Soft Computing based Artificial Neural Network and Multiple Feature set Intelligence system for Image Retrieval

Venkataravana Nayak K (venkatcsuvce@gmail.com)
Visvesvaraya Technological University
J S Arunalatha
Visvesvaraya Technological University
G U Vasanthakumar
NITTE Meenakshi Institute of Technology
K R Venugopal
Bangalore University

Research Article

Keywords: Artificial Neural Networks, Features Cascade, Image Retrieval, Overlapping Local Binary Pattern, Scaled Conjugate Gradient.

Posted Date: January 17th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1238656/v1

Abstract

This paper proposes an Image Retrieval model using Multiple Feature Sets and Artificial Neural Network (IR-MFS-ANN), where the multiple features Histogram of oriented Gradient (HoG), Overlapping Local Binary Pattern (OLBP), Color and Statistical features are considered. Visual information is one of the most important data in the field of social networking, medicine, military, and these areas contain an enormous volume of semi-organized and organized heterogeneous information related to explicit subjects. However, retrieval and usage of suitable information from the comprehensive information archives are important to meet the content extraction and retrieval challenges. To improve retrieval performance, the image representation, modeling, scalable algorithm that permit accessing large archives are integrated into the retrieval framework. The proposed model utilizes ANN to find the distance between feature vectors. The proposed algorithm is tested and analyzed with various retrieval techniques and it is found that ANN-based image retrieval outperforms the state-of-the-art techniques [1–5]. The proposed method results in accuracy of 94%, 92%, 95%, and 94% for Wang, Cifar-10, Oxford Flower, and ImageNet standard databases respectively.

1 Introduction

Due to many image generating resources and tools such as Skitch, Cloudapp, Easel.ly, Placeit, Smush.it, Recite, Infogr.am, Aviary, *etc.*, for social media and many registered trademarks by the research in various areas, the size of data is continuously growing on a daily basis. Finding needful information from large data sources is a necessary task and requires efficient searching mechanisms for the image data.

Generally, images are retrieved by developing image retrieval systems based on text and image contents. Text-Based Image Retrieval (TBIR) and Content-Based Image Retrieval (CBIR) are the two types of existing systems [1]. The TBIR is carried out by two methods: (i) Retrieval by Annotation: The retrieval is flexibly carried out through the keywords. The images are annotated by text with keywords or descriptors and stored in a database. By formulating numerical or textual queries based on the annotations, images are retrieved with this criterion. The manual process of annotation is difficult in practice. If the volume of databases is large, requires more manpower, hence it is very difficult to provide different descriptors due to the presence of more details in the contents of an image and textual annotations are language-dependent *i.e.*, the retrieval process should be universal. Human perception is subjective; if humans have biases it is difficult to get the correct query with the inclusion of labels. (ii) Retrieval by Semi-Automated Annotation: If the images have labeled regions, recognition of new and rare phenomena leads to a challenge. The process of describing a query that represents visual features with the words is difficult. The query representation of a visual feature *i.e.*, annotations does not support abstract concepts.

Content-Based Image Retrieval (CBIR) is a common task of finding images that meet relevant descriptions with image content instead of metadata. The retrieval is flexibly carried out through image features. It permits each image to be described by its own features and takes the task of forming queries away from the end-user. The retrieval of the whole image is explored earlier but selectivity is a problem.

Due to descriptors dimensionality [2], indexing is not used. The lookup tables computation, region describing method do not fit the dynamic dataset.

Need to produce feature set with the dimensionality constraint, use of efficient similarity matching method for the retrieval leads to a purpose of developing a model with multiple extracted features along with ANN to find the similarity. The proposed method addresses the image selectivity and dimensionality problem. Objective: To retrieve images accurately from the dataset through the discriminative visual features extraction by creating distinct and strong features for obtaining reliable results with the machine learning approach.

Contribution: The HoG and OLBP descriptors are used along with the color and statistical features as the basis to extract features. This increases the number of features and improves the performance and serves as elementary data for decreasing feature dimensionality. The extracted features are cascaded and stored in a feature vector, which helps to utilize feature information efficiently in similarity computation which in turn helps in enhancing model accuracy and reducing model retrieval time [1], [2], [3], [4], and [5]. Organization: This paper is organized as follows: A brief description of Related Works on image retrieval is presented in section 2. In section 3, an explanation of the Background work is presented. The Proposed Method and Algorithm are described in section 4. The Results and Discussions analysis is depicted in section 5. Section 6 presents the Conclusions.

Table I: Related Works

Author	Concept	Algorithm	Merits	Demerits	Performance
Ren *et al.*, [2017]	Image Recognition	Multi-scale Overlapped Block LBP descriptor	Compact feature representation	The precision of the model is less	51.13% Average Precision
Wu *et al.*, [2020]	Image Recognition	Back Propagation NN	Image format is Independent	Consumes time in the retrieval process	97.26% Accuracy
Takala *et al.*, [2005]	Image Retrieval	Local and Global features, Minkowski metric	Obtained descriptive details	Performance analysis is complex	45.3% Precision
Liao *et al.*, [2020]	Image Retrieval	GLCM with Artificial Bee Colony Algorithm, Squared Euclidean distance metric	Convergence is fast with the control parameters	Inadequate in the fulfilment of accuracy requirement	76% Precision
Zhang *et al.*, [2020]	Image Retrieval	K-mean Clustering and Nearest Neighbour	Provides better performance for the small number of samples	Less scalability	93.65% Accuracy
Mahale *et al.*, [2018]	Emotion Identification	Artificial Information Integration algorithm	Requirement for prior knowledge of objects is low	Consumes time in information processing, Learning time is more	90% Accuracy
Firuzi *et al.*, [2020]	Image Matching	Multi-Source Image Matching and Bilateral Matching methods	Eliminates incorrect matches	Optimization is difficult	65.1% Recall Rate
Singh *et al.*, [2019]	Image Identification	DW, HoG, LTE, GLCM and Tamura features, SVM, k-NN, logistic regression and LDC	Reduces features dimensions	Takes time in the retrieval task	98.9% Accuracy
Khaladi *et al.*, [2019]	Image Representation	Multi-Texton Histogram	Takes less retrieval time	Does not eliminates the incorrect matches	22.5% Precision

Author	Concept	Algorithm	Merits	Demerits	Performance
Khan *et al.*, [2020]	Image Matching	Texture Descriptor algorithm, Overlapped Multi-oriented and Triscale Local Binary Pattern approach	Decreases the effects of artefacts	Optimization is difficult	65.1% Recall Rate
Wu *et al.*, [2017]	Image Reconstruction	Image Quality Enhancement	Eliminates the effects of artefacts, Preserves image structures	Domain dependent	86.3% Accuracy
Galshetwar *et al.*, [2018]	Image Retrieval	Feature Learning	Preserves image structures	Not considered complete microscopic and directional information,	64.81% Accuracy

2 Related Works

This section describes the momentary facts on the work related to image retrieval.

Ren *et al.*, [6] presented a Multi-Scale Overlapped Block LBP descriptor to recognize the leaf of a plant. The histograms of LBP are extracted by dividing the image into many overlapping blocks of equal size and the dimension of the LBP feature is reduced with Principal Component Analysis and Support Vector Machine(SVM) classifier in recognition of leaf. Each cell of an overlapped block contributed more than once to the final descriptor; thus the descriptor gives compact, discriminate information to represent leaf and helps in increasing the performance of recognition.

Wu *et al.*, [7] devised a method to identify and locate the tampered image. The color components are extracted by transforming the image to the color space with mean deviation and divide the image vertically into overlapping blocks [8], then applied the two-scale LBP with Discrete Cosine Transform to compute dispersion degree to classify tamper image using a backpropagation NN. The vertical division of an image into overlapping blocks generally helps in capturing more pixels compared to horizontal; this results in increasing detection accuracy of tampered images. The method is not dependent on the format of the image and takes more time for retrieval.

Takala *et al.*, [9] presented two approaches for CBIR, in the first method the image is divided into equal size blocks, computed histograms used Minkowski distance to measure similarity, and compared the histogram of each block with the query image. Effective in the retrieval of a single structure and more

objects in an image. Reduces the size of the image dataset but descriptive power is less because of difficulty in noticing low-level features and not enough details are present in the case of global features. The second method divides query images into many blocks of arbitrary size, then computed the histograms individually and compared with the query image. Useful in obtaining descriptive details of low-level features, but hard to analyze performance due to the involvement of the user in processing more number of sub-blocks.

Liao *et al.*, [10] devised LBP with a multi-scale blocks approach for face identification. The average of the subblocks pixels are taken into consideration instead of individual pixels and similarity is measured by taking histogram bins absolute difference. An AdaBoost classifier is used for identification. It is effective in encoding image patterns of micro/macro structures and fast in feature extraction and the cost is more due to the block division.

Zhang *et al.*, [11] to detect a face, presented a feature descriptor - Multi-block Local Binary Patterns. The multi-branch regression tree is used in AdaBoost learning to reduce computation cost and takes less training time with the reduced feature set. Mahale *et al.*, [12] introduced a framework for detecting inconsistency in the image using block a division strategy. Divided the image into blocks of equal size, Then the LBP value for each block is computed and added all the block values. The similarity amongst the blocks is computed with the Euclidean Distance metric to find similarity to check the image inconsistency. Firuzi *et al.*, [13] presented an approach for monitoring power transformer conditions using high-level image features extraction with LBP and HoG descriptors. The image is pre-processed for taking care of noise, resolution, phase shift, and the SVM to classify images. Chaurasia *et al.*, [14] presented a search methodology for image compression with feature extraction. The time complexity has been reduced by considering less number of image blocks for comparison. The time consumption for the process is independent of the image complexity and can be further reduced for the multi-modal features.

Singh *et al.*, [15] proposed an approach for the detection of fruit conditions. The fruit texture features are extracted with HoG, Discrete Wavelet (DW), Laws Texture Energy (LTE), Gray Level Co-occurrence Matrix (GLCM), Tamura features, and then SVM classifier is used for monitoring the condition of the object. It provides accurate accuracy but takes more time for retrieval. Khaldi et al., [16] introduced descriptors to enhance the discriminative ability of an image. It extracts details of color features, edge gradients and considers various categories of textons, and represents texture and heterogeneous color images appropriately. Retrieval time is less but performance is average.

Khan *et al.*, [17] presented a texture descriptor that imparts features such as rotational invariant, geometrical transformations by capturing macro and microstructures in the texture. The overlapped multi-oriented and multiscale fusion are considered in Overlapped Multi-oriented Triscale Local Binary Pattern approach. The descriptor is useful in various texture-based applications. Wu *et al.*, [18] introduced an approach to enhance the quality of the reconstructed images by exploring the complex structure of an image with the help of an artificial neural network. The incidence of noise is reduced and structure is

preserved by extracting overlapped patches at random locations for each iteration in an image and is domain-dependent.

