

## CBIR Features Extraction with GLCM, HC and HOG using SVM-RF Classifier



<sup>1</sup>Anil Mishra, <sup>2</sup>Tanmay Kasbe

<sup>1</sup>Research Scholar, Oriental University, Indore, Madhya Pradesh- 453555, India [mishra.anil91@gmail.com](mailto:mishra.anil91@gmail.com)

<sup>2</sup>Associate Professor, Oriental University, Indore, Madhya Pradesh- 453555, India  
[tanmay.kasbe@gmail.com](mailto:tanmay.kasbe@gmail.com)

### ABSTRACT

Content-Based Image Retrieval (CBIR) allows search and recovery of pictures that are similar to a known picture, by using attributes that represent the visual content of the pictures. Our proposed study aimed to model a method for the recovery of indexed pictures in databases from their visual content, without the need for textual annotations. Gray Level Co-Occurrence Matrix (GLCM), Harris Corner, and Histogram of Oriented Gradients (HOG) attributes are extracted from the publicly available databases (Corel and Caltech Datasets). A novel non-parametric method of texture combination is applied by means of Principal Components Analysis (PCA). Finally the classification is accomplished by using Support Vector Machine (SVM) and Random Forest Classifiers. The simulation outcomes show satisfactory performance accordant with accuracy, recall, precision and F-score.

**Key words :** CBIR, GLCM, HOG, PCA, Random Forest, SVM.

### 1. INTRODUCTION

Presently, with the increasing demand for information, there is a need to resort much faster and more efficient methods for recovering it. That is why it is essential to have large capacity databases not only to store information but also for pictures that require much more space and without loss of any of their characteristics. Once the pictures are stored in the database, it is needed to recover the said pictures. These pictures will be processed and treated using some specific technique or method. In our work we use rotated, translated and scaled variants of training pictures for comparing purpose.

CBIR is based on comparing an “X” picture with another “Y” picture and identifying its similarities. It is widely used in many areas such as medicine, police records, face detection, etc.

In CBIR there are different content-based comparison techniques [1] such as:

- Color: It involves comparing the color histogram, this technique being the most used since it is independent of its dimension or its orientation.
- Shape: The detection of shapes is carried out by segmentation or location of edges.
- Texture: It concern with viewable arrangement and its direction in space.

CBIR techniques provide a solution to an information recovery problem using only characteristics or features extracted from said pictures [2]. The problem may be recovery a picture of interest or obtain similar pictures among large collection. The criteria to consider a picture similar are variations in perspective, lighting conditions or scale. The stored pictures are pre-processed and then their corresponding descriptors are indexed. The query picture is also pre-processed to extract its descriptor. After that it is compared to the stored ones by applying appropriate similarity measures, which allow the recovery of those pictures that are alike.

CBIR techniques are used in various branches of science such as medicine, agriculture, security and protection, weather forecasting, modeling of biological processes, picture classification, web, crime prevention, satellite picture processing, among others [3,4,5,6,7,8,9,10,11,12].

Traditional approaches mainly include the elaboration of descriptors from the picture content, through the so-called secondary-stage attributes such as color [13], texture [14] and shape [15] or combination of some of these [16]. A positive aspect of the techniques developed based on these approaches is that they do not require large amounts of data or time to obtain satisfactory results during the training and inference stages. On the other hand, they try to obtain local and global descriptors of the pictures from manually elaborated features, which are not generic, but rather have a strong dependence on the classes represented in the pictures. They generally restrict their chances of successfully scaling up picture collections with large numbers of classes or categories.

Recent methods combine secondary-stage attributes with other primary-stage attributes, which provide a representation closer to human perception, allowing a semantic description of the pictures to be achieved and better recovery results. The main advances in this direction are related with the rapid development of machine learning techniques [17,18,19,20]. Neural models learn global descriptors constructed on the basis of a hierarchy of traits and adjusted through a training process. These descriptors are generic and robust to challenges such as variability between classes, or changes in perspective or lighting. However, the feature vectors obtained by these techniques have, in most cases, a large dimension (2048, 4096 components), negatively impacting the use of memory for storage and the time complexity of the comparison process and recovery. Apart from these, there is a major disadvantage of using deep learning approaches

i.e. requirement of longer training time. This problem is easily resolved by using the supervised classification approaches such as support vector machine [21], random forest classifier [22], etc. For high-dimensional data, the SVM scales relatively well. There is less risk of over-attachment in SVM. In Random forest, the decision tree is essential when dealing with multiple features that may be correlated.

The paper tries to resolve the problem posed by developing a method where supervised classification is used. It uses specifically SVM and Random forest classifier, to obtain global descriptors of the pictures. The size of these vector representations will be reduced by applying Principal Component Analysis. Robust measurements, widely used within the picture recovery domain, will be used to find out the likeness among pictures.

## 2. DEFINITION OF THE PROBLEM

CBIR system is used for managing large amount of multimedia data for specific applications. The keywords were used to index the pictures which were a traditional approach, but when it comes to a high volume of pictures emerging from different sensors, annotations become a costly and imprudent endeavor. Pictures of different types are captured daily and using different techniques. Particularly in hospitals, clinics and health centers, X-rays, mammograms, ultrasounds, MRIs, CT scans, histology pictures, and many more are available for biomedical analysis.

In [23] a production of more than 12,000 pictures per day is reported at the University Hospital of Geneva during 2002. Thanks to the capture facilities of today we are able to produce much more information than we can process and manage, which is why it is necessary to develop CBIR systems.

Although this need has been present in recent years and is becoming more urgent today, it continues to be an unsolved problem due to the inherent difficulties in the treatment and understanding of pictures, concepts that are explained below.

### A. Understanding the Pictures

In the research work on artificial vision and CBIR systems, two fundamental gaps have been identified when trying to understand or work with a digital picture:

- Sensory Void: Refers to the void the actual world article with the facts captured by any of the physical storage methods [24].
- Semantic Void: It has to do with the mismatch between the information that can be extracted from the visual data and the interpretation that the user makes of the same data [24].

Research to achieve a better understanding of pictures is useful in CBIR systems to be able to more accurately express the properties of a picture. Much of the subsequent work in that system builds upon the chosen representation. So to increase the accuracy with which a system retrieves a picture for its content, a structure must be carefully

selected to decrease or eliminate the information gap that exists in digital pictures.

### B. Efficiency and Computational Load

CBIR systems must work efficiently with a huge number of pictures. In the assessment methodology proposed in [25], the need for interactive systems, defined as systems that are capable of responding to a query in less than one second, is discussed.

Several works have been developed using techniques whose average execution time is greater than this requirement. However, these results are mainly due to the complexity in the management of the representation structures, making it necessary to balance the compression between precision and performance. It should also be noted that interactivity is achieved after the integration of good hardware architecture and an excellent recovery algorithm.

