood et al., 2018), Corel-5k (Dua & Graff, 2017), Corel-10k (Tian, Jiao, Liu, & Zhang, 2014), and Caltech-101 (Tian et al., 2018) datasets. Corel-1k dataset contains one thousand images, which are divided into ten classes. Corel-5k dataset is an upgraded version of Corel-1k and contains five thousand images classified into fifty classes. Corel-10k dataset contains ten thousand images categorized into one hundred classes, whereas, each class contains one hundred images. Caltech-101 dataset contains nine thousand one hundred and forty-five images in total classified into one hundred and two classes. The number of images in individual classes vary between the range of thirty-two to eight hundred per class. Therefore, it is highly imbalanced dataset. These datasets contain diverse scenes such as, beach, sunset, horses, cars, humans, mountains, flowers etc. Images are labelled with numeric values [0 to n − 1], where n represents the number of classes in a dataset. Visual representation of images from Corel and Caltech-101 datasets is shown in Fig. 4.

4.2. Implementation details

The model is built using keras library which is a popular deep learning library for designing deep neural network architectures (Charles, 2013). System is implemented in python. 5-fold cross validation is employed to select the best performing model on the given dataset. Hyper-parameters such as learning rate, momentum, initializers and activation functions are varied during training process to minimize error and maximize accuracy. Model is trained on 80% of the images and validated on remaining 20% for all datasets. Images are selected

Fig. 5. Results of top 20 images retrieved from semantic class 7 of Corel-1k dataset.

randomly for training and testing purposes.

Stochastic gradient descent (SGD) (Hussain, Anwar, & Majid, 2018) optimizer is used for the purpose of updating weights in the DCNN. SGD calculates negative log probability to determine gradient change ΔW_i in weights. ΔW_i is disseminated back to update weights of the network using the back-propagation algorithm. Objective function $G(\theta)$ is optimized by using Nesterov accelerated gradient velocity (NAGV) (Sajid et al., 2019). Velocity at current iteration V_i, learning rate L, momentum coefficient μ, and true gradient at current iteration $\nabla G(\theta_i)$ is used to calculate velocity for next iteration V_{i+1}. Gradient at $i+1$ is computed by adding V_{i+1} to gradient θ_i at i.

$$V_{i+1} = \mu V_i - L\nabla G(\theta_i + \mu V_i), \tag{9}$$

$$\theta_{i+1} = \theta_i + V_{i+1}. \tag{10}$$

Model is trained using mini-batches which is a popular method to reduce resource utilization in training DCNNs. To ensure generalization, a spatial dropout of 0.3 is used after each inception block. A dropout value of 0.1 is added just before the softmax probabilities to ensure extraction of quality features and diminish effects of over-fitting. Bernoulli equation is implemented in the dropout layer to generate a dropout vector V_i. Elements in V_i represent probability p of feature retention. V_i is overlapped with input feature map F and after multiplying $\otimes$ corresponding elements in two maps, the sparse feature map $\widetilde{F}$ is generated.

$$V_i = Bernoulli(p), \tag{11}$$

$$\widetilde{F} = V_i \otimes F_i. \tag{12}$$

Model is trained for twenty epochs initially on Corel-10k dataset. By using the concept of transfer learning, it only requires eight epochs for

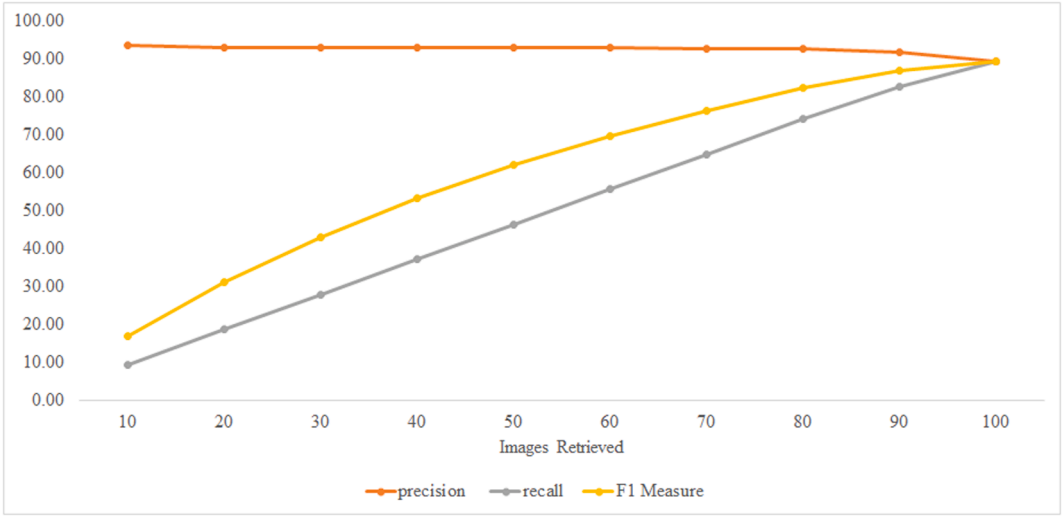

Fig. 6. Class-wise results achieved by proposed method in terms of precision, recall, and F1 measure on Corel-1k dataset.

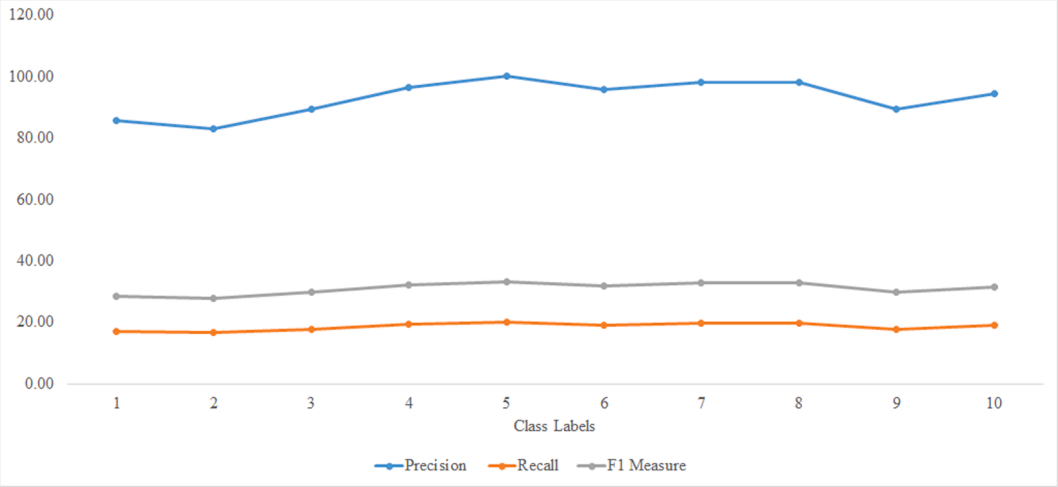

Fig. 7. Image retrieval results in terms of precision, recall, and F1 measure on Corel-1k dataset for TR_N ranging between [10 and 100].

training on Corel-5k dataset. These trained weights are further used to fine-tune MaxNet model on Corel-1k and Caltech-101 datasets.