Galshetwar *et al.*, [19] devised a feature learning approach using an Artificial Neural Network for image retrieval. Using gradient computation, the intensity variations at four directions are calculated. The retrieval accuracy of CBIR is enhanced by taking into account microscopic and directional information. Arunalatha *et al.*, [20] presented Iris Recognition using Hybrid Domain Features (IRHDF). The frequency and spatial image features are extracted and combined to obtain a feature set, then similarity match is computed using Euclidean Distance for the iris detection.

The summary of the related works is given in Table 1. Through the previous studies perceived that the features combination selection leads to less representative and consumes more time in the retrieval task. To overcome, the proposed IR-MFS-ANN model is developed towards accurate performance.

3 Background And Motivation

Motivation

A huge number of digital images are captured through technically advanced digital devices for analysis perspective and deposited in a database. In the case of retrieval applications such as the CBIR technique, the image multi-modal visual features are extracted to improve the system performance. An increase in the number of features extracted helps to find accurate matches which increases retrieval performance but consumption of less time in the matching process reduces the retrieval time. Thus, the dimensionality of the features has to be reduced in order to reduce the matching process time. Hence, there is a need for appropriate feature extraction, dimensionality reduction, and similarity matching mechanisms for developing retrieval models. The model shows the improvement in the image selectivity and keeps descriptor vector constraint to a size that the current similarity finding algorithm capable of handling.

3.1 Optimization

It is an iterative process of minimizing error function estimate in a local neighborhood of the current point in Euclidean space.

The basic steps involved in all the methods are: (i) Starting with a preliminary point (ii) Determine search direction with a particular assumption (ii) move in a search direction that gradually minimizes objective function (iv) at a specific point, a new search direction is discovered and this procedure is repetitive. But the difference lies in the selection of successive search directions. The Taylor expansion of 1st or 2nd order is used in the calculation. The steps followed in minimizing the error function are shown in Algorithm 1.

Algorithm 1: Optimization process

1: Select W_1 -initial weight vector and assign for step$k = 1$

2: Find $\tilde{d}_k$ a search direction and α_k a step size, so that$E(\tilde{w}_k + \alpha_k \tilde{d}_k) cript >$

3: Update Weight Vector $i.\ e.\ \tilde{w}_{k+1} = \tilde{w}_k + \alpha_k \tilde{d}_k$

4: If $E'\left(\tilde{w}_k\right) \neq \tilde{0}$ then assign $k = k + 1$ and go to step (2) otherwise return $\tilde{w}_{k+1}$ as the chosen is minimum

where, $W_i -$ is weight vector, $d_k -$ is search direction, $k -$ represents the number of steps and E is error function. Finding the next current-point in the process, consists of two steps, (i) find search direction: direction in Euclidean space to follow, (ii) find step size: the direction distance in search space.

In the 2nd step of algorithm 1, assigning negative gradient to search direction and constant to step size, the algorithm resembles Gradient Descent Method (GDM)[21], it is a Backpropagation (BP) method without the parameter momentum [22]. The gradient descent is minimized by linear approximation.

The important complexities of optimization methods are 1. Need to store conjugate vectors, complex for large systems. 2. The solution relies on round-off errors 3. At each step, the length of the error vector decreases, the error function also decreases but the squared residual may increase 4. No formula for computing directions.

Several optimization algorithms were developed based on the GD method [23], it depends on the user-specified parameters, so poor in convergence rate. The reasons for poor results are specified as 1). The Gradient Descent is minimized by linear approximation; it leads to poor convergence. 2). Constant step size- is inefficient and makes the algorithm weak. 3). Momentum term – inclusion makes use of 2nd order derivative information but its speed is less, this could make the execution of the algorithm weak. 4. Momentum constant- user-dependent parameter, this constant term introduced in two ways based on weights update: (a) offline: to update weights, considered all patterns information of the training set. (b) online: to update weights, considered single patterns information of the training set, but shows less performance towards several applications compared to offline due to compatibility of the theory of optimization. The reason is the presence of redundant information in the large-sized datasets [24] and the uneven distribution of information (information sparsity).

3.2 Conjugate Directions (CD)

CD method also based on general optimization approach, employed to handle complex systems using direct methods, selects step size and search direction through the second-order estimate. Because of the advantage of quadratic equations, crucial points are computed as a solution for the linear system. With conjugates, search directions linear combination is used from the initial point to crucial point. The crucial points computed are in N epochs. The crucial point may be a maximum or saddle point, not necessarily minima [25]. The Hessian matrix assumed positive definite only if the Hessian matrix is positive definite,

then quadratic estimate to the error function in weight space has a unique global minimum. In the iterative process, the inner product is minima for quadratic estimate error function intermediate. The Hessian matrix eigenvalues fall into multi groups of the same size, then the algorithm probably terminates within much less than N epochs. The global minimum is computed with the algorithm to a quadratic function in a maximum of N epochs.

The Hessian matrix assumed positive definite, only if the Hessian matrix is positive definite then quadratic estimate to the error function in weight space has a unique global minimum. In the iterative process, the inner product is minima for quadratic estimate error function intermediate. The Hessian matrix eigenvalues fall into multi groups of the same size, then the algorithm probably terminates within much less than N epochs. The global minimum is computed with the algorithm to a quadratic function in a maximum of N epochs.

3.3 Conjugate Gradients (CG)

It is a conjugate direction approach, the directions are computed sequentially in each step of the iteration process and are based on gradients so it makes a uniform movement towards the solution in each step.

The strategy takes conjugate system as input and finds the next conjugate gradient *i.e.*, $\tilde{d}_1$, $\tilde{d}_2$, $\tilde{d}_3$,..... ..... $\tilde{d}_N$. In each of the iteration within the weight space of the current-point $\tilde{w}$, CG strategy is applied to the quadratic approximation E_{qw}, of the global error function E. Due to the nonquadratic nature of the error function $E\left(\tilde{w}\right)$, the algorithm in N steps does not necessarily converge. If it does not restart after N steps, the values are initialized *i.e.*, initialized $\tilde{d}_{k+1}$ to $\tilde{r}_{k+1}$, residuals are computed [26]. It is valid only if $E = E_{qw}$, thus it converging fast.

The steps involved in the Conjugate Gradient method are shown in Algorithm 2.

Algorithm 2: Conjugate Gradient (CG) Method

1: Select W_1 weight vector and set $\tilde{d}_1 = \tilde{r}_1 = -E'\left(\tilde{w}_1\right), k = 1$

2: Computer 2^{nd} order information, $\tilde{S}_k = E''\left(\tilde{w}_k\right)\tilde{d}_k, \delta_k = \tilde{d}_k^T\tilde{S}_k$

3: Compute step size, $\mu_k = \tilde{d}_k^T\tilde{r}_k, \alpha_k = \frac{\mu_k}{\delta_k}$

4: Update weight vector, $\tilde{w}_{k+1} = \tilde{w}_k + \alpha_k\tilde{d}_k, \tilde{r}_{k+1} = -E'\left(\tilde{w}_{k+1}\right)$

5: if $k \bmod N = 0$ then repeat from step1, $\tilde{d}_{k+1} = \tilde{r}_{k+1}$ otherwise, $\beta_k = \frac{(|\tilde{r}_{k+1}^2|) - \tilde{r}_{k+1}^T\tilde{r}_k}{\mu_k}$, $\tilde{d}_{k+1} = \tilde{r}_{k+1} + \beta_k\tilde{d}_k$

6: if steepest descent direction $\tilde{r}_k \neq \tilde{0}$ then set k=k+1 go to step 2 otherwise stop, return $\tilde{w}_{k+1}$ as minimum desired

Where, $\tilde{d}_k -$ is search direction, $\tilde{d}_{k+1} -$ is new search direction, $\tilde{r}_k -$ is residual vector, $\tilde{r}_{k+1} -$ is new residual vector, $- E' \left(\tilde{w}_k \right) -$ is negative gradient, $E'' \left(\tilde{w}_k \right) -$ is Hessian matrix positive definite, $\alpha_k -$ is step length, $\mu_k -$ is step size, $\delta_k -$ is direction distance, $\beta_k -$ is improvement in direction.

3.4 Scaled Conjugate Gradients (SCG)

It is used in the learning process, uses the Levenberg-Marquardt approach [27] for computing the step size. The steps involved in the process are depicted in Algorithm 3.

Algorithm 3: Scaled Conjugate Gradient (SCG) Algorithm

1: Select W_1 weight vector and scalars - σ, λ_1, $\tilde{\lambda}_1$ $i.e.,$ $0cript >$, $0cript >$ and

Set $\tilde{d}_1 = \tilde{r}_1 = - E' \left(\tilde{w}_1 \right)$, $k = 1$ and condition=true

2: if condition = true then compute the 2^{nd} order information $i.e.,$ $\sigma_k = \frac{\sigma}{\left| \tilde{d}_k \right|}$,

$\tilde{S}_k = \frac{(E' (\tilde{w}_k) + \sigma_k \tilde{d}_k) - E' \left(\tilde{w}_k \right))}{\sigma_k}$, $\delta_k = \tilde{d}_k{}^T \tilde{S}_k$

3: Scale δ_k, $\delta_k = \delta_k + (\lambda_k - \tilde{\lambda}_k) | \tilde{d}_k |^2$

4: if $\delta_k \leq 0$ then do-Hessian matrix positive definite, $\tilde{\lambda}_k = 2(\lambda_k - \frac{\delta_k}{| \tilde{d}_k |^2}$, $\delta_k = -\delta_k + \frac{\lambda_k}{| \tilde{d}_k |^2}$, $\lambda_k = \tilde{\lambda}_k$

5: Compute step size, $\mu_k = \tilde{d}_k{}^T \tilde{r}_k, \alpha_k = \frac{\mu_k}{\delta_k}$

6: Compute Comparison parameter,

$$\Delta_k = \frac{2\delta_k [(E\left(\tilde{w}_k \right) - E(\tilde{w}_k + \alpha_k \tilde{d}_k))]}{\mu_k^2}$$

7: if $\Delta_k \geq 0$ then reduction in error has done successfully, $\tilde{w}_{k+1} = \tilde{w}_k + \alpha_k \tilde{d}_k$,

$\tilde{r}_{k+1} = -E'(\tilde{w}_{k+1})$, $\tilde{\lambda}_k = 0$, condition=true

if $k \bmod N = 0$ then repeat from step1,

$\tilde{d}_{k+1} = \tilde{r}_{k+1}$ else $\beta_k = \frac{|\tilde{r}_{k+1}|^2 - \tilde{r}_{k+1}{}^T \tilde{r}_k}{\mu_k}$, $\tilde{d}_{k+1} = \tilde{r}_{k+1} + \beta_k \tilde{d}_k$

8: if $\Delta_k \geq 0.75$, then reduce scale parameter, $\tilde{\lambda}_k = \frac{1}{4}\lambda_k$,

condition=false end if if $\Delta_k \, cript > \ 0.25$, then increment scale parameter,

$$\lambda_k = \lambda_k + \left(\frac{\delta_k (1 - \Delta_k)}{|\tilde{d}_k|^2} \right)$$

9: if steepest descent direction $\tilde{r}_k \neq \tilde{0}$ then set k=k+1 go to step 2 otherwise stop, return $\tilde{w}_{k+1}$ as minimum desired

The algorithm presents a scalar λ_k for regulating Hessian matrix indefiniteness to overcome the complexity with it. It adjusts λ_k using a constant factor in every iteration with reference to the δ_k sign assistance to ensure Hessian matrix is not positive definite by this process repetitiveness. To obtain the optimal solution in the case of Hessian matrix positive definite, a comparison parameter is used. It is a measure of the quadratic estimate of the error function, for the better approximation it should be nearer to 1.