### C. Consultation Problems

There are several consultation paradigms in CBIR systems:

- Query by keywords: keywords are used to retrieve pictures that have the grammatical concept expressed by the user associated. In [26], automatic annotations are made from the visual properties of the picture that require specific knowledge of the application area.
- Query for visual controls: In this case controls are used that allow the user to select desired visual properties in the resulting pictures. Controls include color selection, texture, contrasts, brightness, and combinations of these and more [27].
- Query for an example picture: Systems with this query paradigm allow selecting an picture that has desired characteristics in the results. The system takes the sample picture, analyzes it and then searches the database for the most outstanding patterns that were found.
- Query by strokes: Allows the user to trace in the ways they consider most important in the results. You can combine the strokes with visual operators like color or texture. Shapes can be built with the help of the system, interpolating points or deformed primitives.

The consultation problems appear due to the impossibility of human beings to express some properties or meanings of the pictures, which cannot be represented in natural language. Research in human-machine interaction systems can provide concepts that make it easier for users to express their consultation needs, in a simple and intuitive way, to increase user satisfaction.

## 3. PROPOSED METHODOLOGY

The section layout of the system is presented the in figure 1. It is divided into two parts:

- 1) **User Interface:** User input reference picture and processed pictures are obtained.

2) **Picture Analysis:** This is done in two phases. In first, the attributes vectors of the reference picture are created by GLCM, Harris Corner and HOG. Also, the attributes vectors of the database pictures are created in the same manner. Now using classifier SVM-Random Forest, on both feature set, the like pictures are recovered.

The description of the methods is as given below:

**A. Gray Level Co-Occurrence Matrix (GLCM)**

In the analysis of textures, the extraction of characteristics is carried out from the statistical distribution with which combinations of certain intensities are observed in relative positions of the picture. The co-occurrence matrix is a square matrix where rows and columns coincides with the gray levels in the picture to be analyzed and where each element of the matrix  $C(i, j)$  contains pair-pixel of picture, with intensities  $i$  and  $j$  respectively, and with  $gapd$  and direction  $\theta$ , occur in a certain neighborhood. In other words, the element  $C(i, j|d, \theta)$  contains the probability that any two pixels at a distance  $d$  and an angle  $\theta$  have respectively gray levels  $i$  and  $j$ . GLCM are constructed from the analysis of pairs of pixels for a distance and orientations given between them.

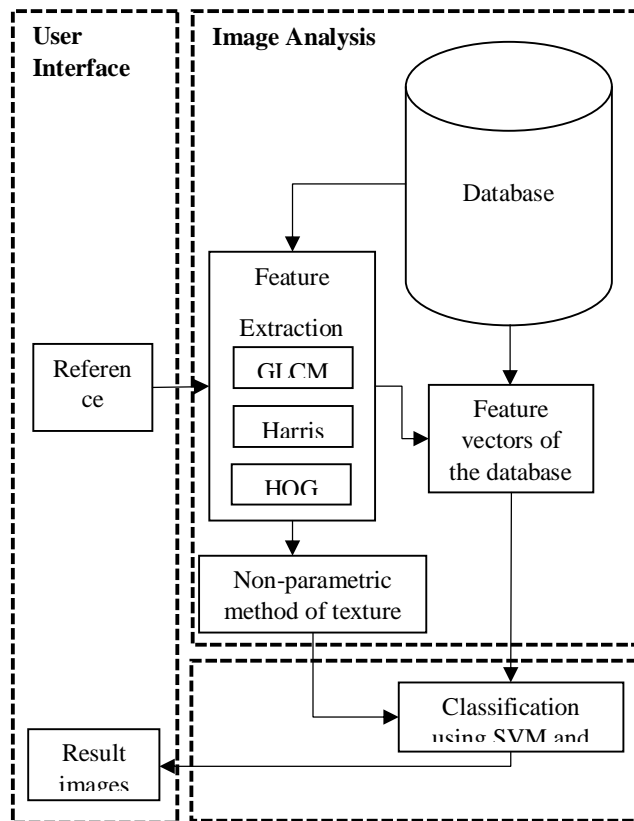


Figure 1: Block diagram

Below are some measures of detection of texture features [28]:

- **Angular Second Moment (ASM):** It is also called uniformity. The higher the ASM, the greater uniformity (less variation in gray levels), if the ASM is 1, the picture is completely uniform.

$$\sum_{i,j=0}^{N-1} P_{i,j}^2 \quad (1)$$

- **Contrast:** It is the amount of local variations in the shades of gray in the picture. The greater the variation in shades of gray, the greater the contrast.

$$\sum_{i,j=0}^{N-1} P_{i,j} (i - j)^2 \quad (2)$$

If the contrast is 0, the gray levels are constant throughout the picture.

- **Correlation:** It is the linear dependence of the shades of gray in the picture:

$$\sum_{i,j=0}^{N-1} P_{i,j} \frac{(i - \mu_i)(j - \mu_j)}{\sqrt{(\sigma_i \sigma_j)}} \quad (3)$$

If the correlation is 0, there is no linear correlation between the gray levels.

- **Entropy:** This is a computation of the complexity of the picture. The greater its value, greater is the complexity.

$$\sum_{i,j=0}^{N-1} P_{i,j} (-\ln P_{i,j}) \quad (4)$$

- **Inverse Difference Moment (IDM):** It is also called homogeneity. The IDM increases when the contrast between the pixel pairs decreases.

$$\sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1 + (i - j)^2} \quad (5)$$

Each one at three distances (1, 5 and 10 pixels) and four orientations (0°, 45°, 90° and 135°), and their corresponding mean values as shown in figure 2.

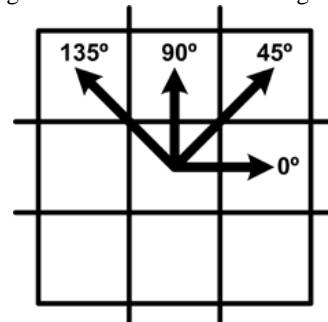


Figure 2: GLCM distribution [29]

**B. Histogram of Oriented Gradients (HOG)**

HOG is a procedure for extracting characteristics that acts by dividing the picture into Gradient vectors. Gradient vectors are vectors that, in addition to direction and sense, give us the greatest value regarding some magnitude (in our case the gradient applies to edges), so our gradient vector at each point will have direction, direction and magnitude pointing to where we have the greatest contrast relative to edges.

In the procedure, the picture is segregated into cells of the same size and the gradient vectors are calculated in each cell with its direction and magnitude [30]. After calculating the gradients, all these vectors are organized in a Histogram depending on their angle and magnitude. After the generation of the histograms is complete, blocks are normalized, aiming at the elimination of noise and distortions.

The HOGs is based on the hypothesis that every picture will have the same changes in its illumination regardless of whether it is light or dark, and it also works efficiently

when evaluating pictures with change of scale; thanks to the fact that it gives a description of the gradients within the picture and their direction.