4.3. Query matching

The proposed deep learning model is used to extract feature vector from its last layer for all images in the datasets. While testing the model, each image in the dataset is used as the query image and thus it is passed through the MaxNet model. The proposed model outputs a feature vector for each query image, which is then compared with feature vectors of all the images stored in the database. For this comparison, Euclidean distance measure (Kokare et al., 2003) is employed in this study. The goal of query matching is to get the most similar images to the query image. Feature vectors with lowest Euclidean distance between them and query vector are identified. Corresponding images of these feature vectors are presented as retrieval results. Euclidean distance is calculated as follows,

$$D_{(Q,R)} = \left(\sum_{i=1}^{n} \left(f_{Q_i} - f_{R_i} \right)^2 \right)^{1/2} \tag{13}$$

where $D_{(Q,R)}, f_{Q_i}, f_{R_i}$ and n represent the Euclidean distance measure, feature vector of the query image, feature vector of images stored in the database and the total number of images in feature database respectively.

4.4. Evaluation parameters

The model is evaluated using three metrics, namely precision, recall and F1 measure (Mehmood et al., 2018). System is evaluated in terms of performance on individual classes as well as based on the number of images retrieved. Images are retrieved ranging between ten and one hundred against all images in the dataset. This practice presents the detailed examination of performance of the proposed system.

The precision evaluation metric is the ratio of correctly retrieved instances against all retrieved instances. For query image Q, it is given as,

$$P_Q = \frac{I_r}{I_t} \tag{14}$$

where P_Q, I_r and I_t are the precision value for the query image, total number of correctly retrieved images and the total number of images retrieved respectively.

Table 1
Comparison with state-of-the-art methods @$TR_N = 10$ and @$TR_N = 20$ for Corel-1k dataset in terms of precision, recall and F1 measure.

Algorithms	$TR_N = 10$			$TR_N = 20$		
	MAP	MAR	F1-Measure	MAP	MAR	F1-Measure
SIFT + BRISK (Sharif et al., 2019)	–	–	–	84.39	16.87	–
WATH (Mehmood et al., 2018)	–	–	–	87.85	17.37	–
BACOBTC (Chen et al., 2019)	–	–	–	80.56	16.11	–
MTBCD (Yu et al., 2018)	89.59	17.92	–	–	–	–
LTxXORP (Bala & Kaur, 2016)	79.83	7.98	–	73.90	14.78	–
BOW (Ahmed et al., 2019)	76.50	15.3	–	–	–	–
MaxNet (max)	84.94	8.49	15.44	83.71	16.74	27.90
MaxNet (min)	76.56	7.66	13.92	73.11	14.62	24.37
MaxNet (concat)	90.21	9.02	16.40	89.98	18.00	30.01
Proposed MaxNet (add)	93.41	9.34	16.98	93.04	18.61	31.02

Recall on the other hand, is measured in terms of total number of relevant images retrieved from all the images in a particular class. It is given as,

$$R_Q = \frac{I_r}{I_T} \tag{15}$$

where R_Q, I_r and I_T are the recall value for the query image, total number of relevant images retrieved and the total number of images in a particular class respectively.

F1 measure also known as the F1 score, maintains balance between precision and recall values. It is given as,

$$F1 = 2 \times \frac{P_Q \times R_Q}{P_Q + R_Q} \tag{16}$$

Fig. 8. Image retrieval results for top 20 images retrieved from semantic class 18 of Corel-5k dataset.

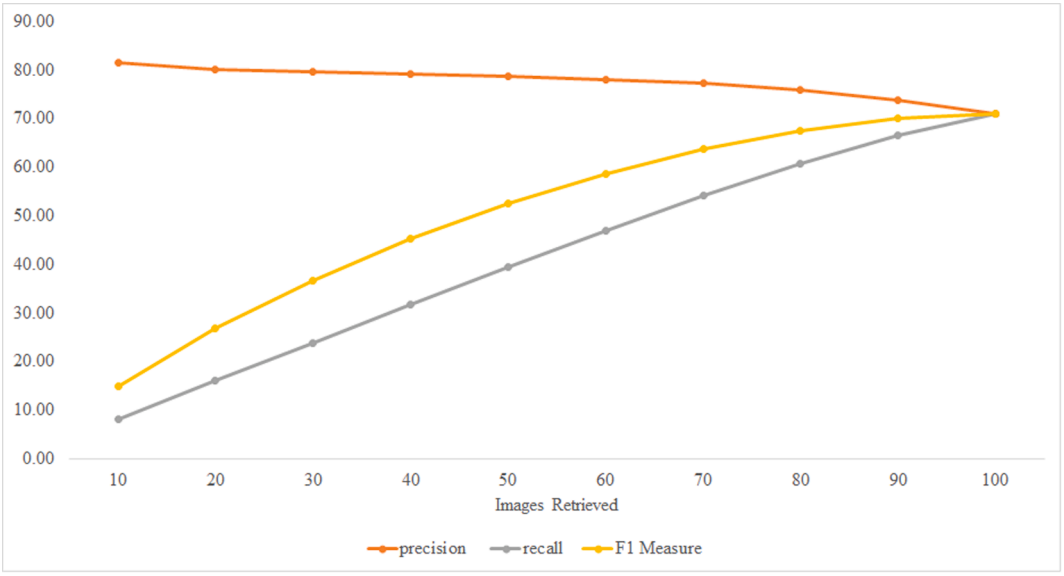

Fig. 9. Class-wise results achieved by proposed method on Corel-5k dataset.

5. Results and discussion

Experiments have been carried out on Corel-1k, Corel-5k, Corel-10k and Caltech-101 datasets. Model is evaluated by determining evaluation metrics for both individual classes as well as the complete dataset. Average value of precision, recall and F1- measure gives the broader picture that represents the entire dataset. Each image in the dataset is used as query image and n relevant images are retrieved using the Euclidean measure given in Eq. 13. The number of correctly retrieved images (CR_N) from the same class as the query image gives the measure of system performance or accuracy. The higher the CR_N, the better the system is performing. CR_N is different from total images retrieved TR_N as TR_N includes incorrect retrieved images as well.