CG method is used in the training process, suits for large scale problems, relevant and alternative to learning algorithms. The parameters that can be eliminated includes user-dependent parameters [21], [25], [26], [28]. The CGM introduced by (CGL) [29], [30]: Theoretically described to employ feed-forward NN and it is faster compared to BP algorithm. Battiti and Masulli [31], have introduced a variation of the standard conjugate gradient method, the one-step Broyden-Fletcher-Goldfarb-Shanno memoryless quasiNewton algorithm (BFGS), as an alternative learning algorithm. It works faster than BP. But computation complexity is high, Both CGL and BFGS to compute step size with line search contains many computations (a) global error function or (b) derivatives of it, in each iteration of learning which leads to complexity.

CG fails, due to error function nonquadratic nature and thus works with functions of Hessian positive definite. Gives poor convergence with respect to quadratic approximations. For the error function E, the Hessian matrix show indefinite at various locations in the weight space. So to regulate Hessian definite indefiniteness, introduced scalar factor based on L-M approach in CG called Scaled Conjugate Gradient method. This approach works on the basis of the model trust region strategy. Scaled Conjugate Gradients (SCG) uses the Levenberg-Marquardt approach [27] for computing the step size.

4 Proposed Method

The architecture of an ANN-based CBIR model - IR-MFS-ANN is depicted in Fig. 1. The proposed framework has four phases namely Preprocessing, Feature Extraction, Training, and Testing, or Image Retrieval. At the initial phase, the images from the databases are imported and they are resized to 32 $*$ 32, forwarded to the feature extraction phase. In the feature extraction phase, extracted the features of images and are cascaded and stored in the matrix. In the training phase, the Artificial Neural Network (ANN) is trained using extracted features. In the training phase initially, the query image is pre-processed and then the features are extracted and cascaded, then the trained ANN is tested for the query image. Finally, the system performance is evaluated in the form of accuracy.

The steps followed in evaluating retrieval performance are shown in Algorithm 4. The appropriate features extracted are fused to make it more discriminative and stored in the feature vector. Using ANN, the features are trained in order to make the parameters that exist in the training data are distributed properly in the test data and the final data in order to apply the algorithm. The similarity matching is computed to determine the accuracy of the retrieval process.

Algorithm 4: Image Retrieval

Input: Database images

Output: Retrieval Accuracy

1: Extraction of color, texture, statistical features from the database images with features descriptors

2: Feature cascading

3: Construction of Feature Vector from the extracted features

4: Training of features with ANN

5: Compute similarity match through the feature

vectors comparison of query and database images

6. Evaluate Performance

4.1 Feature Extraction

The feature extraction step is crucial because the proposed model performance relay on the features of training data in the image retrieval process. The Machine Learning strategies emphasize discriminative models for retrieval tasks.

4.1.1 Texture Features

• **Overlapping LBP features**:

In the proposed method, the fixed-size blocks division approach is used for addressing the spatial properties of images. The image is divided into sub-images of fixed-size blocks and these are overlapped [9]. The histograms for each of the image sub-blocks are drawn *i.e.*, sub-histograms, and the LBP operator is employed for these overlapped blocks to compute the distribution of local binary patterns, then formed a single histogram of an image representation by combining each of these sub-histograms. The mathematical equation for the division and histogram formation of an image is represented with Equation (1) and Equation (2).

$$I - Block = B_1 + B_2 + \ldots . + B_n = \sum_{n=1}^{N} I - block_i$$

1

$$I - Hist = H_1 + H_2 + \ldots . + H_n = \sum_{n=1}^{N} I - hist_i$$

2

To obtain Overlapping LBP features, the process is continued for all the database images *i.e.* for n images, n LBP operators are used and obtained n LBP codes which are combined to form a vector of nLBP codes with Equation (3).

$$nLBP = \sum_{n=1}^{N} LBP\left(I^N\right)$$

3

The LBP operator is frequently used for greyscale images in generating binary code for each pixel. In the proposed method, the LBP operator is used for a color image. Where, initially red, green, and blue components are individually extracted and stored in a matrix after employing LBP to color constituents distinctly. For *e.g.*, 3∗3 pixels cell, starting from an upper left pixel in a clockwise direction, the centre pixel compared with and subtracted from neighbouring pixels. The corresponding cell will be assigned with a value less than or equal to 0 or 1. The value obtained is a binary number taken into account in a clockwise direction from the upper left corner of all cells. The decimal value is computed by considering weights in the corresponding cells [13], [32] is shown in Fig. 2. The procedure of computing the LBP code using Equation (4) is shown in Fig. 3.

$$LBP_{p,r} = \sum_{p=0}^{p-1} Th(N_p - C_c)2^p$$

4

where, N_p – neighbourhood pixels in each block is thresholded by C_c - center pixel, p is sampling points, R – radius for $3 * 3$ cell *i.e.*, 1, coordinates of C_c is $(0,0)$ and $N_p = x + R\cos\frac{2\pi p}{P}, y - R\sin\frac{2\pi p}{P}$ binary threshold function.

$$Th(v) = \begin{cases} 0, & if x \leq 0 \\ 1, & if x \geq 0 \end{cases}$$

5

Even though LBP is computationally simple, limited in structural information capture, uses the only difference in pixels, ignore magnitude information. Increases computation complexity with respect to time and space as the exponential increase in the size of features with the neighbor number. it is not invariant to rotations [33]. In some cases, the distribution of features is not uniform, it is accounted in terms of a number of transitions between 0 and 1 or 1 and 0 represented as t. The number of transitions between 0 and 1 or 1 and 0 *i.e.*, t is irregular, so the distribution of features is not uniform. Hence, for uniformity measure, t is considered as less than or equal to 2. In mapping of uniform LBP, for every uniform pattern separate label is assigned and for every non-uniform pattern assigned a single label (neglected) by considering assumption – in natural images, many of the LBPs are uniform (90%) [33]. Uniform LBP gives comparatively better performance by reducing the dimensionality of the features, but it has no rotation invariant. This issue can be repressed with the use of rotated LBP *i.e.*, RLBP.

Computing dominant direction in the neighborhood through shifting weights of LBP operator circularly and using the difference of magnitude. The dominant direction is nothing but the maximum difference between neighboring pixels from the central pixel and it is considered as a reference.

4.1.2 Color Features

A significant direct component that individuals see as soon as seeing the color image. The Vision perception of humans is with time gentle to color information compared to indistinct proportions. Thus, it is the crucial opponent used for feature extraction. To describe color substance, the usual strategy used is Color histogram. In the proposed model RGB and LAB color features for image retrieval are used.

RGB Feature

RGB is a color space, generally familiar for images ever since the digital systems started using the necessary color components (Red, Green, Blue) for showing any seeming color. Each pixel within a screen is made by the essential components and animated through the color primaries of the electron gun individually [34]. Moreover, perceptually the RGB color space is not uniform, so separation of color in it does not relate to a color discrepancy in observation. With respect to this, changing information of an

image in it to the other perceptual constant space before feature extraction. The image RGB constituents are extracted distinctly in the proposed method.

LAB Features

It is a color space and device-independent, to specify a color uses a three-axis system (L - luminance axis, a - green to the red axis, and b - blue to yellow). The Human Vision system recognizes this continuous uniform change in the color distribution due to consistent dispersion along the axis [35]. The conversion of RGB image to LAB in the color space helps to extract image global features.

4.1.3 Histogram oriented Gradient (HoG)

HoG descriptors represent images with the help of very important details of an image. It breaks down the complete image into smaller portions (localized regions) and for each portion, it calculates gradient and orientation. Finally generates histograms for all the regions and counts the gradient orientation occurrences [13].

• **Block Normalization**: L_1 normalization is the summation of the absolute values of the vector. The non-normalized vector v consists of all histograms in a block $||v||_k$ is vector *k*-norm (*k = 0, 1, 2*) and *e* is a small constant. The normalization factor is one among Equations (6), (7), and (8).

$$L_2 norm: f = \frac{v}{\sqrt{||v||_2^2 + e^2}}$$

6

$$L_1 norm: f = \frac{v}{||v||_1 + e}$$

7

$$L_1 sqt: f = \sqrt{\frac{v}{(||v||_1 + e)}}$$

8

V - is a vector, $||v||_k$ - vector size with its norm k

4.1.4 Statistical Features

- **Mean (Averaging filter)**: The individual pixel intensity contribution for the whole image is the mean value, works on a sliding window size of m*n through finding the normal of all pixels values within the window and replacing center pixel value in the main image along with the resulted value. Mathematically Mean is given by Equation (9)

f(x,y)=\frac{1}{mn}\sum _{(r,c)\epsilon W}\text{}g(r,c)
9
Here, g(r,c) is a noisy image, W Sliding Window, c and r stands column and rows.

- **Variance**: Computes variation of each pixel from the centre pixel to divide image pixel region in finding out sharp details such as edges [36]. Mathematically variance is given by Equation (10)

\tilde{f}(x,y)=\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}{\left(g(r,c)-\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}\left(g\right(r,c)\right)}^{2}
10
Here, \overline{f}(i,j) is mean of the image block.

- **Standard Deviation (SD)**: It specifies the presence of pixel deviations from the mean [36]. Mathematically SD is given by Equation (11)

\tilde{f}(x,y)=\sqrt{\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}{\left(g(r,c)-\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}\left(g\right(r,c)\right)}^{2}}
11

- **Kurtosis**: Specifies the distribution of frequency in noise reduction after the removal of low frequencies in case of separating the image from the noise and its contents. It is a [37] measure of the probability distribution shape of a random variable [36]. Mathematically kurtosis is given by Equation (12)

\tilde{f}(x,y)=\frac{\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}\left(\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}g\right(r,c)-\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}g(r,c){)}^{4}}{\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}\left(\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}g\right(r,c)-\frac{1}{mn-1}\sum _{(r,c)\epsilon W}\text{}g(r,c){)}^{2}}
12

4.2 Artificial Neural Network (ANN)

ANN resembles a biological neural network, it has different elements called neurons that operate in parallel. only if the sum of multiplication of inputs, and weights with the addition of bias is positive at the node then ANN discharges energy to the next element otherwise energy will not be discharged. In ANN, all the nodes are interconnected with the adjacent nodes. Here, the input to a node is the weighted sum of the outcome of all the other nodes [38]. The weight Wj is the multiplicative weighing factor among the input node i and output node j.