Given a grayscale picture divided into cells (in this case contemplated as 8x8), the computational gradient described by equations (6) and (7):

$$G_p = [-1,0,1] \cdot [I((p-1),q), I(p,q), I((p+1),q)]^T \tag{6}$$

$$G_q = [-1,0,1] \cdot [I(p,(q-1)), I(p,q), I(p,(q+1))]^T \tag{7}$$

The magnitude of each pixel is defined by equation (8):

$$G(p,q) = \sqrt{G_p(p,q)^2 + G_q(p,q)^2} \tag{8}$$

While its direction is given as follows (9):

$$\varphi(p,q) = \arctan \frac{G_q(p,q)}{G_p(p,q)} \tag{9}$$

Conventionally 9 containers are used that will have all angles between 0 and 180 degrees. When the angle of a pixel falls within the range of the container, its magnitude is added to it and considered as 0 for all the others, as described in equation (10):

$$hog_i = \begin{cases} G(p,q) \text{ and } \varphi(p,q) & \in (i) \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

We add each of the obtained values to the feature vectors of each of the cells within the entire picture. In order to have a correct description of all the lighting and contrast changes, it is necessary to normalize locally, so we must group the cells into larger groups. Given the above, the block descriptor is given by the integration of the normalized histograms generated in each cell. These blocks generally overlap, which means that each of the cells may be considered on more than one occasion for the final descriptor [31]:

$$hog_B = \frac{hog_B}{\sqrt{\|hog_B\|^2 + \epsilon}} \tag{11}$$

Where  $hog_B$  is the descriptor of each block and  $\epsilon$  represents a small constant value (theoretically it doesn't matter which one) added to avoid numerical problems in the square root. Then, the final descriptor will be given by the concatenation of each of the blocks within the 9 containers.

**C. Harris Corner Method**

It was invented by Harris and Stephens. We take a picture  $I(p,q)$  we start by taking the derivatives of the picture in line and in column [31]:

$$I_p(p,q) = \frac{\partial I}{\partial p}(p,q) \tag{12}$$

$$I_y(p,q) = \frac{\partial I}{\partial q}(p,q) \tag{13}$$

For the calculations we will need three matrices with the derivatives with the squares and their multiplications:

$$A(p,q) = I_p^2(p,q) \tag{14}$$

$$B(p,q) = I_q^2(p,q) \tag{15}$$

$$C(p,q) = I_p(p,q) \cdot I_q(p,q) \tag{16}$$

These three matrices must be smoothed by a Gaussian low pass filter. However, you can choose the filter that is best [32].

$$k(p,q) = k(p,q) \cdot H_{gauss} \tag{17}$$

Where,  $k = A, B$  or  $C$  and  $H_{gauss}$  is the Gaussian filter. This is a convolution.

We can now calculate the corner force for each pixel. It is calculated such that:

$$Q(p,q) = (AB - C)^2 - \alpha \cdot (A + B)^2 \tag{18}$$

Where,  $\alpha$  is the sensitivity of the detector.  $Q$  is called corner response function (CRF) and returns large values when a corner is detected. The larger  $\alpha$ , the lower the sensitivity and the lower the number of corners. Typically  $\alpha$  is between 0.04 and 0.06 but does not exceed 0.25. A thresholding is carried out to eliminate the low values of the CRF.

$$Q(p,q) < th = 0 \tag{19}$$

Generally this threshold ( $th$ ) is between  $10^4$  and  $10^7$

We can then arrange the detected corners in descending order and carry out a detection of local maxima to eliminate the too close corners. The local maxima must be made from the corner of highest value and eliminate the corners of low value.

**D. A New Non-Parametric Method of Texture Combination**

This work offers a novel combination method texture based principal component analysis. The proposed method provides a composite texture gradient which will then be used by the RF and SVM classifier for categorization of the picture.

**1. Principal Components Analysis (PCA)**

PCA is a procedure that exploits the aspect geometric and graphical representations of a set of data called observations, in order to study its variability and its dispersion in the space in which it is represented [33]. Let  $P$  be a data set with  $m$  quantitative variables having  $n$  units (often denoted individuals), defined by equation [34]:

$$P = \begin{bmatrix} \begin{pmatrix} p_{11} \\ p_{21} \\ \vdots \\ p_{i1} \\ \vdots \\ p_{n1} \end{pmatrix} & \begin{pmatrix} p_{12} \\ p_{22} \\ \vdots \\ p_{i2} \\ \vdots \\ p_{n2} \end{pmatrix} & \dots & \begin{pmatrix} p_{1j} \\ p_{2j} \\ \vdots \\ p_{ij} \\ \vdots \\ p_{nj} \end{pmatrix} & \begin{pmatrix} p_{1m} \\ p_{2m} \\ \vdots \\ p_{im} \\ \vdots \\ p_{nm} \end{pmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} (\vec{p}_1)^t \\ (\vec{p}_2)^t \\ \vdots \\ (\vec{p}_i)^t \\ \vdots \\ (\vec{p}_n)^t \end{bmatrix} \tag{20}$$

↓

$$\vec{p}_1 \vec{p}_2 \dots \vec{p}_j \dots \vec{p}_m$$

Starting from its matrix write, we can rewrite the dataset  $P$  in two vectorial forms (Equation 21): the vertical vector is composed of a set of points where each point, denoted as  $\vec{p}_i$ , is of dimension  $m$ . The horizontal vector is the second vector form and is made up of a set of points where each, noted  $\vec{p}_j$ , is of dimension  $n$ . Note that  $t$  denotes the transposed function. The points  $\vec{p}_i$  and  $\vec{p}_j$  can be given by the following equations:

$$\vec{p}_j = \begin{pmatrix} p_{1j} \\ p_{nj} \end{pmatrix}_{(1 \leq j \leq m)} \tag{21}$$

And

$$\vec{p}_i = \begin{pmatrix} p_{i1} \\ p_{im} \end{pmatrix}_{(1 \leq i \leq n)} \tag{22}$$

Where  $i$  and  $j$  indicate the individual and the variable respectively.

**PCA Algorithm: Non-Parametric-Texture Combination Method**

Inputs: I, a texture picture.

Calculate the m different descriptors on picture I in the color space  $E_1, E_2$  and  $E_3$ .

Repeat

Generate all the data

$$P^s[\vec{g}_{1,1}^s, \vec{g}_{2,2}^s, \dots, \vec{g}_{j,j}^s, \dots, \vec{g}_{s,m}^s]$$

Calculate the local covariance matrix

$$\text{Cov}_p^s = \begin{pmatrix} \text{var}(\vec{g}_1^s) & \text{cov}((\vec{g}_1^s)^t, \vec{g}_2^s) & \text{cov}((\vec{g}_1^s)^t, \vec{g}_3^s) \\ \text{cov}((\vec{g}_2^s)^t, \vec{g}_1^s) & \text{cov}((\vec{g}_2^s)^t, \vec{g}_2^s) & \text{cov}((\vec{g}_2^s)^t, \vec{g}_3^s) \\ \text{cov}((\vec{g}_3^s)^t, \vec{g}_1^s) & \text{cov}((\vec{g}_3^s)^t, \vec{g}_2^s) & \text{var}(\vec{g}_3^s) \end{pmatrix}$$

Extract the eigenvalues  $\lambda_s$  and their associated eigenvectors

$$\widehat{G}(s) = \text{argmax}_{j \in \{1, \dots, m\}} \left\{ \frac{1}{3} (g_{1j}^s + g_{2j}^s + g_{3j}^s) \right\}$$

End

Until the end of pixel  $S$  in frame I;

Outputs: Structural gradient  $\widehat{G}(s)$

Further the processed outcome of PCA is sampled for training and testing of random forest classifier and SVM.