5.1. Corel-1k dataset

A pictorial representation of retrieval results obtained by the proposed CBIR system is shown in Fig. 5. It shows the top twenty images retrieved against a query image. The Euclidean distance measure between retrieved images and the query image is shown on top of individual images. Low distance measure indicates high relevance of retrieved results. The system performs exceedingly well over such a small dataset and retrieves the most relevant images. Fig. 6 presents a

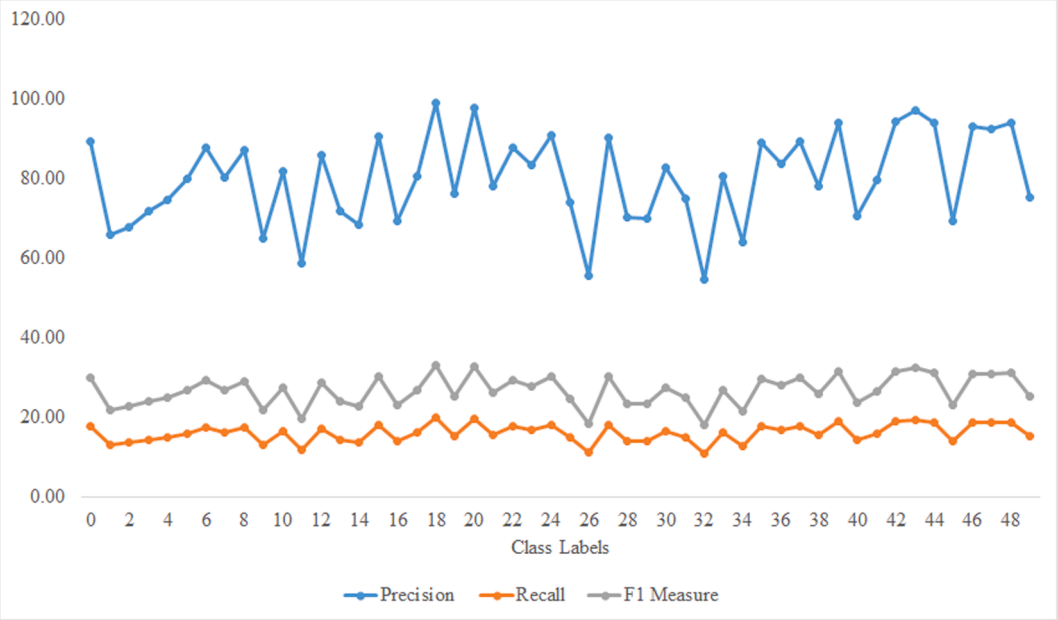

Fig. 10. Image retrieval results for Corel-5k dataset for TR_N ranging between [10 and 100].

Table 2
Comparison with state-of-the-art methods @$TR_N = 10$, @$TR_N = 20$ and @$TR_N = 100$ for Corel-5k dataset in terms of precision, recall and F1 measure.

Algorithms	$TR_N = 10$			$TR_N = 20$			$TR_N = 100$		
	MAP	MAR	F1-Measure	MAP	MAR	F1-Measure	MAP	MAR	F1-Measure
SIFT + BRISK (Sharif et al., 2019)	–	–	–	57.37	11.47	–	–	–	–
MTBCD (Yu et al., 2018)	–	–	–	50.07	10.14	–	–	–	–
CRM (Rana et al., 2019)	67.96	13.6	–	–	–	–	–	–	–
LEOP (Galshetwar et al., 2019)	53.04	5.30	–	44.90	8.98	–	–	–	–
LTxXORP (Bala & Kaur, 2016)	60.82	6.08	–	51.16	10.23	–	–	–	–
LBP (Singh et al., 2018)	47.74	9.55	–	–	–	–	41.68	41.68	41.68
LBP + LBPH + CH (Singh et al., 2018)	55.38	11.01	–	–	–	–	43.81	43.81	43.81
MaxNet (max)	73.98	7.40	13.45	71.06	14.21	23.69	61.75	61.75	61.75
MaxNet (min)	68.35	6.84	12.43	66.86	13.37	22.28	57.13	57.13	57.13
MaxNet (concat)	77.52	7.75	14.09	75.09	15.02	25.03	66.19	66.19	66.19
Proposed MaxNet (add)	81.49	8.15	14.82	80.04	16.01	26.68	70.90	70.90	70.90

graphical view of the performance of proposed system over Corel-1k dataset. It gives class-wise performance in terms of precision, recall and F1 measure. The system gives satisfactory results on all classes and performs exceptionally well for spatially distinct classes. Fig. 7 evaluates the proposed CBIR system in a different dimension. The three-performance metrics are computed in terms of TR_N. It is only logical that system performance decreases with increase in the total number of images retrieved. The proposed model retains its effectiveness even under extreme conditions and only a small drop in performance metrics is observed with increase in TR_N. These results show that the proposed system is very effective in retrieving data from a small repository and maintains its efficacy as data increases. Comparison with state-of-the-art methods on Corel-1k dataset is presented in Table 1. Results shown in Table 1 are computed for $TR_N = 10$ and $TR_N = 20$ as these are the most common evaluation parameters. The proposed CBIR system outperforms its contemporary counterparts in all major performance indicators. Different versions of MaxNet aggregation are also compared and it is evident from results that additive aggregation outperforms all other forms of inception aggregation. The MaxNet model only uses ten features extracted from the output layer of the network making it an extremely viable retrieval system in terms of retrieval time. Once the feature database is established, system needs to compute only a small feature vector to compare Euclidean distance. As establishment of feature space is a one-time task, it significantly boosts system's response time.

5.2. Corel-5k dataset

Corel-5k dataset contains five thousand images classified into fifty classes (Sharif et al., 2019). Fig. 8 shows top twenty images retrieved against a random query image from Corel-5k dataset. MaxNet model adapts well to a diverse dataset and achieves a mean average precision value of 80 over fifty classes in Corel-5k dataset.