ANN is a non-linear and adaptive system that learns to perform functions from data. ANN is said to be adaptive because it can change its parameters when the parameters of the system change in its training phase. After training the parameters of the network are fixed then this network is used to solve the problems. There are two modes of operation one is feed-forward mode and another one is

backpropagation mode. In the proposed method, implemented ANN has one input and one output layer with one or two hidden layers. In the backpropagation method, the error is computed by comparing output every time with the target. The weight of the ANN is adjusted to reduce the error every time when the weight changes and the neural model get closer to the targeted output.

In the proposed system, a conjugate gradient backpropagation algorithm is used as a training method and a Scaled Conjugate Gradient (SCG) method as a learning method [39]. Because of the effortlessness of their cycle, numerical productivity, and low memory necessities, SCGs are used as learning methods. The steps of SCG are shown in algorithm 3.

For feedforward neural networks, SCG is a supervised-learning algorithm and member of the class of conjugate gradient methods, highly-automated, does not consist of critical user-dependent parameter, sidesteps a time-consuming line search, which CGL and BFGS use in each iteration to find an appropriate step size.

4.3 Artificial Neural Network (ANN) Training

In the proposed system ANN is used as a classifier, initially, the ANN is trained for classification. A special method is used, where initially two sets of features are selected randomly and they are cascaded one by one, later a target data is created by comparing two sets of features. If both of the features are from the same class then minimum distance ‘0’ is assigned else maximum distance ‘1’ is assigned. Then the cascaded features are given as input data to ANN, and the obtained data is considered as target data. In the proposed method, a total of 1000 images are used for the training and they belong to 10 different classes. Fig. 4 shows the ANN training procedure.

In the first row of Fig. 4, the feature number 36 indicates that the image belongs to the 3rd class and 64 indicates that the image belongs to the 6th class as the two elements belong to different classes and the target is assigned to 1. In the 4th row, feature numbers show that they belong to the same class so the target is assigned to ‘0’. Each of the features used for training has a size of 1*6657. So totally the training data has the size of 100*13314 and the target has the size of 100 *1 samples.

5 Results And Discussions

An ANN-based CBIR model - IR-MFS-ANN is executed in MATLAB with an i5 processor of 4 GB RAM. Three publicly available databases are used to measure the model performance with respect to the accuracy, retrieval time, accuracy with respect to a varying number of features. The method is analyzed, compared to the prevailing methods.

5.1 Datasets

In the proposed framework three different databases are used, namely WANG, CIFAR-10 and OXFORD FLOWERS [1] for image retrieval each of the databases have colored photographs of different objects and they are classified into different classes.

- **WANG**: It has 10 classes each of the class has 100 images totally it has 1000 images [1] [40] with a dimension of 384x256 among 1000 images; 900 are used in training and 100 are for testing.
- **Oxford Flower**: It contains images of flowers that are commonly present in the U.K. It has 17 classes and each class has 80 images, a total of 1360 images [1] [41], among 1360 images; 1020 are used in training and 340 for testing.
- **CIFAR-10** : It contain totally 6000 images[1] [42] with dimension 32x32. It is categorized into 10 classes each class has 6000 images, among 6000 images; 5000 are used in training and 1000 for testing. Each image accompanies a "fine" name (the class to which it has a place) and a "coarse" name (the superclass to which it has a place).
- **ImageNet**: It contains 14,197,122 images [1] used in classification and retrieval of multi-class images. We have considered 3 classes, a total of 8000 images with the dimension of $384 * 256$; 6000 are used in the process of training and 2000 for testing.

5.2 Performance Analysis

5.2.1 Training and Validation Performance

The target data is divided in to three groups *i.e.*, training - 70%, validation - 15% and testing - 15%.

(a) The Training Performance: The input to the network is 12960 features, the number of hidden units is 2, the output unit is one. The training of network takes place with respect to the CG Method. The values of weights and biases are tuned with respect to the training parameters of it. The training parameters are shown in the Table 2. The record about the performance of training, validation and test is maintained. The training process stops by following the specified conditions: upon reaching the maximum iterations, exceeds maximum specified time, minimising performance to target, gradient performance reaches under minimum specified gradient value, number of failures go beyond specified.

Table 2
Conjugate Gradient method parameters

Parameters	Value
Number of epochs	1000
Time in seconds	infinity
Gradient	1e-6
Sigma	5.0e-5
Lambda	5.0e-7
Number of failures	6
Performance	infinite

Training Performance plot

The best training performance of 1.40002e-07 is obtained in the iteration. The graph in Fig. 5, shows the training record error value *v/s* the number of training epochs. In the training process, the error decreases after a specific number of epochs. On the validation set, the error may increment, once NN begins overfitting the training data. The best performance is taken from the epoch when the validation error is low.

Training state plot

The training state values from the record of training are shown in Fig. 6. The gradient value at epoch is 9.8743e-07, training time is 28 seconds, performance is 1.40e-07, the number of validation checks is 0.

(b) The Validation Performance: The SCG method is used for the learning process. The network learns with the SCG optimization, the parameter values are obtained from the record of training and tuned with respect to the parameters of SCG as shown in the Table 3. The NN stops learning, if the performance on validation vectors does not increment or same value for maximum number of failures *i.e.*, failure of validation vectors. To check the NN generalization, the test vectors are used.

The learning process stops by following these specified conditions: upon reaching the maximum iterations, exceeds maximum specified time, minimising performance to target, gradient performance reaches under minimum specified gradient value, number of failures go beyond specified, \mu value go beyond its specified maximum value.

Table 3
Scaled Conjugate Gradient method parameters

Parameters	Value
Number of epochs	1000
Time in seconds	infinity
Gradient	1e-6
Sigma	5.0e-5
Lambda	5.0e-7
Number of failures	6
Performance	infinite
Initial mu	0.001
Mu decrement	0.1
Mu increment	10
Mu Max	1e10

Validation Performance plot

The best validation performance of 23.2352ms is obtained in the iteration. The graph of in Fig. 7, shows Mean Square Error value *v/s* the number of epochs. In the training process, the error decreases after a specific number of epochs, but over validation set error may increment, once NN begins overfitting the training data. The best performance is taken from the epoch when the validation error is low.

Validation state plot

The validated values from the record of training are shown in Fig. 8. The gradient value at epoch is 29.5, value is 0.01, training time is 1 second, performance 7.55secs. the number of validation checks is 6.

5.2.2 Retrieval Accuracy and Time

The proposed IR-MFS-ANN method performance is analyzed with respect to the accuracy, retrieval time, Retrieval Accuracy v/s Number of Features, and comparisons are made to the state-of-the-art methods [2], [3], [4], and [5].

Retrieval Accuracy: It is the fraction of a number of actual likelihoods to the total number of likelihoods. It is given by using Equation (13)

Accuracy=\frac{TP+TN}{TP+TN+FP+FN}

13

The parameters, TP – True positive, FP – False positive, TN - True negative and FN- False negative. *TP* is the number of samples that belong to the retrieved class. *FP* is the number of samples that belong to the retrieved class but are non-relevant. *FN* is the number of samples not retrieved but is relevant samples and *TN* is the number of samples not belong to the retrieved class and are non-relevant. For the image retrieval model, the performance measures are considered with the limitation of giving ”one” as actual value to the retrieved relevant image and ”zero” for the retrieved non-relevant image. Thus the values of *TN* and *FN* become zero. The accuracy values are tabulated in Table 4. The proposed method attains an average accuracy of 0.94, 0.92, 0.95, and 0.94 for various images from Wang, Cifar-10, Oxford Flower, and ImageNet standard databases respectively.

Table 4
Accuracy and Retrieval Time of IR-MFS-ANN

Dataset	Accuracy (%)	Retrieval Time (Sec)
Wang	94	12.59
Cifar-10	92	09.54
Oxford Flower	95	11.35
ImageNet	94	10.60

Retrieval Time

The time from the point at which an image is requested to the point at which the retrieval system retrieves the image. The proposed method retrieval time values are tabulated in Table 4. The retrieval time is 12.59, 9.54, 11.35, and 10.60 seconds for various images from standard Wang, Cifar-10, Oxford Flower, and ImageNet databases respectively.

5.2.3 Retrieval Accuracy *v/s* Number of Features

The accuracy performance with respect to the number of features is shown in Table 5 and depicted in the Fig. 9. It depicts that the accuracy purely depends on the extracted number of features through the input image. Totaly four sets of features are extracted from the images namely OLBP, Color, Statistical and HoG features. Initially, the system is tested with the Color feature alone that is shown in Fig. 9 in blue color for that it produces an accuracy of 84%. After that the system is tested with OLBP and Color features for that the system produces the accuracy of 86% that is depicted in orange color. Then the system is tested with 75% of features that is OLBP, Color, and statistical feature for that it produces the accuracy of 88% that is depicted in light green color. Finally, the system is tested with all the features for that it produces the accuracy of 94%, it is shown in light blue color. Overall the system produces comparatively better accuracy when the greater number of features are added to the system and it is shown in the Table 4.

Table 5
Model Performance on Wang Dataset

Features	Accuracy(%)
Color	84
Color + OLBP	86
Color+OLBP+Statistical	88
Color+OLBP+Statistical+HoG	94

Table 6
Comparison of Accuracy of IR-MFS-ANN

Dataset	Method	Accuracy (%)
Wang	**IR-MFS-ANN**	94
	KNN [3]	82
	CNN [4]	80
	Color [2]	88
	ED [5]	81
Cifar-10	**IR-MFS-ANN**	92
	KNN [3]	84
	CNN [4]	72
	Color [2]	81
	ED [5]	80
Oxford Flower	**IR-MFS-ANN**	95
	KNN [3]	94
	CNN [4]	89
	Color [2]	93
	ED [5]	87
ImageNet	**IR-MFS-ANN**	94
	KNN [3]	92
	CNN [4]	92
	Color [2]	94
	ED [5]	92

5.3 Performance Comparison

Comparison of Retrieval Accuracy

The retrieval accuracy of various methods for various databases is tabulated in Table 6. The proposed method IR-MFS-ANN is compared with existing methods [2–5] for performance analysis is shown in Fig. 10. It illustrates that the proposed approach produces accuracy higher than the existing approaches.