**E. Classification**

**1. Support Vector Machine (SVM)**

The classification is present in many real-world problems, initially vector support machines were designed to deal with binary (+/ -1) problems. Now we will see how to deal with this problem. The objective function in expressed by [20]:

$$w_r \in H, \epsilon^r \in R^m, b_r \in R^{\frac{1}{2} \sum_{r=1}^M \|w_r\|^2} + \frac{c}{m} \sum_{i=1}^m \sum_{r=y_1} \epsilon_i^r \quad (23)$$

Subject to:

$$\langle W_{y_i}, X_i \rangle + b_{y_i} \geq \langle W_r, X_i \rangle + b_r + 2 - \epsilon_i^r, \quad \epsilon_i^r \geq 0 \quad (24)$$

Where,  $m \in \{1, \dots, M\} \setminus Y_i$  and  $Y_i \in \{1, \dots, M\}$  is the multi-class label of the  $X_i$  pattern.

In terms of precision, the results obtained with this approach are comparable to those obtained directly using the one against the rest method. For practical problems, the choice of approach will depend on the available limitations; relevant factors include the precision required, the time available for development, the processing time and the nature of the classification problem.

**2. Random Forest Classifier**

Random forests were introduced by Breiman (2001) by the following very general definition [35]:

The classifier comprising a set of elementary classifiers of the decision tree type, noted:

$$\{h(x, \theta_k), k = 1, \dots, L\} \quad (25)$$

Let  $(\hat{h}(\theta_1), \dots, \hat{h}(\theta_q))$  a collection of tree predictors, with  $\theta_1, \dots, \theta_q$  random variables independent of  $\mathcal{L}_n$ . The

predictor of random forests  $\hat{h}_{RF}$  is obtained is aggregating this collection of random trees as follows:

- $\hat{h}_{RF}(x) = \frac{1}{q} \sum_{l=1}^q \hat{h}(x, \theta_l)$  Average of individual tree predictions in regression.
- $\hat{h}_{RF}(x) = \text{arg max}_{1 \leq k \leq K} \sum_{l=1}^q 1_{\hat{h}(x, \theta_l)=k}$  Majority vote among individual predictions trees in classification.

The term random forest comes from the fact that individual predictors are, here, explicitly predictors per tree, and that each tree depends on an additional random variable (that is, in addition to  $\mathcal{L}_n$ ).

The classification process followed by Random Forest consists of:

1. Assignment to a node if it is terminal, deciding whether a node will be labeled as a sheet or it will carry a test.
2. If the node is not terminal, then we have to select a test to assign it.
3. If the node is terminal, then we must give it a class.

The general algorithm for decision trees is as follows:

Input: sample S

Initialize the current tree to the empty tree;

The root designates the current node

Repeat

See if the current node is terminal

If the node is terminal then

Assign it a class

If not

Select a test and generate as many new child nodes as there are

answers to this test

End if

Explore another node if there is one

Until a decision tree A is obtained

Exit: decision tree A.

**4. SIMULATION RESULTS**

**A. Datasets**

Currently, the number of picture databases available to compare CBIR systems is scarce and many of them are not freely available. Here a description will be made of each of the databases that will be used throughout this work. It should be noted that all these bases are free and open access.

**1. Corel Dataset**

For the intention of experimentation, a picture dataset having real life is used. A group of 500 Corel pictures divided into 5 classes with 100 pictures in each class construct the dataset. The pictures are of size 384x256 or 256x384. The 5 picture categories available are: (1) Africa, (2) Beach, (3) Transportation, (4) Architecture and (5) Dinosaur. The pictures in every class are numbered(Class 1: 0-99, Class 2: 100-199 etc.). This picture database is known as Corel database in the literature and has been used for experimentation in the SIMPLIcity system proposed by Wang et al., 2001 [36]. This database is publicly available for experimentation [37, 38].



**Figure 3:** Retrieved pictures of Africa Category from Corel Dataset

**2. Caltech Dataset**

For the experiment one more dataset Caltech-256 containing 30607 pictures is used. It has 256 real life categories. Each category in this dataset contains lowest of 80 pictures [39].



**Figure 4:** Retrieved pictures of Caltech Dataset

**B. Performance Evaluation Parameters**

**1. Metrics**

When comparing CBIR systems, the need arises to find metrics that measure the quality of the systems. Depending on the application or context where it is used the system, user requirements may vary and it is difficult to find a single metric that fits all cases.

There are two important points to keep in mind: first it is a measure with simple interpretation and, second it try to capture the quality notion of users of the system.

**2. Precision and Recall**

Two of the best known metrics in the literature are precision (P) and recall (R). P is the ratio of recovered pictures that are appropriate, while R is ratio of appropriate pictures that are recovered. Formally,

$$\text{Precision} = \frac{1}{N_H} \sum_{i=1}^{N_H} \text{Rel}[i] \tag{26}$$

$$\text{Recall} = \frac{1}{N_R} \sum_{i=1}^{N_H} \text{Rel}[i] \tag{27}$$

An advantage of having two metrics that evaluate different issues is that, in certain applications, one of them can be assigned more importance than the other. Finally, these metrics are defined for a single query, but when it comes to the precision or recall of the system, reference is made to the average of these values among all queries.

**3. F-Score**

Accuracy and recall can be combined into one measurement: F-Measure. The idea of this measure is to unite the precision and the recall in a single value, penalizing systems in which either of these two measures is low. It is an interesting option for contexts where both accuracy and recall are important.

$$F = \frac{2PR}{P+R} \tag{28}$$

**4. Precision at k**

In many applications such as web searches, the most important thing is to know how good the first results are. With this idea in mind, the Precision at metric is defined k, where the precision is calculated considering only the first k results. Values typical for k are 10 or 20.

**C. Results**

The experiments will be performed on the picture bases presented in this section using the aforementioned databases. They will consist of analyzing the behavior of histograms to different thresholds and metrics equations 26, 27 and 28 will be used For each consultation, a maximum of 1,000 results will be considered (for avoid returning the entire picture base, in cases of low thresholds).

		Confusion Matrix					
		an	di	fe	ha	ne	sa
Output Class	an	99 19.8%	1 0.2%	0 0.0%	0 0.0%	0 0.0%	99.0% 1.0%
	di	0 0.0%	98 19.6%	0 0.0%	1 0.2%	1 0.2%	98.0% 2.0%
	fe	0 0.0%	1 0.2%	99 19.8%	0 0.0%	0 0.0%	99.0% 1.0%
	ha	0 0.0%	0 0.0%	1 0.2%	99 19.8%	1 0.2%	98.0% 2.0%
	ne	1 0.2%	0 0.0%	0 0.0%	0 0.0%	98 19.6%	99.0% 1.0%
	sa	99.0% 1.0%	98.0% 2.0%	99.0% 1.0%	99.0% 1.0%	98.0% 2.0%	98.6% 1.4%
		Target Class					