Results achieved by MaxNet model on different classes of Corel-5k dataset are shown in Fig. 9. It presents the mean average precision, recall and F1 measure values over individual classes, while retrieving top twenty images. Fig. 10 on the other hand, shows the performance of the proposed system in terms of TR_N. Model performs better when TR_N is low but still maintains good performance indicators as TR_N increases up to one hundred. Since maximum number of images per class is hundred,

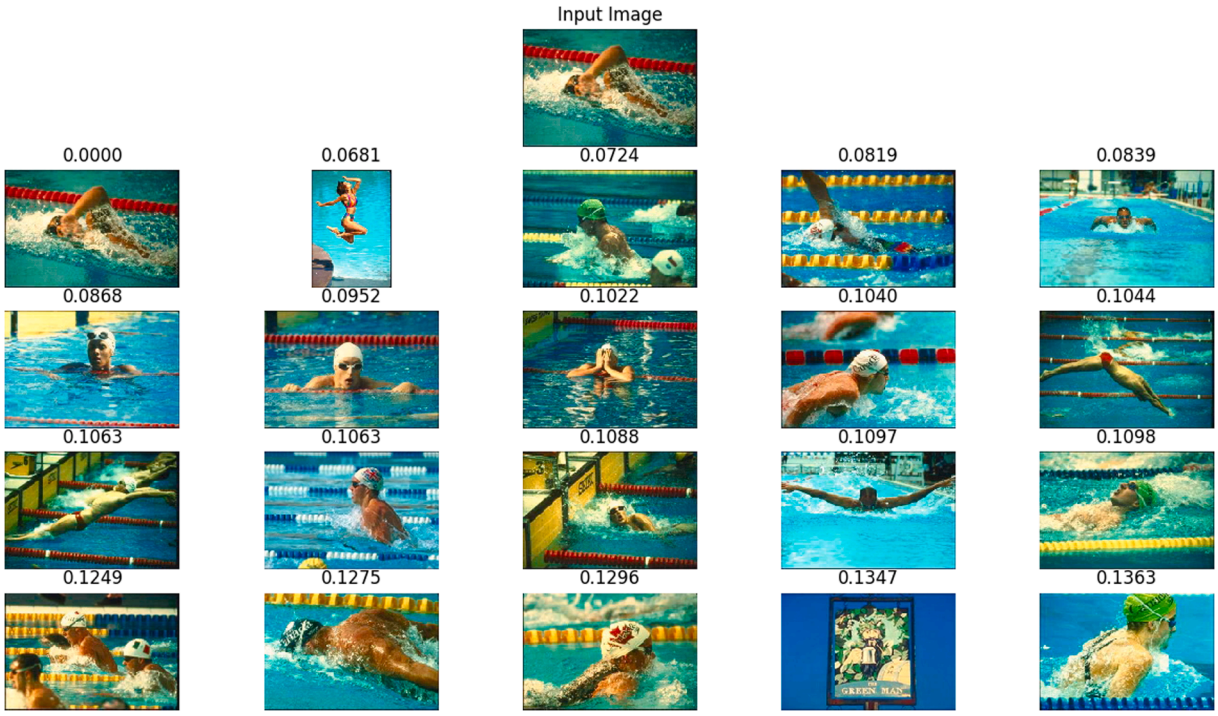

Fig. 11. Image retrieval results for top 20 images retrieved from semantic class 18 of Corel-10k dataset.

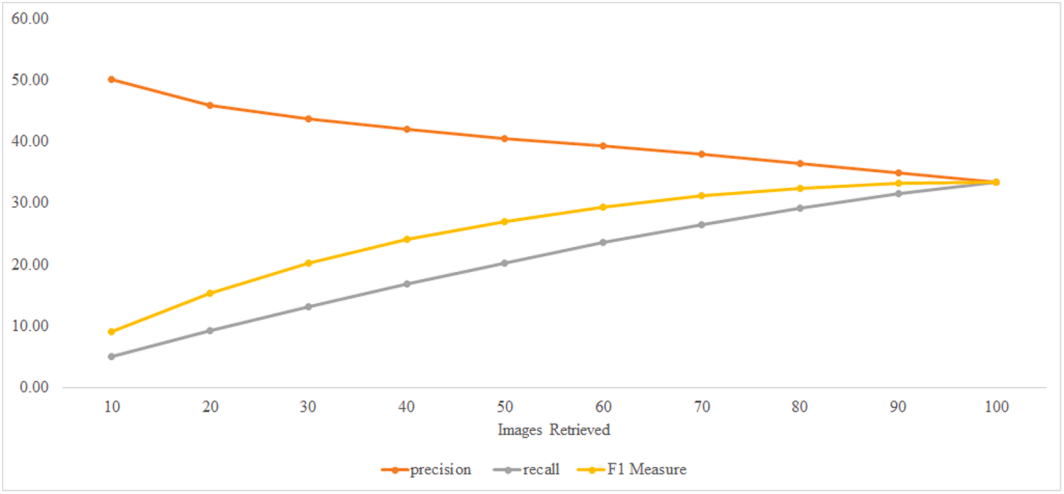

Fig. 12. Class-wise distribution of performance indicators achieved by proposed method on Corel-10k dataset.

therefore, that is the maximum possible value of TR_N. Increase in the number of classes, lowers the capability of system to distinguish different images as it becomes hard for even human evaluators to distinguish between strongly spatially correlated images. But the proposed system still performs better for Corel-5k dataset compared to state-of-the-art retrieval systems as shown in Table 2. Additive aggregation significantly improves results on Corel-5k dataset as well. The proposed system performs better than most state-of-the-art techniques using only fifty features for computing similarity index.

5.3. Corel-10k dataset

Corel-10k is the largest dataset used in this study. It contains ten thousand images segregated into one hundred classes (Singh et al., 2018). In such a diverse dataset, images belonging to the same class can differ spatially and correlate more with other classes. It is extremely challenging task to retrieve relevant images from such complex data. Fig. 11 shows the retrieval results achieved against a query images from Corel-10k dataset by the proposed MaxNet model. Most of the images retrieved are relevant to query image bar few exceptions.

Fig. 12 represents the class-wise distribution of precision, recall and F1 measure over one hundred classes of Corel-10k dataset for top twenty TR_N. Classes that have strong spatial correlation within, achieve much higher values compared to classes with spatially distinct images. Images belonging to the same class vary greatly in some cases leading to lower performance indicators in those classes. The performance of MaxNet