Table 7
Retrieval Time comparison with the existing methods

Dataset	IR-MFS-ANN	KNN[3]	CNN[4]	Color[2]	ED[5]
Wang	**12.59**	17.94	21.74	13.75	12.88
Cifar-10	**09.54**	23.29	27.61	12.19	15.93
Oxford Flower	**11.35**	24.87	26.88	13.22	14.21
ImageNet	**10.60**	16.00	15.22	10.68	11.66

Comparison of Retrieval Time

The retrieval time of various methods for various databases is tabulated in Table 7. The comparisons are made with the state-of-the-art methods [2–5] and are shown in Fig. 11. It illustrates that the systems without classifiers have low retrieval time and the systems with classifiers require more time.

6 Conclusions

An ANN-based CBIR model - IR-MFS-ANN is proposed. Image analysis is performed using ANN and retrieved more true positive samples. The retrieval time of the model is decreased due to the extraction of multiple features with color, statistical, OLBP and HoG descriptors. The model is validated and result shows 94%, 86%, 95% and 94% of average accuracy for Wang, Cifar-10, Oxford Flower, and ImageNet standard databases respectively which are comparatively better than the existing methodologies [2–5]. The extracted features combination adds more descriptive information to the retrieval process thus it makes the model flexible in retrieving images. Developing framework which provides recommendation based results to fulfil user requirements based on deep learning technique is important. In future, with the deep features proposed model can be evaluated.

Declarations

Ethical Approval: The manuscript should not be submitted to more than one journal for simultaneous consideration. Results should be presented clearly, honestly, and without fabrication, falsification or inappropriate data manipulation.

Funding Details: There is no funding for this research work

Conflict of Interest: The authors have no conflicts of interest to declare.

Informed Consent: All co-authors have seen and agree with the contents of the manuscript and there is no financial interest to report.

Authorship contributions: Corresponding author has implemented the proposed methodology, results and paper writeup. Co-author has done the literature survey, results development and manuscript preparation. Total manuscript is prepared by both the authors together.

References

[1] J. Yang, B. Jiang, B. Li, K. Tian, and Z. Lv, A Fast Image Retrieval Method Designed for Network Big Data, IEEE Transactions on Industrial Informatics, 13(5) (2017) 2350–2359.

[2] J. Yue, Z. Li, L. Liu, and Z. Fu, Content based Image Retrieval using Color and Texture Fused Features, Journal of Mathematical and Computer Modelling, 54(3-4) (2011) 1121–1127.

[3] M. Fachrurrozi, A. Fiqih, B. R. Saputra, R. Algani, and A. Primanita, Content Based Image Retrieval for Multi-Objects Fruits Recognition using K-Means and K-Nearest Neighbor, in the Proceedings of International Conference on Data and Software Engineering (ICoDSE), (2017) 1–6.

[4] Y. Cai, Y. Li, C. Qiu, J. Ma, and X. Gao, Medical Image Retrieval based on Convolutional Neural Network and Supervised Hashing, IEEE Access, 7 (2019) 51877–51885.

[5] N. Khan and W. Khan, Normalised Euclidean Distance based Image Retrieval using Coefficient Analysis, Journal of Advances in Computing and Information Technology, (2013) 777–785.

[6] X. M. Ren, X. F. Wang, and Y. Zhao, An Efficient Multi-Scale Overlapped Block LBP Approach for Leaf Image Recognition, in the Proceedings of International Conference on Intelligent Computing, (2012) 237–243.

[7] M. L. Wu, C. S. Fahn, and Y. F. Chen, Image-format-independent Tampered Image Detection based on Overlapping Concurrent Directional Patterns and Neural Networks, Journal of Applied Intelligence, 47(2) (2017) 347–361.

[8] G. Zolynski, T. Braun, and K. Berns, Local Binary Pattern based Texture Analysis in Real Time using a Graphics Processing Unit, Journal of Verein Deutscher Ingenieure Berichte, 12 (2008) 321–328.

[9] V. Takala, T. Ahonen, and M. Pietikainen, Block-Based Methods for Image Retrieval using Local Binary Patterns, in the Proceedings of Scandinavian Conference on Image Analysis, (2005) 882–891.

[10] S. Liao, X. Zhu, Z. Lei, L. Zhang, and S. Z. Li, Learning Multi-Scale Block Local Binary Patterns for Face Recognition, in the Proceedings of International Conference on Biometrics, (2007) 828–837.

[11] L. Zhang, R. Chu, S. Xiang, S. Liao, and S. Z. Li, Face Detection based on Multi-Block LBP Representation, in the Proceedings of International Conference on Biometrics, (2007) 11–18.

[12] V. H. Mahale, M. M. H. Ali, P. L. Yannawar, and A. T. Gaikwad, Image Inconsistency Detection using Local Binary Pattern, Journal of Procedia Computer Science, 115 (2017) 501–508.

[13] K. Firuzi, M. Vakilian, B. T. Phung, and T. R. Blackburn, Partial Discharges Pattern Recognition of Transformer Defect Model by LBP and HoG Features, IEEE Transactions on Power Delivery, 34(2) (2018) 542–550.

[14] V. Chaurasia and Vaishali, Statistical Feature Extraction based Technique for Fast Fractal Image Compression, Journal of Visual Communication and Image Representation, 41 (2016) 87–95.

[15] S. Singh and N. P. Singh, Machine Learning based Classification of Good and Rotten Apple, Journal of Recent Trends in Communication, Computing, and Electronics, 524 (2019) 377–386.

[16] B. Khaldi, O. Aiadi, and K. M. Lamine, Image Representation using Complete Multitexton Histogram, Journal of Multimedia Tools and Applications, 79 (2020) 1–19.

[17] M. J. Khan, M. A. Riaz, H. Shahid, M. S. Khan, Y. Amin, J. Loo, and H. Tenhunen, Texture Representation through Overlapped Multioriented Tri-scale Local Binary Pattern, IEEE Access, 7 (2019) 66668–66679.

[18] D. Wu, K. Kim, G. El Fakhri, and Q. Li, Iterative Low-dose CT Reconstruction with Priors Trained by Artificial Neural Network, IEEE Transactions on Medical Imaging, 36(12) (2017) 2479–2486.

[19] G. M. Galshetwar, P. W. Patil, A. B. Gonde, L. M. Waghmare, and R. Maheshwari, Local Directional Gradient Based Feature Learning for Image Retrieval, in the Proceedings of IEEE Thirteenth International Conference on Industrial and Information Systems (ICIIS), (2018) 113–118.

[20] J. S. Arunalatha, Y. Rangaswamy, K. Shaila, K. B. Raja, D. Anvekar, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, “IRHDF: Iris Recognition using Hybrid Domain Features,” in the Proceedings of Annual IEEE India Conference (INDICON), (2015) 1–5.

[21] P. E. Gill and W. Murray, Safeguarded Steplength Algorithms for Optimization using Descent Methods, Scientific and Technical Aerospace Reports, 13(7) (1975) 5601–6503.

[[22] D. E. Rumelhart, Learning Representations by Error Propagation, in DE Rumelhart, JL McClelland and PDP Research Group, Journal of Parallel Distributed Processing, 1 (1986) 318–362.

[23] G. E. Hinton, Connectionist Learning Procedures, Journal of Machine Learning, 3, (1990) 555–610.

[24] Y. LeCun, Generalization and Network Design Strategies, Journal of Connectionism in Perspective, 19, (1989) 143–155.

[25] M. R. Hestenes, Conjugate Direction Methods in Optimization, 12 (2012) 231–310. [26] M. J. D. Powell, Restart Procedures for the Conjugate Gradient Method, Journal of Mathematical Programming, 12(1) (1977) 241–254.

[[27] P. E. Gill, W. Murray, and M. H. Wright, Practical Optimization, Society for Industrial and Applied Mathematics, (2019) 357–401.

[28] R. Fletcher, Practical Methods of Optimization, John Wiley and Sons, Scotland, (2013).

[29] E. M. Johansson, F. U. Dowla, and D. M. Goodman, Backpropagation Learning for Multilayer Feed-Forward Neural Networks using the Conjugate Gradient Method, International Journal of Neural Systems, 2(4) (1991) 291–301.

[30] M. Moller, Supervised Learning on Large Redundant Training Sets, International Journal of Neural Systems, 4(1) (1993) 15–25.

[31] R. Battiti and F. Masulli, BFGS Optimization for Faster and Automated Supervised Learning, in the Proceedings of International Conference on Neural Network, (1990) 757–760.

[32] O. A. Vatamanu, M. Frandes, M. Ionescu, and S. Apostol, Content based Image Retrieval using Local Binary Pattern, Intensity Histogram and Color Coherence Vector, in the Proceedings of e-Health and Bioengineering Conference (EHB), (2013) 1–6.

[33] T. Ojala, M. Pietikainen, and T. Maenpaa, A Generalized Local Binary Pattern Operator for Multiresolution Gray Scale and Rotation Invariant Texture Classification, in the Proceedings of International Conference on Advances in Pattern Recognition, (2001) 399–408.

[34] T. Barbu, A. Ciobanu, and M. Costin, Unsupervised Color-based Image Recognition using a LAB Feature Extraction Technique, Journal of Buletinul Institutului Politehnic Iasi, Universitatea Tehnica, Gheorghe Asachi, 57 (2011) 33–41.

[35] C. S. Hlaing and S. M. M. Zaw, Tomato Plant Diseases Classification using Statistical Texture Feature and Color Feature, in the Proceedings of IEEE/ACIS Seventeenth International Conference on Computer and Information Science (ICIS), (2018) 439–444.

[36] V. Kumar and P. Gupta, Importance of Statistical Measures in Digital Image Processing, International Journal of Emerging Technology and Advanced Engineering, 2(8) (2012) 56–62.

[37] Y. Li, Y. Zhang, X. Huang, and J. Ma, Learning Source-Invariant Deep Hashing Convolutional Neural Networks for Cross-Source Remote Sensing Image Retrieval, IEEE Transactions on Geoscience and Remote Sensing, 56(11) (2018) 6521–6536.

[38] H. Karamti, M. Tmar, and F. Gargouri, Content-based Image Retrieval System using Neural Network, in the Proceedings of IEEE/ACS Eleventh International Conference on Computer Systems and Applications

(AICCSA), (2014) 723–728.

[39] I. El Nabarawy, A. M. Abdelbar, and D. C. Wunsch, Levenberg Marquardt and Conjugate Gradient Methods Applied to a High-Order Neural Network, in the Proceedings of International Joint Conference on Neural Networks (IJCNN), (2013) 1–7.

[40] https://www.kaggle.com/ambarish/wangdataset.

[41] https://www.kaggle.com/cantonioupao/oxfordflower-17categorieslabelled.