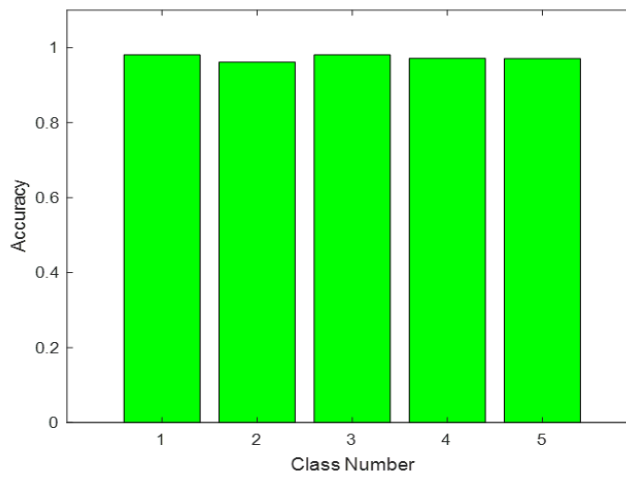
**Figure 5:** For Corel dataset with SVM

Figure 5 shows a confusion matrix plot for Corel dataset. As per the dataset, it exhibits five classes for the five picture category. Calculation for each class is given as follows:

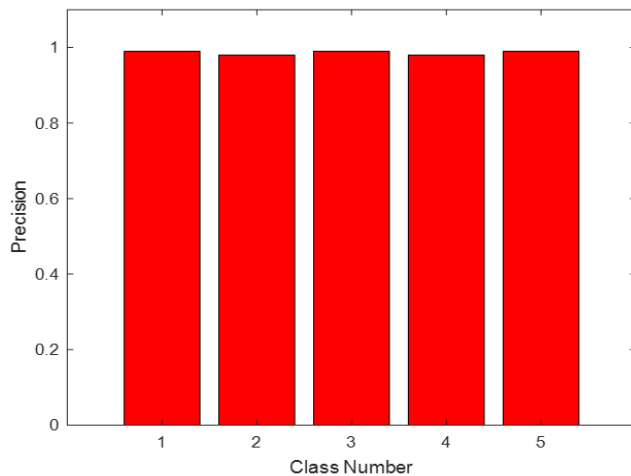
**Table 1:** Result for Corel datasetwith SVM

Parameter	'an'	'di'	'fe'	'ha'	'ne'
TP	99	98	99	99	98
FP	1	2	1	1	2
FN	1	2	1	2	1
TN	399	398	399	398	399

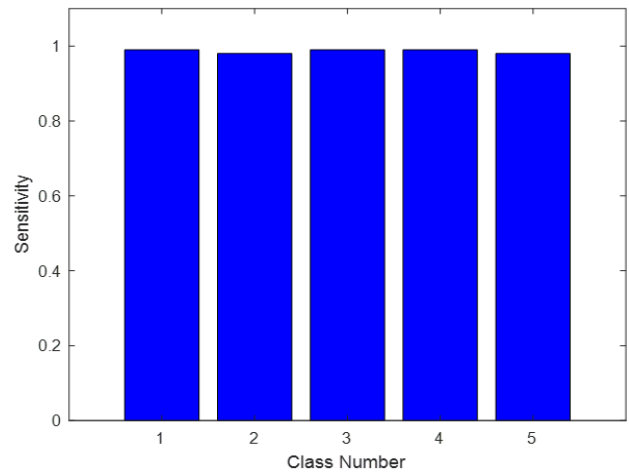
Parameter	'an'	'di'	'fe'	'ha'	'ne'
Accuracy	99.6	99.2	99.6	99.4	99.4
Precision	99	98	99	99	98
Recall	99	98	99	98	99
MissRate	1	2	1	2	1
ErrorRate	0.4	0.8	0.4	0.6	0.6
F_Score	99	98	99	98.5	98.49



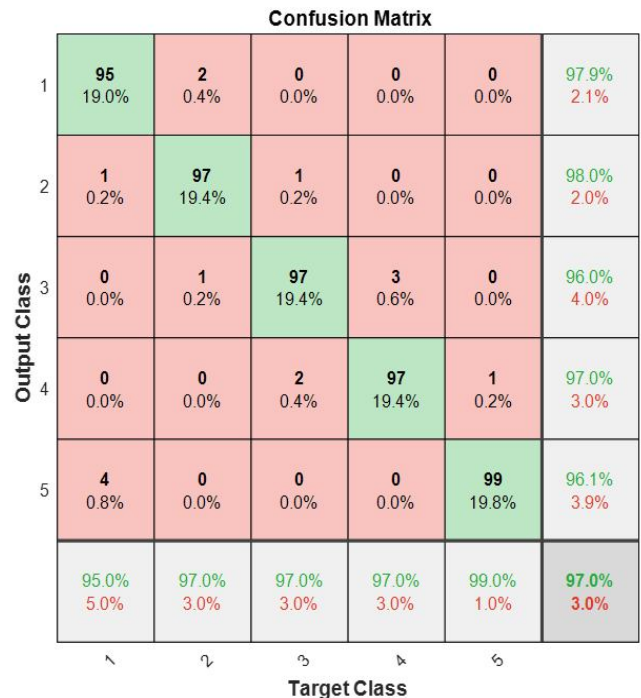
**Figure 6:** Accuracy graph for Corel datasetwith SVM



**Figure 7:** Precision graph for Corel datasetwith SVM



**Figure 8:** Sensitivity graph for Corel datasetwith SVM



**Figure 9:** For Caltech datasetwith SVM

Figure 9display a CM plot for Caltech dataset. As per the dataset, it exhibits five classes for the five picture category. Calculation for each class is given as follows:

**Table 2:** Result for Caltech dataset using SVM classifier

Parameter	1	2	3	4	5
TP	95	97	97	97	99
FP	5	3	3	3	1
FN	2	2	4	3	4
TN	398	398	396	397	396

Parameter	1	2	3	4	5
Accuracy	98.6	99	98.6	98.8	99
Precision	95	97	97	97	99
Recall	97.93	98	96.03	97	96.1
MissRate	2.1	2	3.96	3	3.9
ErrorRate	1.4	1	1.4	1.2	1
F_Score	96.44	97.48	96.51	97	97.53