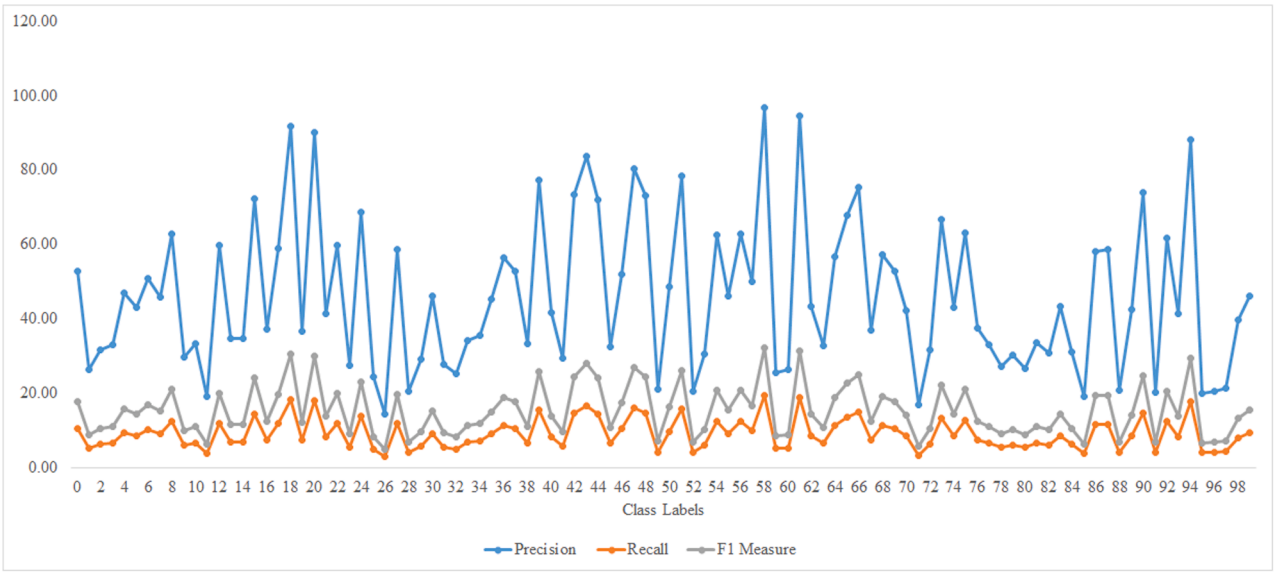

Fig. 13. Image retrieval results for Corel-10k dataset for TR_N ranging between [10 and 100].

Table 3
Comparison with state-of-the-art methods @$TR_N = 10$ and @$TR_N = 20$ for Corel-10k dataset in terms of precision, recall and F1 measure.

Algorithms	$TR_N = 10$			$TR_N = 20$		
	MAP	MAR	F1-Measure	MAP	MAR	F1-Measure
LBP (Singh et al., 2018)	36.99	7.40	–	–	–	–
LBP + LBPH + CH (Singh et al., 2018)	44.94	8.99	–	–	–	–
LEOP (Galshetwar et al., 2019)	48.13	4.81	–	38.05	7.70	–
LTxXORP (Bala & Kaur, 2016)	51.05	5.11	–	41.87	8.37	–
DCD (Fadaei et al., 2016)	–	–	–	36.37	7.27	–
Wavelet Feature (Fadaei et al., 2016)	–	–	–	40.03	8.01	–
MaxNet (max)	43.04	4.30	7.83	38.16	7.63	12.72
MaxNet (min)	39.31	3.93	7.15	33.32	6.66	11.11
MaxNet (concat)	48.22	4.82	8.77	42.97	8.60	14.32
Proposed MaxNet (add)	50.17	5.02	9.13	45.82	9.16	15.27

model in correspondence with TR_N is shown in Fig. 13. These graphs indicate that model performs satisfactorily well even under conditions, which can be very difficult for human cognition. Model achieves a mean average precision and recall values of 50.16 and 5.01 for top ten images retrieved from Corel-10k dataset.

Corel-10k dataset presents a huge multiclass classification problem. Therefore, it has been an exciting problem for researchers around the globe. A comparative analysis with state-of-the-art techniques on Corel-10k dataset is presented in Table 3. Similar trend as in Corel-1k and Corel-5k datasets is observed in terms of aggregation step as additive aggregation is still significantly better. The proposed model uses one hundred features for retrieval task. Increasing the number of features does not significantly boost system performance and extends retrieval time. Therefore, one hundred softmax features extracted from the last layer of MaxNet model are used as the feature vector to compute similarity index.

5.4. Caltech-101

There is a total of nine thousand one hundred and forty-five images in Caltech-101 dataset distributed into one hundred and two classes (Li et al., 2018). The number of images in individual classes vary greatly making the model biased towards highly dense classes. Fig. 14 shows the top twenty images retrieved against a random query image from Caltech-101 dataset. Despite high variance and a large number of classes in the dataset, the proposed CBIR system does a satisfactory job at retrieving most relevant images.

A detailed examination of the performance of proposed system on Caltech-101 dataset is shown in Fig. 15. As the number of images in individual classes vary greatly, model is more biased towards classes with high number of images. Therefore, highly dense classes form the peaks, while sparse classes form the troughs in the graph. Fig. 16 shows the performance of the proposed system against different values of TR_N. It shows the mean average values of three performance measures over a range of $[10-100]$ images retrieved. Since a large number of classes have images less than one hundred and some even have less than fifty, performance indicators fall down rapidly as TR_N increases. This effect surfaces because in some cases, the system is retrieving one hundred images from a class that has only thirty-nine images. The remaining sixty-one images will not belong to the class of query image leading to lower overall performance values for the system. However, this does not compromise the effectiveness of the proposed system as it can only retrieve data, that is available in the repository. Therefore, it's a viable solution to the image retrieval problem in practice.

Table 4 presents a comparative analysis of the proposed system with the recent methodologies applied on Caltech-101 dataset. The results shown are based on top ten and twenty images retrieved by various techniques in the literature. The proposed framework uses one hundred and two softmax features for this dataset to compute similarity index. It is testament to the capability and effectiveness of the proposed system that it performs significantly better than state-of-the-art methods despite comparatively low feature space.

The updated Inception module has proven to be a capable device for learning complex features despite being a simpler version of original inception module proposed in Szegedy et al. (2015). A variation of the proposed system adding residual from previous inception block to current inception block has also been tested but it failed to produce significant results. Therefore, they have not been mentioned in this paper.