[42] http://www.cs.toronto.edu/ kriz/cifar.html.

Figures

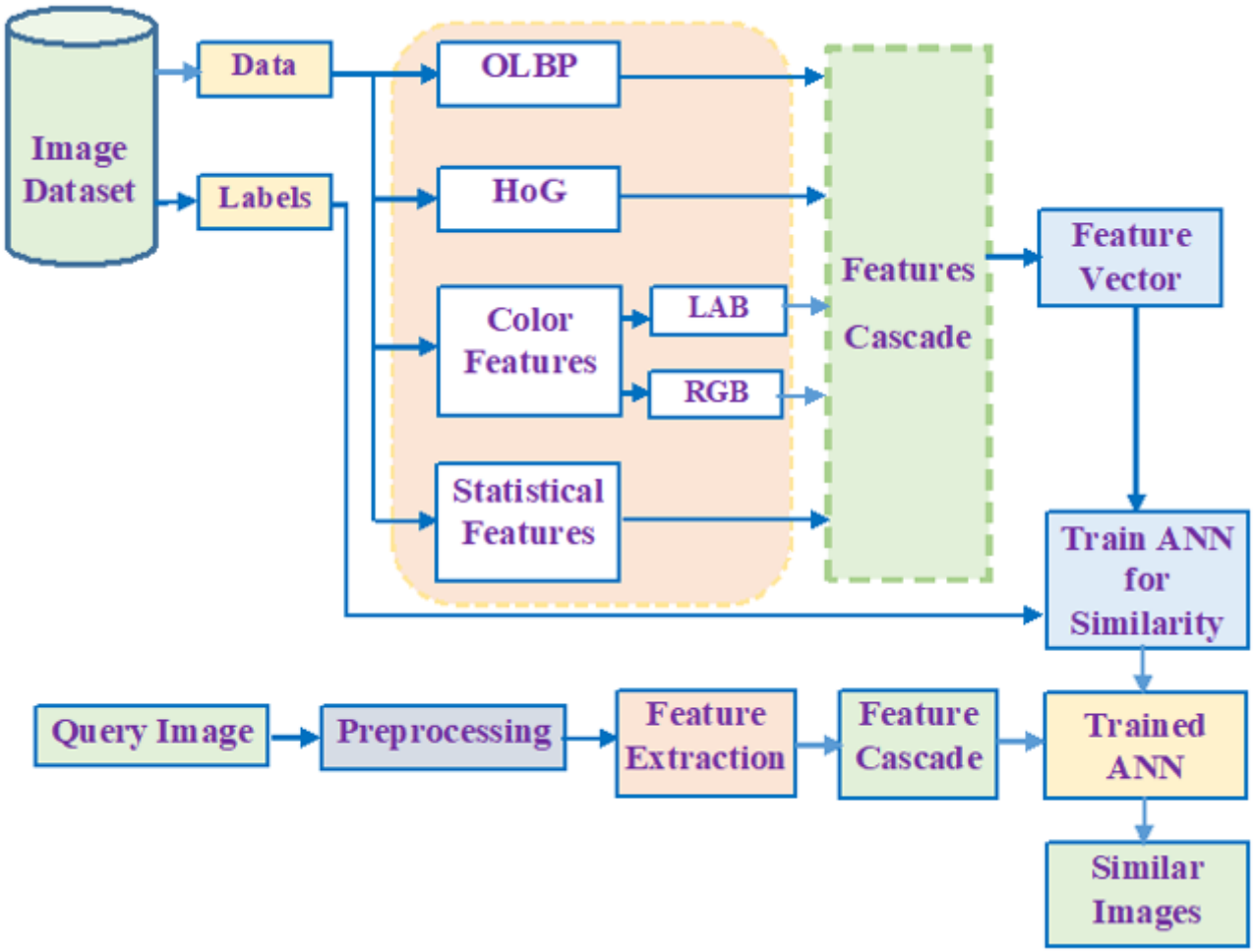

Figure 1

System Architecture

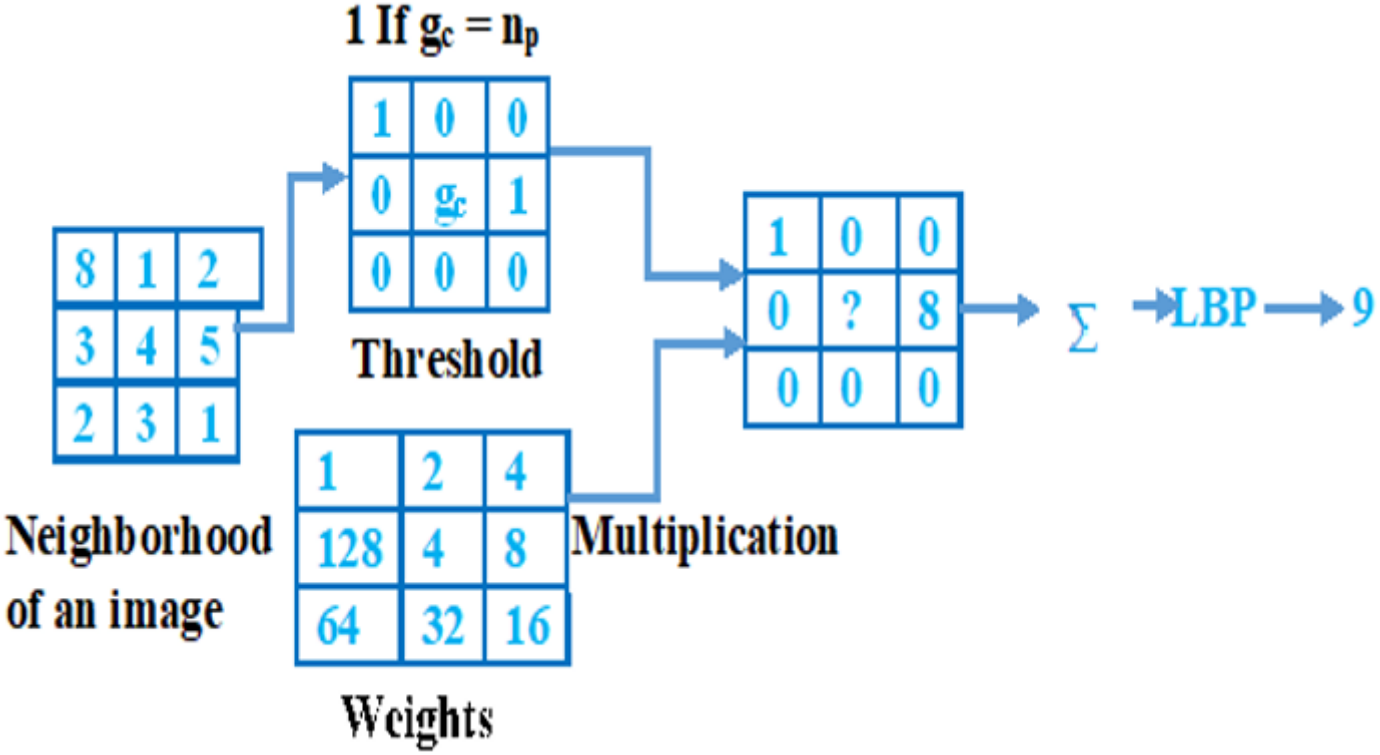

Figure 2

LBP Computation

Binary Pattern	1	0	0	1	0	0	0	0
Weight(2^P)	$1*2^7$ =1	$0*2^6$ =0	$0*2^5$ =0	$1*2^4$ =1	$0*2^3$ =0	$0*2^2$ =0	$0*2^1$ =0	$0*2^0$ =0
LBP	1	0	0	8	0	0	0	0 =9

Figure 3

LBP Code

Feature Set-1	Feature Set-2	Target
36	64	1
84	45	1
24	13	1
7	5	0
. . .	. . .	. . .
n	n	n