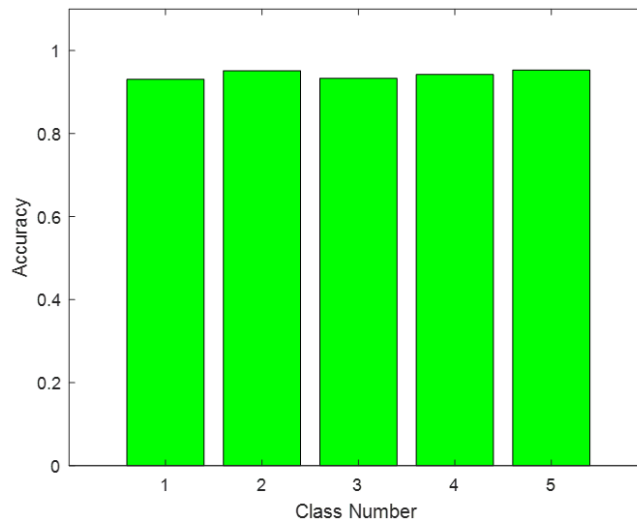


Figure 10: Accuracy graph for Caltech dataset with SVM

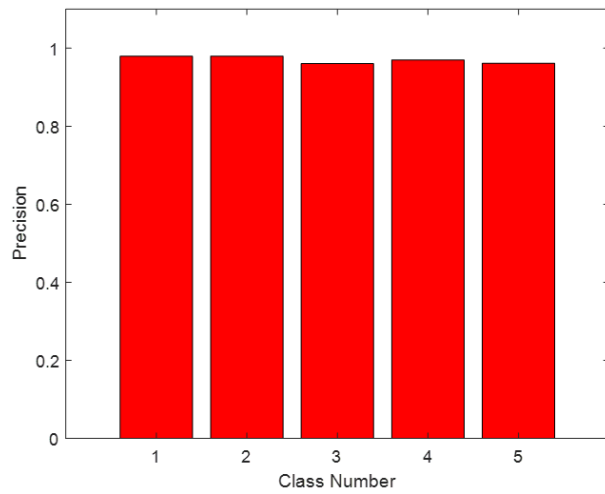


Figure 11: Precision graph for Caltech dataset with SVM

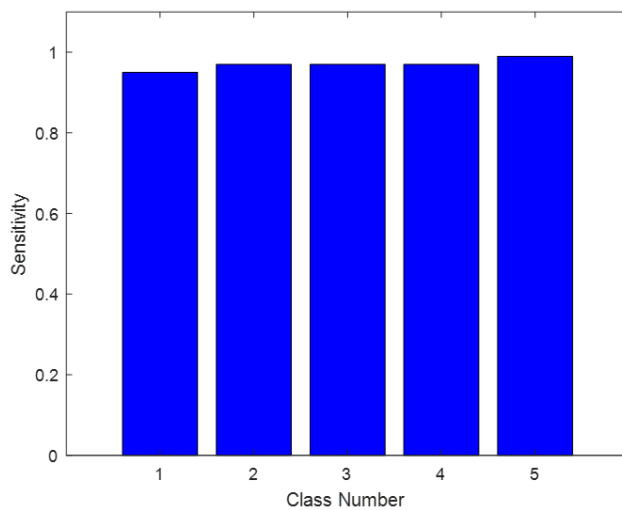


Figure 12: Sensitivity graph for Caltech dataset with SVM

**Confusion Matrix**

Output Class	an	di	fe	ha	ne	sa
an	96 19.2%	2 0.4%	0 0.0%	0 0.0%	0 0.0%	98.0% 2.0%
di	1 0.2%	98 19.6%	1 0.2%	0 0.0%	0 0.0%	98.0% 2.0%
fe	0 0.0%	0 0.0%	97 19.4%	3 0.6%	0 0.0%	97.0% 3.0%
ha	0 0.0%	0 0.0%	2 0.4%	97 19.4%	1 0.2%	97.0% 3.0%
ne	3 0.6%	0 0.0%	0 0.0%	0 0.0%	99 19.8%	97.1% 2.9%
sa	96.0% 4.0%	98.0% 2.0%	97.0% 3.0%	97.0% 3.0%	99.0% 1.0%	97.4% 2.6%
Target Class	an	di	fe	ha	ne	sa

Figure 13: For Corel dataset with random forest

Figure 13 displays a CM plot for Corel dataset. As per the dataset, it exhibits five classes for the five picture category. Calculation for each class is given as follows:

Table 3: Result for Corel dataset using random forest classifier

Parameter	'an'	'di'	'fe'	'ha'	'ne'
TP	96	98	97	97	99
FP	4	2	3	3	1
FN	2	2	3	3	3
TN	398	398	397	397	397

Parameter	'an'	'di'	'fe'	'ha'	'ne'
Accuracy	98.8	99.2	98.8	98.8	98.8
Precision	96	98	97	97	99
Recall	98	98	97	97	97.1
MissRate	2	2	3	3	2.9
ErrorRate	1.2	0.8	1.2	1.2	0.8
F_Score	96.96	98	97	97	98.01

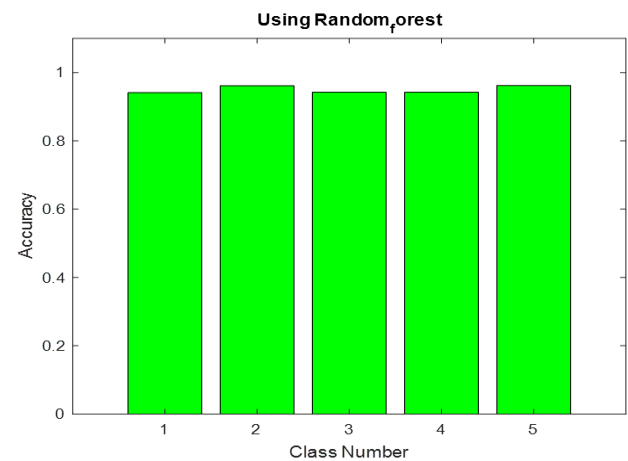


Figure 14: Accuracy graph for Corel dataset with random forest



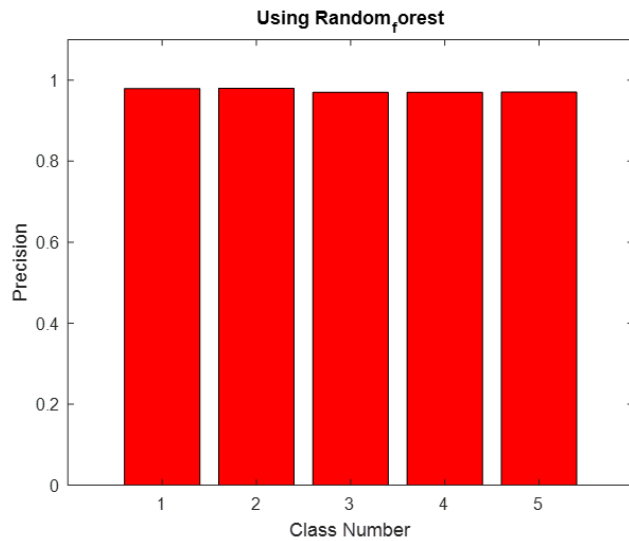


Figure 15: Precision graph for Corel dataset with random forest



Figure 16: Sensitivity graph for Corel dataset with random forest

Figure 17 displays a CM plot for Caltech dataset. As per the dataset, it exhibits five classes for the five picture category. Calculation for each class is given as follows:

Table 4: Result for Caltech dataset using random forest classifier

Parameter	1	2	3	4	5
TP	99	98	100	98	99
FP	1	2	0	2	1
FN	0	1	3	0	2
TN	400	399	397	400	398

Parameter	1	2	3	4	5
Accuracy	99.8	99.4	99.4	99.6	99.4
Precision	99	98	100	98	99
Recall	100	99	97.1	100	98
MissRate	0	1	2.9	0	2
ErrorRate	0.2	0.6	0.6	0.4	0.6
F_Score	99.49	98.49	98.52	98.98	98.5