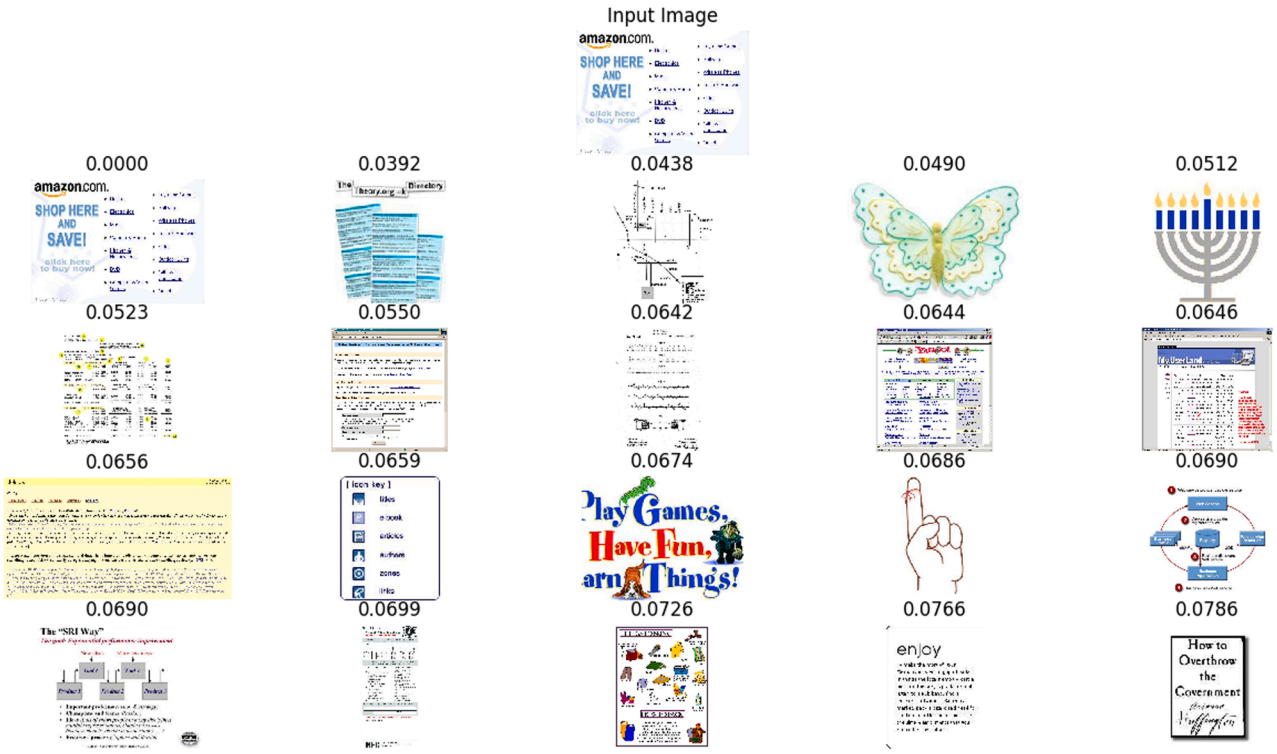

Fig. 14. Image retrieval results for top 20 images retrieved from semantic class 38 of Caltech-101 dataset.

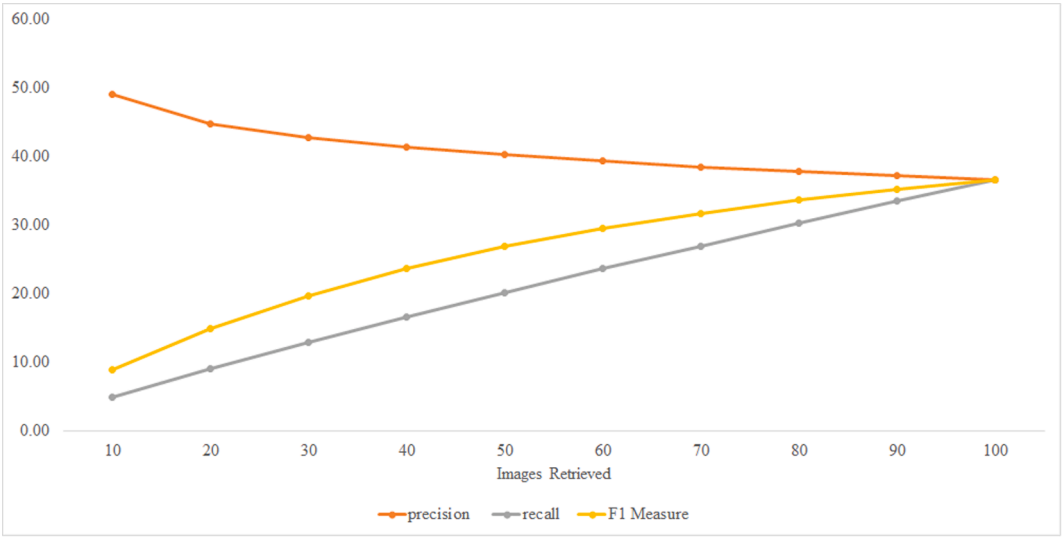

Fig. 15. Results achieved by the proposed method on Caltech-101 dataset for various classes ranging between [0 and 101]. High fluctuations occur due to high variance in number of images in each class.

Dividing the proposed neural network architecture into various pipelines leads to better performance as it uses parallel processing thus reducing its running time. In order to reduce running time further, the number of features in the fully connected layer has been varied from one hundred to two thousand. Increasing features leads to overfitting problem whereas, decreasing features compromises system performance. Therefore, optimal features are selected for the proposed system. It is observed that reducing the features improves running time as convolution layers are much faster than fully connected layers.

It is also noted that most of the studies in literature do not provide extensive results under different conditions. It is also evident from comparison tables as results for various conditions could not be found in the literature. Therefore, this paper serves as a benchmark for future studies as it presents extensive results for varying values of TR_N as well as extensive results for individual classes.

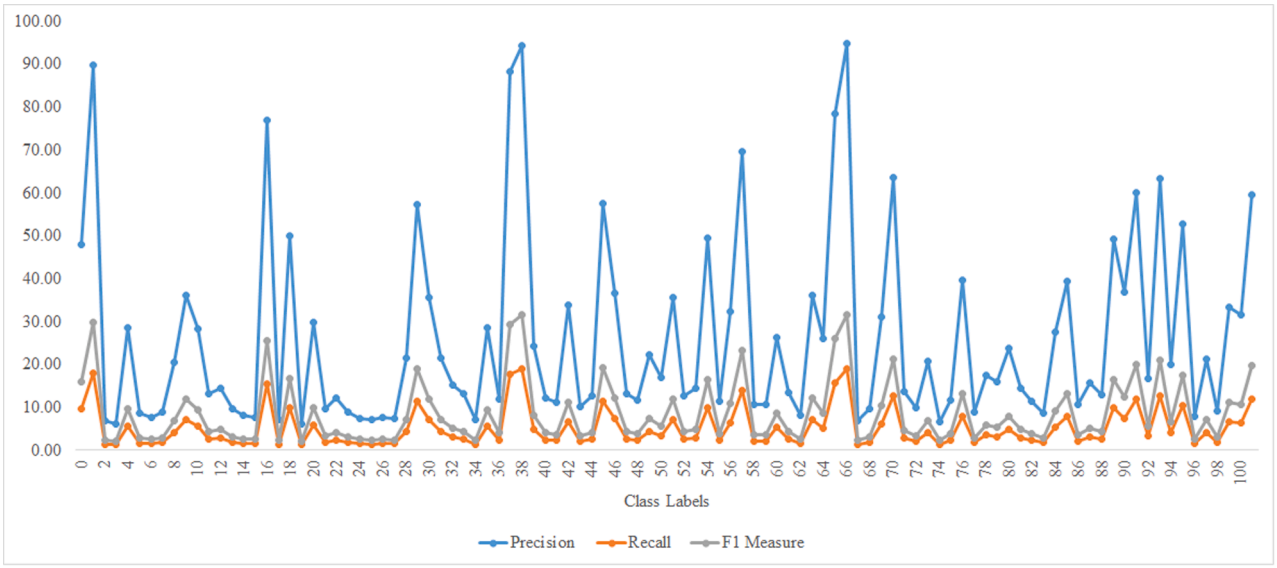

Fig. 16. Image retrieval results for Caltech-101 dataset for TR_N ranging between [10 and 100]. A high number of classes have the total number of images less than 100 leading to slightly lower performance as the TR_N increases.