Figure 4

ANN Training Data Model

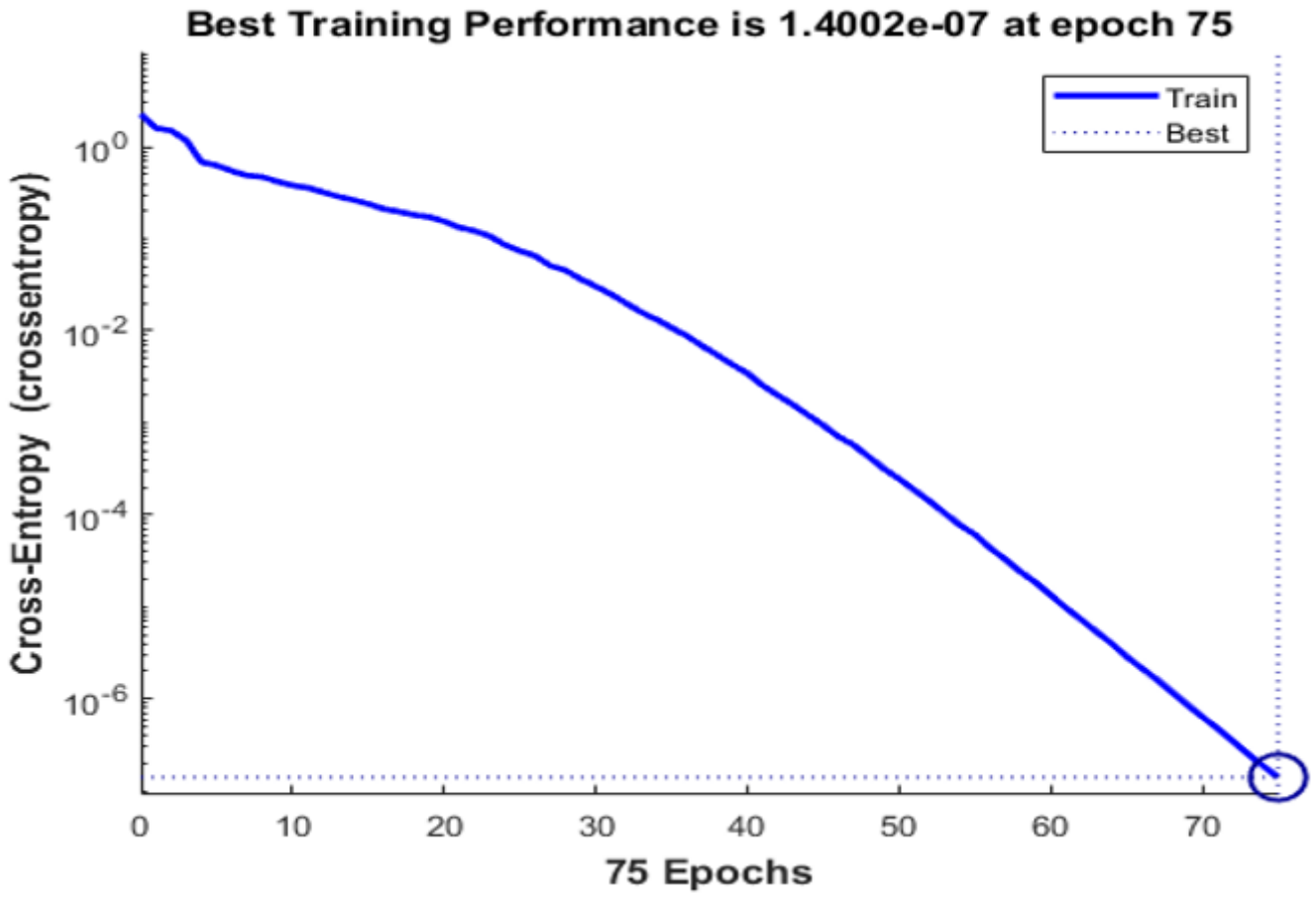

Figure 5

Conjugate Gradient Training performance plot

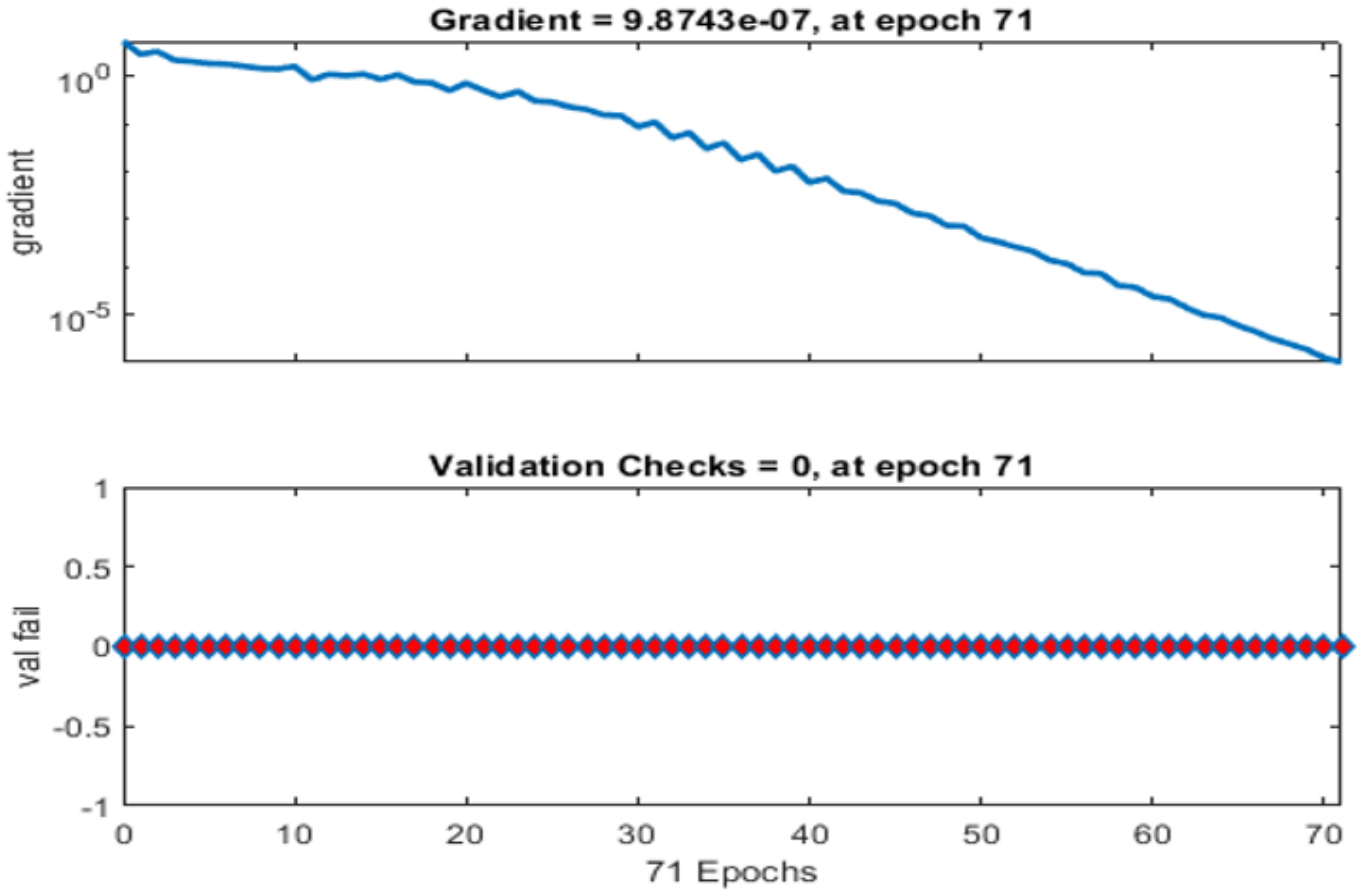

Figure 6

Conjugate Gradient Training State plot

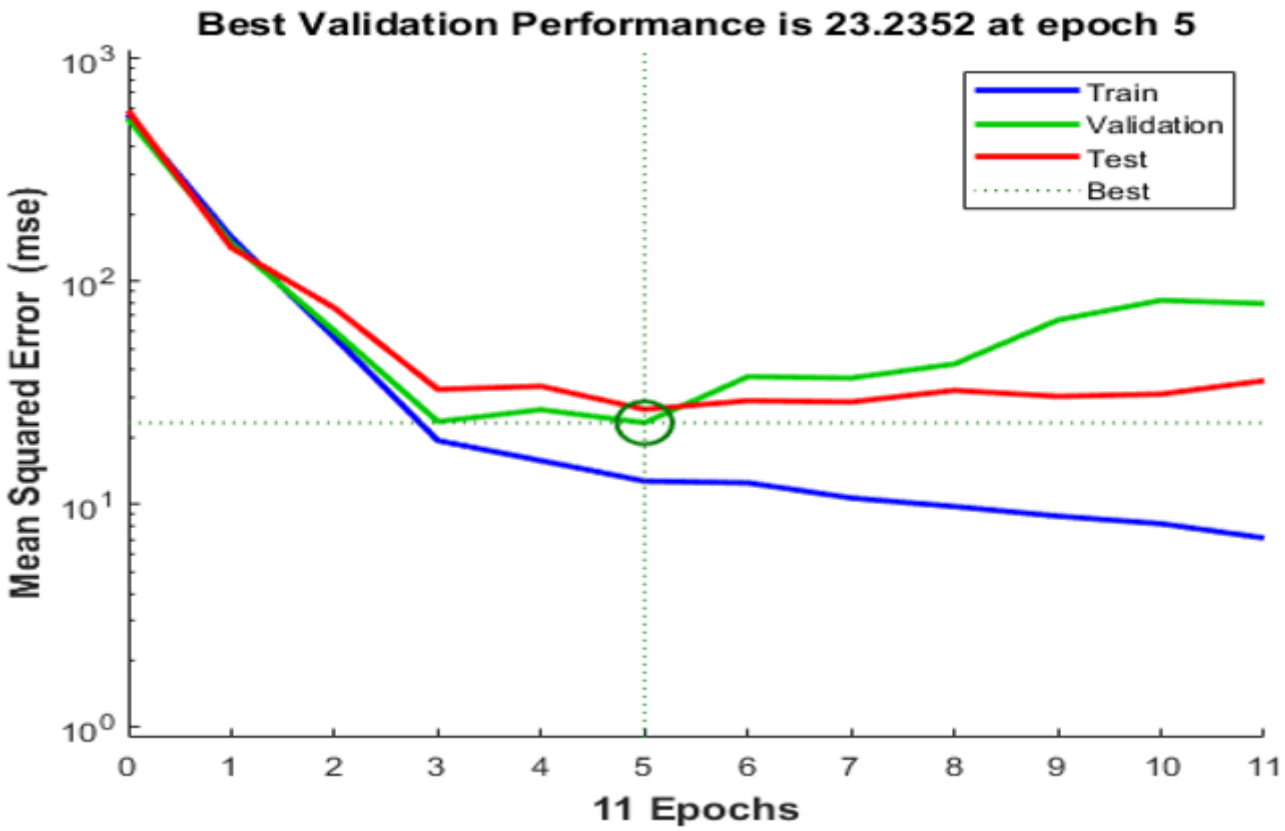

Figure 7

Scaled Conjugate Gradient Validation performance plot

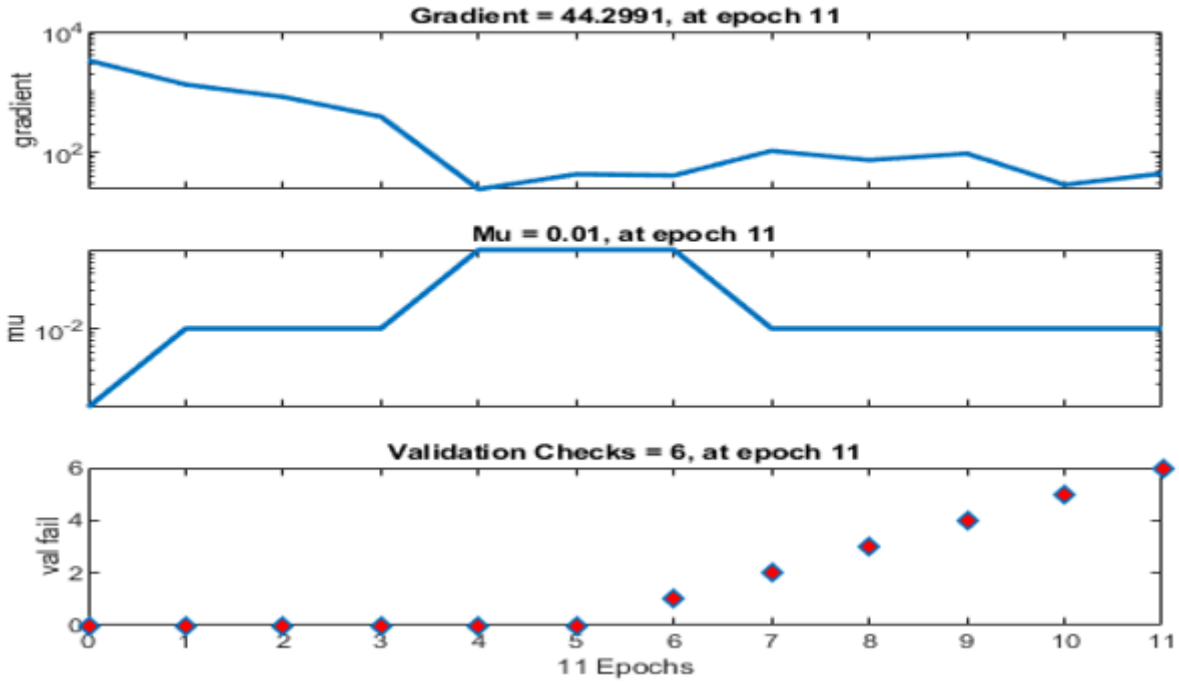