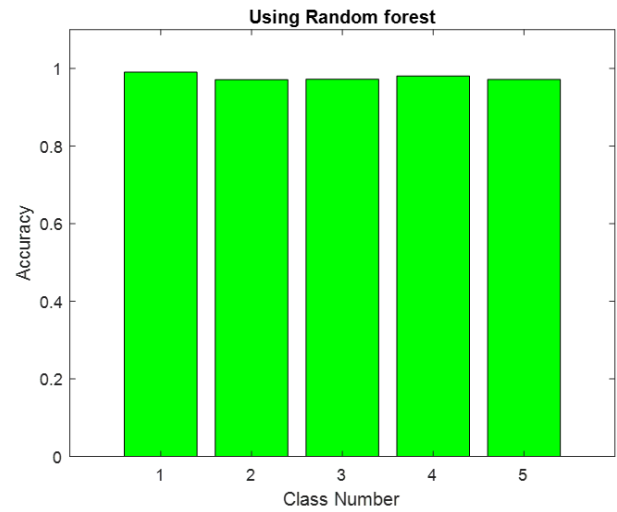


Figure 18: Accuracy graph for Caltech dataset with random forest

		Confusion Matrix					
		1	2	3	4	5	
Output Class	1	99 19.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	2	0 0.0%	98 19.6%	0 0.0%	0 0.0%	1 0.2%	99.0% 1.0%
	3	0 0.0%	2 0.4%	100 20.0%	1 0.2%	0 0.0%	97.1% 2.9%
	4	0 0.0%	0 0.0%	0 0.0%	98 19.6%	0 0.0%	100% 0.0%
	5	1 0.2%	0 0.0%	0 0.0%	1 0.2%	99 19.8%	98.0% 2.0%
			99.0% 1.0%	98.0% 2.0%	100% 0.0%	98.0% 2.0%	99.0% 1.0%
		Target Class					
		1	2	3	4	5	

Figure 17: For Caltech dataset with random forest

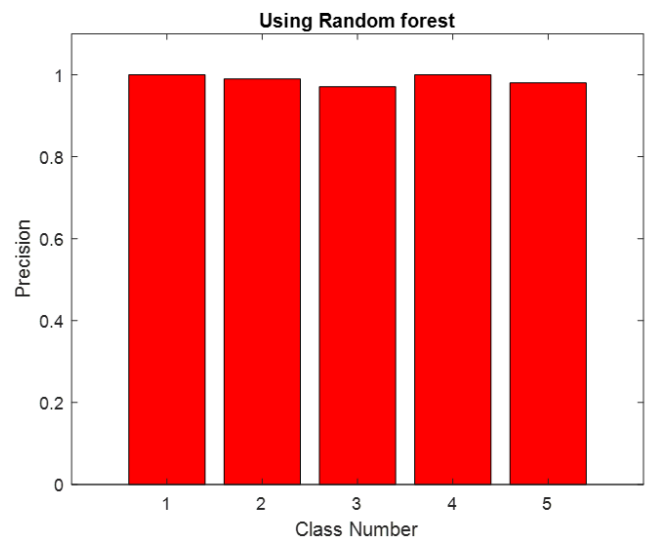
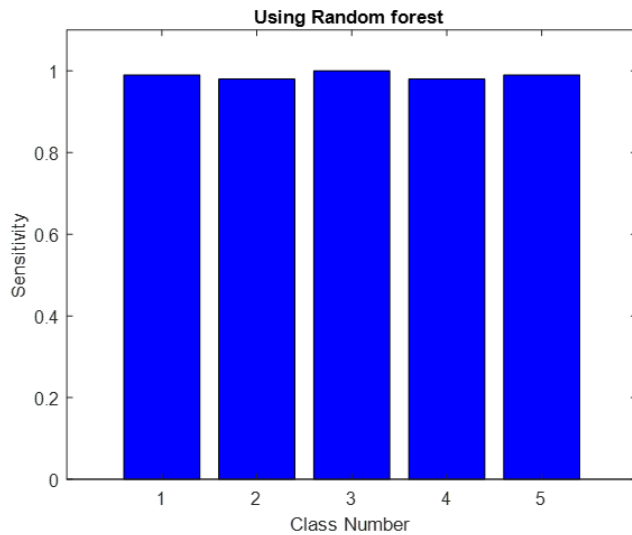


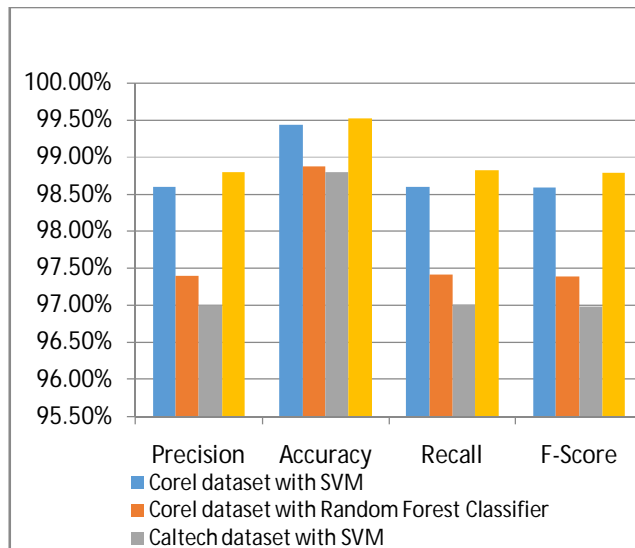
Figure 19: Precision graph for Caltech dataset with random forest



**Figure 20:** Sensitivity graph for Caltech dataset with random forest

**Table 5:** Comparative analysis

Dataset and Method	Precision	Accuracy	Recall	F-Score
Corel dataset with SVM	98.6%	99.44%	98.6%	98.59%
Corel dataset with Random Forest Classifier	97.4%	98.88%	97.42%	97.39%
Caltech dataset with SVM	97%	98.8%	97.01%	96.99%
Caltech dataset with Random Forest Classifier	98.8%	99.52%	98.82%	98.79%



**Figure 21:** Performance graph of SVM and Random Forest classifier

On observing the Table 5, it was found that both the classifiers have nearly similar performance. It reveals that the SVM outperforms the Random Forest classifier for Corel database while the Random Forest classifier outperforms the SVM for Caltech dataset.

**5. CONCLUSION**

A CBIR method has been build up that addressed all stages of this process, making use of recent advances in Random forest and SVM. Further PCA is applied to reduce the training overhead for redundant information filtering. Results reported by recommended method are better than those were achieved using traditional techniques. The performance on international databases validated the effectiveness of the method, making it a point of heading for future work in this area. It is intended to continue developing the method with the addition of stages for pre-processing and integration of the descriptors obtained, in others with a higher degree of abstraction.