Table 4
Comparison with state-of-the-art methods $@TR_N = 10$ and $@TR_N = 20$ for Caltech-101 dataset in terms of precision, recall and F1 measure.

Algorithms	$TR_N = 10$			$TR_N = 20$		
	MAP	MAR	F1-Measure	MAP	MAR	F1-Measure
Clustering (Pandey & Khanna, 2016)	–	–	–	35.00	7.00	–
Conv-Deconv + Robust-KSH (Li et al., 2018)	25.7	5.14	–	–	–	–
Conv-Deconv + KSH (Li et al., 2018)	24.1	4.82	–	–	–	–
DPQ-ASym 3-layer + IN (Klein & Wolf, 2019)	–	–	–	42.53	8.51	–
MaxNet (max)	42.80	4.28	7.78	36.27	7.25	12.09
MaxNet (min)	36.19	3.62	6.58	29.91	5.98	9.97
MaxNet (concat)	47.37	4.74	8.61	41.99	8.40	13.40
Proposed MaxNet (add)	49.08	4.91	8.93	44.81	8.96	14.93

6. Conclusion

Deep learning-based models form the basis of search engines that retrieve images similar to the query image from extensive datasets. Accuracy of such systems becomes of prime importance as the user expects to see contextually similar images as the query image. In this paper, a deep learning-based method for image retrieval has been proposed. Model is thoroughly tested on four different datasets and outperforms state-of-the-art methods in the image retrieval domain. The proposed MaxNet model uses an updated inception module that is repeated several times to deepen the network and adds the deep features after each inception in the aggregation step. Model generates satisfactory results using fewer features compared to most recent techniques being used for image retrieval. Very low number of features increases the response time of the system as Euclidean distance between the query image and entire feature database requires fewer computations. Therefore, a realistic system that performs well on different datasets with considerably good response time is presented in this study. The results show that system is very effective in identifying visually similar images and can be used as a retrieval system in a variety of applications. Thus, the system can be used in fields relying on multimedia retrieval, such as search engines, medical history retrieval systems, security systems relying on fingerprints and face detection, crime prevention and detection systems etc. In future, the system will be evaluated on large-scale image datasets such as ImageCLEF and ImageNET, and data augmentation is under consideration for dealing with class imbalance problem in specific datasets such as Caltech-101.

CRediT authorship contribution statement

Saddam Hussain: Conceptualization, Methodology, Software, Writing - original draft. **Muhammad Ahmad Zia:** Data curation, Validation, Visualization. **Waqas Arshad:** Writing - review & editing, Resources, Project administration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Ahmed, K. T., Naqvi, S. A. H., Rehman, A., & Saba, T. (2019). Convolution, approximation and spatial information based object and color signatures for content based image retrieval. In *2019 International conference on computer and information sciences (ICCIS)* (pp. 1–6). IEEE.

Ahmed, K. T., Ummesafi, S., & Iqbal, A. (2019). Content based image retrieval using image features information fusion. *Information Fusion, 51*, 76–99.

Alfanindya, A., Hashim, N. Eswaran, C. (2013). Content based image retrieval and classification using speeded-up robust features (surf) and grouped bag-of-visual-words (gbovw). In 2013 International conference on technology, informatics, management, engineering and environment (pp. 77–82). IEEE.

Anandh, A., Mala, K. & Suganya, S. (2016). Content based image retrieval system based on semantic information using color, texture and shape features. In 2016 International conference on computing technologies and intelligent data engineering (ICCTIDE'16) (pp. 1–8). IEEE.

Ashraf, R., Ahmed, M., Ahmad, U., Habib, M. A., Jabbar, S. & Naseer, K. (2018). Mdcbir-mf: Multimedia data for content-based image retrieval by using multiple features. Multimedia Tools and Applications, 1–27.

Babenko, A. & Lempitsky, V. (2015). Aggregating deep convolutional features for image retrieval. arXiv preprint arXiv:1510.07493.

Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2014). Neural codes for image retrieval. In *European conference on computer vision* (pp. 584–599). Springer.

Bala, A., & Kaur, T. (2016). Local texton xor patterns: A new feature descriptor for content-based image retrieval. *Engineering Science and Technology, an International Journal, 19*(1), 101–112.

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. In *European conference on computer vision* (pp. 404–417). Springer.

Charles, P. (2013). Project title. https://github.com/charlespwd/project-title.

Chen, Y.-H., Chang, C.-C., Lin, C.-C., & Hsu, C.-Y. (2019). Content-based color image retrieval using block truncation coding based on binary ant colony optimization. *Symmetry, 11*(1), 21.

da Silva Torres, R., & Falcao, A. X. (2006). Content-based image retrieval: Theory and applications. *RITA, 13*(2), 161–185.

Dua, D. & Graff, C. (2017). UCI machine learning repository. http://archive.ics.uci.edu/ml.

Eakins, J. P. (2002). Towards intelligent image retrieval. *Pattern Recognition, 35*(1), 3–14.

Fadaei, S., Amirfattahi, R., & Ahmadzadeh, M. R. (2016). New content-based image retrieval system based on optimised integration of dcd, wavelet and curvelet features. *IET Image Processing, 11*(2), 89–98.

Galshetwar, G., Waghmare, L., Gonde, A., & Murala, S. (2019). Local energy oriented pattern for image indexing and retrieval. *Journal of Visual Communication and Image Representation, 64*, Article 102615.

Hussain, S., Anwar, S. M., & Majid, M. (2017). Brain tumor segmentation using cascaded deep convolutional neural network. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 1998–2001). IEEE.

Hussain, S., Anwar, S. M., & Majid, M. (2018). Segmentation of glioma tumors in brain using deep convolutional neural network. *Neurocomputing, 282*, 248–261.

Ioffe, S. & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

Jain, M., & Singh, S. (2011). A survey on: Content based image retrieval systems using clustering techniques for large data sets. *International Journal of Managing Information Technology, 3*(4), 23.

Jegou, H., Perronnin, F., Douze, M., Sánchez, J., Perez, P., & Schmid, C. (2011). Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 34*(9), 1704–1716.