Figure 8

Scaled Conjugate Gradient Validation state plot

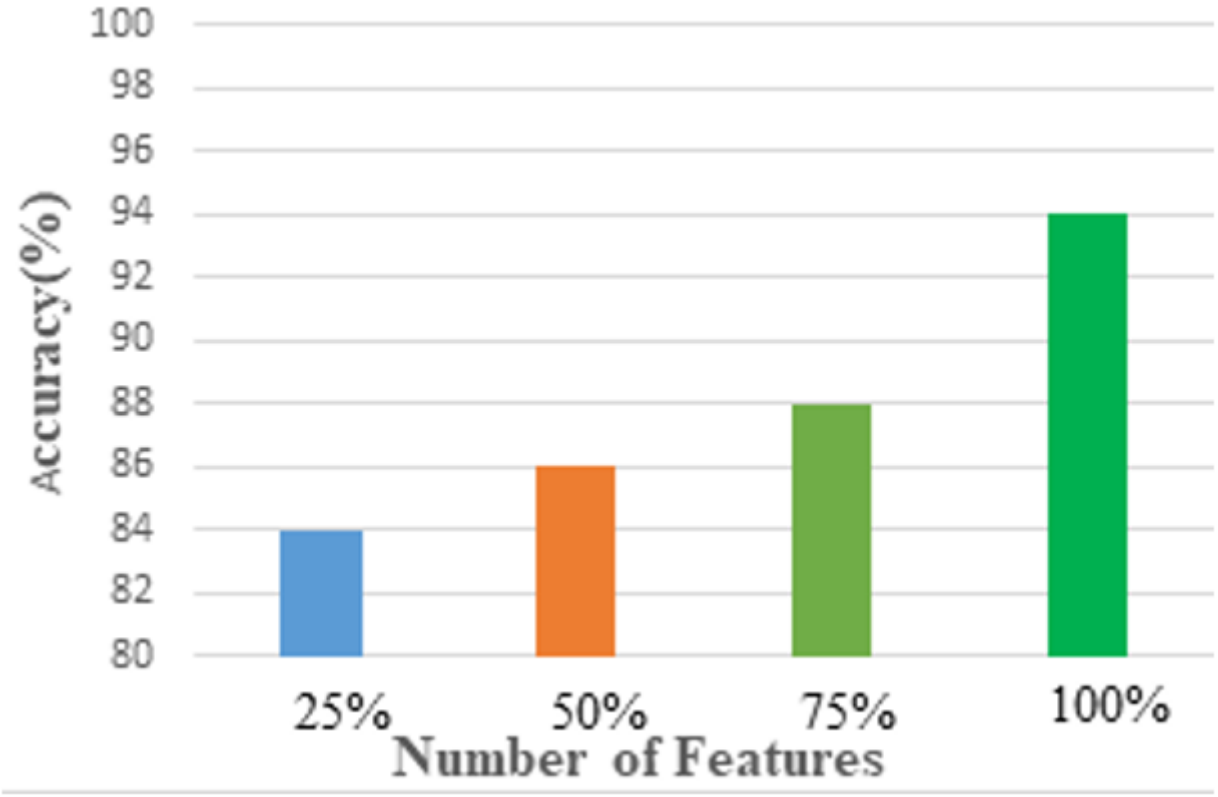

Figure 9

Accuracy *v/s* Number of Features

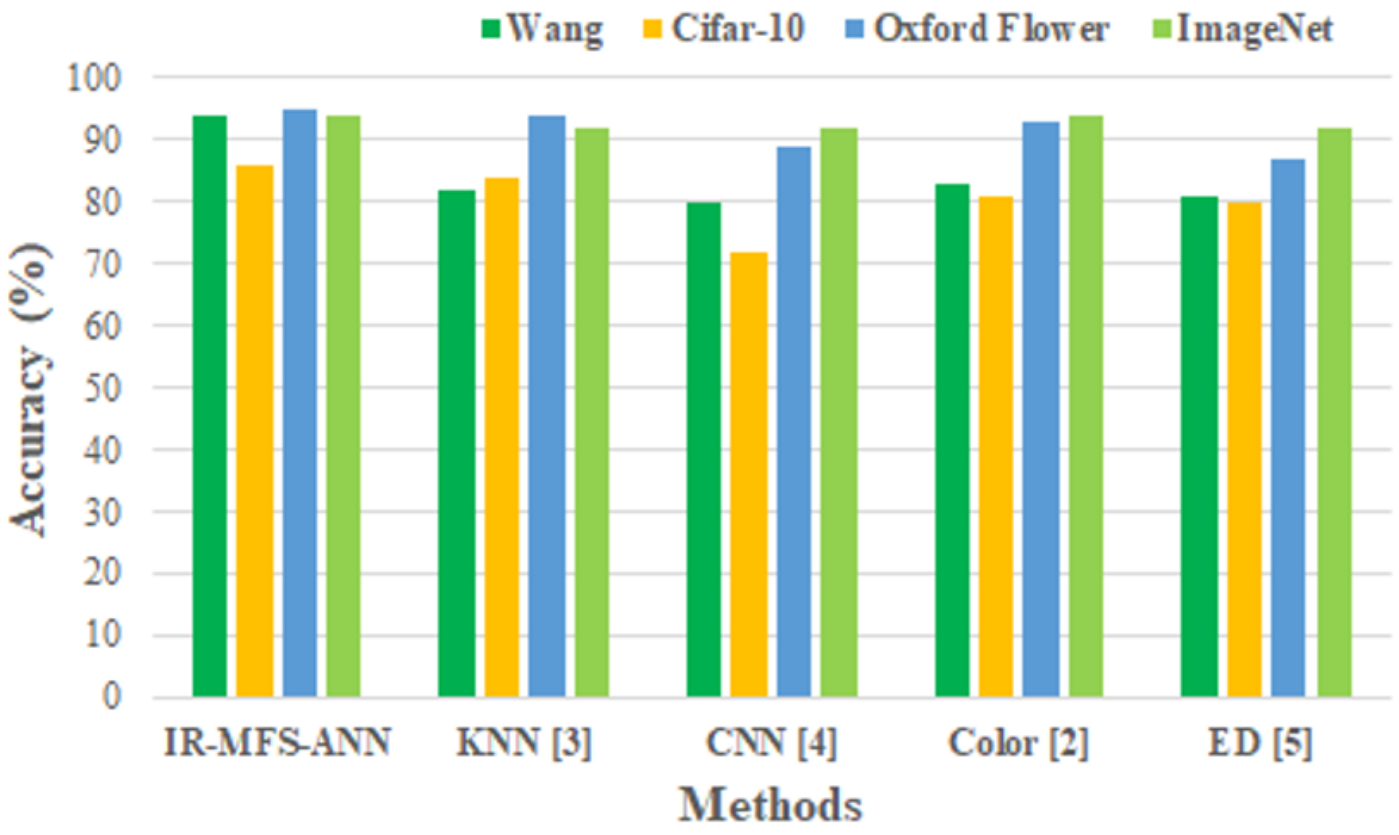

Figure 10

Comparison of Accuracy of various methods on Datasets

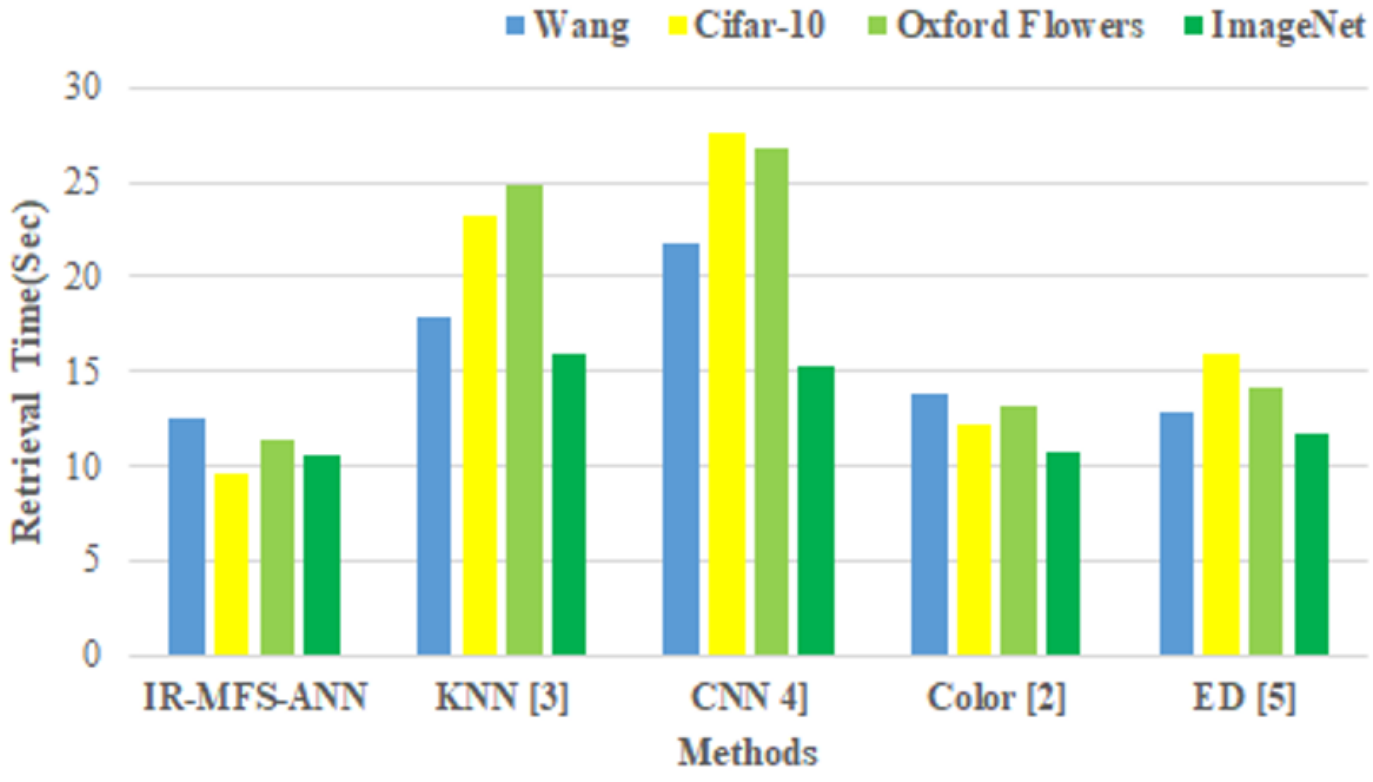

Figure 11

Comparison of Retrieval Time of various methods on Datasets