**REFERENCES**

1. Zhou W., Li H. and Tian Q., “Recent advance in content-based image retrieval: A literature survey”, . arXiv preprint arXiv:1706.06064, 2017.
2. Tunga, S., Jayadevappa D. and Gururaj C., “A comparative study of content based image retrieval trends and approaches”, International Journal of Image Processing (IJIP), 9(3), pp.127-155, 2015.
3. Anavi Y., Kogan I., Gelbart E., Geva O. and Greenspan H., “Visualizing and enhancing a deep learning framework using patients age and gender for chest x-ray image retrieval”. Medical Imaging 2016: Computer-Aided Diagnosis, International Society for Optics and Photonics, Vol. 9785, p. 978510, 2016.
4. Dhara A.K., Mukhopadhyay S., Dutta A., Garg M. and Khandelwal N., “Content-based image retrieval system for pulmonary nodules: Assisting radiologists in self-learning and diagnosis of lung cancer”. Journal of digital imaging, 30(1), pp.63-77, 2017.
5. Mehre S.A., Dhara A.K., Garg M., Kalra N., Khandelwal N. and Mukhopadhyay S., “Content-based image retrieval system for pulmonary nodules using optimal feature sets and class membership-based retrieval”, Journal of digital imaging, 32(3), pp.362-385, 2019.
6. Patil J.K. and Kumar R., “Analysis of content based image retrieval for plant leaf diseases using color, shape and texture features”, Engineering in agriculture, environment and food, 10(2), pp. 69-78, 2017.
7. Belsha N. and Hariprasad N., “The enhanced content based image retrieval system and classification of infected vegetables”. Third International Conference on Sensing, Signal Processing and Security (ICSSS), IEEE, pp. 83-88, 2017.
8. Xia Z., Wang X., Zhang L., Qin Z., Sun X. and Ren K., “A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing”. IEEE transactions on information forensics and security, 11(11), pp.2594-2608, 2016.
9. Xia Z., Xiong N.N., Vasilakos A.V. and Sun X., “EPCBIR: An efficient and privacy-preserving content-based image retrieval scheme in cloud

- computing”, *Information Sciences*, 387, pp.195-204, 2017.
10. Sanu S.G. and Tamase P.S., “Satellite image mining using content based image retrieval”, *Int. J. Eng. Sci.*, 13928, 2017.
  11. Cai W., Song Y., Kumar A., Kim J. and Feng D.D., “Content-based large-scale medical image retrieval”, *Biomedical Information Technology*, Academic Press, pp. 321-368, 2020.
  12. Vakhitov A., Kuzmin A. and Lempitsky V., “Internet-Based Image Retrieval Using End-to-End Trained Deep Distributions”, *arXiv preprint arXiv*, 1612, 2016.
  13. Liu G.H. and Yang J.Y., “Content-based image retrieval using color difference histogram”, *Pattern recognition*, 46(1), pp.188-198, 2013.
  14. Lasmar N.E. and Berthoumieu Y., “Gaussian copula multivariate modeling for texture image retrieval using wavelet transforms”, *IEEE Transactions on Image Processing*, 23(5), pp.2246-2261, 2014.
  15. Wang X., Luo G. and Qin K., “A composite descriptor for shape image retrieval”, *International Conference on Automation, Mechanical Control and Computational Engineering*. Atlantis Press, 2015.
  16. Wang X.Y., Zhang B.B. and Yang H.Y., “Content-based image retrieval by integrating color and texture features”, *Multimedia tools and applications*, 68(3), pp.545-569, 2014.
  17. Chandrasekhar V., Lin J., Liao Q., Morere O., Veillard A., Duan L. and Poggio T., “Compression of deep neural networks for image instance retrieval”, *Data Compression Conference (DCC)*, IEEE, pp. 300-309, 2017.
  18. Yu W., Yang K., Yao H., Sun X. and Xu P., “Exploiting the complementary strengths of multi-layer CNN features for image retrieval”, *Neurocomputing*, 237, pp.235-241, 2017.
  19. Tzelepi M. and Tefas A., “Deep convolutional learning for content based image retrieval”, *Neurocomputing*, 275, pp.2467-2478, 2018.
  20. Suganya K., Pabboju S. and Babu A.V., “A CBIR classification using support vector machines”, *International Conference on Advances in Human Machine Interaction (HMI)* IEEE, pp. 1-6, 2016.
  21. Sarwar A., Mehmood Z., Saba T., Qazi K.A., Adnan A. and Jamal H., “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), pp.117-135, 2019.
  22. Singh V.P. and Srivastava R., “Improved Content-Based Image Classification Using a Random Forest Classifier”, *Advances in Computer and Computational Sciences*, Springer, Singapore, pp. 365-376, 2018.
  23. Müller H., Michoux N., Bandon D. and Geissbuhler A., “A review of content-based image retrieval systems in medical applications—clinical benefits and future directions”, *International journal of medical informatics*, 73(1), pp.1-23, 2004.
  24. Smeulders A.W., Worring M., Santini S., Gupta A. and Jain R., “Content-based image retrieval at the end of the early years”, *IEEE Transactions on pattern analysis and machine intelligence*, 22(12), pp.1349-1380, 2000.
  25. Müller H., Rosset A., Vallée J.P., Terrier F. and Geissbuhler A., “A reference data set for the evaluation of medical image retrieval systems”, *Computerized Medical Imaging and Graphics*, 28(6), pp.295-305, 2004.
  26. Park M. and Ramamohanarao K., “Automatic extraction of semantic concepts in medical images”, *International Conference on Image Processing*, 2004. *ICIP'04, IEEE*, Vol. 2, pp. 1157-1160, 2004.
  27. Haji M.S., Alkawaz M.H., Rehman A. and Saba T., “Content-based image retrieval: a deep look at features prospectus”, *International Journal of Computational Vision and Robotics*, 9(1), pp.14-38, 2019.
  28. Srivastava P., Khare M. and Khare A., “Content-based image retrieval using scale invariant feature transform and gray level co-occurrence matrix”, *Second International Workshop on Pattern Recognition*, International Society for Optics and Photonics, Vol. 10443, p. 104430V, 2017.
  29. Srivastava D., Rajitha B., Agarwal S. and Singh S., “Pattern-based image retrieval using GLCM”, *Neural Computing and Applications*, pp.1-14, 2018.
  30. Agarwal M. and Maheshwari R.P., “HOG feature and vocabulary tree for content-based image retrieval”, *International Journal of Signal and Imaging Systems Engineering*, 3(4), pp.246-254, 2010.
  31. Velmurugan K. and Baboo S.S., “Image retrieval using Harris corners and histogram of oriented gradients”, *International Journal of Computer Applications*, 24(7), pp6-10, 2011.
  32. Kumar M., Chhabra P. and Garg N.K., “An efficient content based image retrieval system using BayesNet and K-NN”. *Multimedia Tools and Applications*, 77(16), pp.21557-21570, 2018.
  33. Sinha U. and Kangarloo H., “Principal component analysis for content-based image retrieval”, *Radiographics*, 22(5), pp.1271-1289, 2002.
  34. Reddy K.R. and Narayana M., “A comparative study of sift and PCA for content based image retrieval”, *Inter. Refereed J. Eng. Sci.(IRJES)*, 5(11), pp.12-19, 2016.
  35. Breiman L., *Random forests*. *Machine learning*, 45(1), pp5-32, 2001.
  36. Wang J.Z., Li J. and Wiederhold G., “SIMPLcity: Semantics-sensitive integrated matching for picture libraries”, *IEEE Transactions on pattern analysis and machine intelligence*, 23(9), pp.947-963. 2001.
  37. Patil P. and Sunag B., “Analysis of image retrieval techniques based on content”, *IEEE International Advance Computing Conference (IACC)* IEEE, pp. 958-962, 2015.
  38. Gali R., Dewal M.L. and Anand R.S., “Genetic algorithm for content based image retrieval”, *Fourth International Conference on Computational Intelligence, Communication Systems and Networks* IEEE, pp. 243-247, 2012.
  39. Griffin G., Holub A. and Perona P., *Caltech-256 object category dataset*, 2007.
  - 40.