Kalantidis, Y., Mellina, C., & Osindero, S. (2016). Cross-dimensional weighting for aggregated deep convolutional features. In *European conference on computer vision* (pp. 685–701). Springer.

Klein, B., & Wolf, L. (2019). End-to-end supervised product quantization for image search and retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5041–5050).

Kokare, M., Chatterji, B. & Biswas, P. (2003). Comparison of similarity metrics for texture image retrieval. In TENCON 2003. Conference on convergent technologies for Asia-Pacific region (Vol. 2, pp. 571–575). IEEE.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097–1105).

Kumar, K. K., & Gopal, T. V. (2014). A novel approach to self order feature reweighting in cbir to reduce semantic gap using relevance feedback. In *2014 International conference on circuits, power and computing technologies [ICCPCT-2014]* (pp. 1437–1442). IEEE.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.

Li, Y., Wan, M., & Xie, B. (2018). A semi-supervised learning method using deep conv-deconv network and robust-ksh for image retrieval. In *2018 IEEE 8th annual international conference on CYBER technology in automation, control, and intelligent systems (CYBER)* (pp. 640–645). IEEE.

Liu, P., Guo, J.-M., Wu, C.-Y., & Cai, D. (2017). Fusion of deep learning and compressed domain features for content-based image retrieval. *IEEE Transactions on Image Processing, 26*(12), 5706–5717.

Lowe, D. G., et al. (1999). Object recognition from local scale-invariant features. *ICCV, 99*, 1150–1157.

Mansoori, Z., & Jamzad, M. (2009). Content based image retrieval using the knowledge of texture, color and binary tree structure. In *2009 Canadian conference on electrical and computer engineering* (pp. 999–1003). IEEE.

Mehmood, Z., Mahmood, T., & Javid, M. A. (2018). Content-based image retrieval and semantic automatic image annotation based on the weighted average of triangular histograms using support vector machine. *Applied Intelligence, 48*(1), 166–181.

Mistry, Y., Ingole, D., & Ingole, M. (2017). Content based image retrieval using hybrid features and various distance metric. *Journal of Electrical Systems and Information Technology*.

Pandey, S., & Khanna, P. (2016). Content-based image retrieval embedded with agglomerative clustering built on information loss. *Computers & Electrical Engineering, 54*, 506–521.

Peng, T., Boxberg, M., Weichert, W., Navab, N. & Marr, C. (2019). Multi-task learning of a deep k-nearest neighbour network for histopathological image classification and retrieval. bioRxiv, p. 661454.

Rana, S. P., Dey, M., & Siarry, P. (2019). Boosting content based image retrieval performance through integration of parametric & nonparametric approaches. *Journal of Visual Communication and Image Representation, 58*, 205–219.

Sajid, S., Hussain, S., & Sarwar, A. (2019). Brain tumor detection and segmentation in mr images using deep learning. *Arabian Journal for Science and Engineering, 44*(11), 9249–9261.

Saritha, R. R., Paul, V., & Kumar, P. G. (2019). Content based image retrieval using deep learning process. *Cluster Computing, 22*(2), 4187–4200.

Sarwar, A., Mehmood, Z., Saba, T., Qazi, K. A., Adnan, A., & Jamal, H. (2019). A novel method for content-based image retrieval to improve the effectiveness of the bag-of-words model using a support vector machine. *Journal of Information Science, 45*(1), 117–135.

Sharif, U., Mehmood, Z., Mahmood, T., Javid, M. A., Rehman, A., & Saba, T. (2019). Scene analysis and search using local features and support vector machine for effective content-based image retrieval. *Artificial Intelligence Review, 52*(2), 901–925.

Singh, C., Walia, E., & Kaur, K. P. (2018). Color texture description with novel local binary patterns for effective image retrieval. *Pattern Recognition, 76*, 50–68.

Sivic, J., Russell, B. C., Efros, A. A., Zisserman, A., & Freeman, W. T. (2005). Discovering objects and their location in images. In *Tenth IEEE international conference on computer vision (ICCV'05)* (Vol. 1, pp. 370–377). IEEE.

Spyromitros-Xioufis, E., Papadopoulos, S., Kompatsiaris, I. Y., Tsoumakas, G., & Vlahavas, I. (2014). A comprehensive study over vlad and product quantization in large-scale image retrieval. *IEEE Transactions on Multimedia, 16*(6), 1713–1728.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1–9).

Tian, X., Jiao, L., Liu, X., & Zhang, X. (2014). Feature integration of eodh and color-sift: Application to image retrieval based on codebook. *Signal Processing: Image Communication, 29*(4), 530–545.

Tian, X., Zheng, Q., & Xing, J. (2018). Content-based image retrieval system via deep learning method. In *2018 IEEE 3rd advanced information technology, electronic and automation control conference (IAEAC)* (pp. 1257–1261). IEEE.

Tolias, G., Sicre, R. & Jégou, H. (2015). Particular object retrieval with integral max-pooling of cnn activations. arXiv preprint arXiv:1511.05879.

Wangming, X., Jin, W., Xinhai, L., Lei, Z., & Gang, S. (2008). Application of image sift features to the context of cbir. In *2008 International conference on computer science and software engineering* (Vol. 4, pp. 552–555). IEEE.

Wu, L., & Hoi, S. C. (2011). Enhancing bag-of-words models with semantics-preserving metric learning. *IEEE MultiMedia, 18*(1), 24–37.

Xie, B., Qin, J., Xiang, X., Li, H., & Pan, L. (2018). An image retrieval algorithm based on gist and sift features. *IJ Network Security, 20*(4), 609–616.

Xu, H., Huang, C., & Wang, D. (2019). Enhancing semantic image retrieval with limited labeled examples via deep learning. *Knowledge-Based Systems, 163*, 252–266.

Yang, J., Jiang, Y. -G., Hauptmann, A. G. & Ngo, C. -W. (2007). Evaluating bag-of-visual-words representations in scene classification. In Proceedings of the international workshop on Workshop on multimedia information retrieval (pp. 197–206). ACM.

Yu, L., Feng, L., Wang, H., Li, L., Liu, Y., & Liu, S. (2018). Multi-trend binary code descriptor: A novel local texture feature descriptor for image retrieval. *Signal, Image and Video Processing, 12*(2), 247–254.

Zhang, L., Zhang, Y., Gu, X., Tang, J., & Tian, Q. (2014). Scalable similarity search with topology preserving hashing. *IEEE Transactions on Image Processing, 23*(7), 3025–3039.

Zhe, X., Chen, S., & Yan, H. (2019). Directional statistics-based deep metric learning for image classification and retrieval. *Pattern Recognition, 93*, 113–123.