Identifying and Indexing Near-Duplicate Images Using Optimizing Technique in Web Search

[#]S.Thaiyalnayaki, [*]J.Sasikala

[#]Assistant Professor, Computer Science and Engineering, DhanalaksmiSrinivasan College Of Engineering and Technology,Deparment of computer science and Engineering,
Anna University, Chennai, India[*]Assistant Professor, Departmentof computer science and Engineering,AnnamalaiUniversity,Chidambaram,India

***Abstract*—** Today's World Wide Web is growing drastically and duplicates occur in many fields. Importantly duplicate images that are uploaded into internet like a food product, document image, medical images, textile fields etc. So it becomes very important to identify those duplicate images. Near duplicates can be similar copies or differ a little in their visual content. Duplicate images introduce many problems of redundancy and copyright infringement in large set of image collections. This paper proposes a methodology for identifying and indexing the near duplicate images on web and optimizing the results. First step is to get the search image from the user and enhance the search image and then Features are extracted from search image using SURF (Speeded up Robust Features) that is to extract the local invariant features of search image. After this calculate the similarity measured among the features extracted images using sim-hash algorithm and then indexing Near duplicate images based on user's search image using Locality Sensitive Hashing (LSH). And finally optimizing the results using Particle swarm optimization (PSO).We demonstrate that our identifying and indexing approach is highly effective for collections of up to a few hundred thousand images.

Keywords:*Indexing,near-duplicates, near-duplicatedetection, Image Enhancement,pso*

1. INTRODUCTION

Near duplicate images carry both informative and redundant signals providing rich visual clues for indexing and summarizing images from different sources. The excessive amount of near duplicates streamed over Internet demands scalable techniques for copyright infringement detection, content monitoring for forensic applications and advertisement tracking. As a result, there is strong interest from industry and governmental agencies in Web-scale search, elimination, detection, and use of near duplicates search for various multimedia applications. This special issue presents some of the most recent advances in the research on Web-scale near-duplicate search and also explores the potential for bringing this research a substantial step further. It contains higher quality contributions addressing various aspects of the Web scale near-duplicate search problem in a number of relevant domains. In this paper, identifying and indexing the near-duplicate images are detected based on user query image and retrieving the near duplicate image based on indexing. This process is achieved by four steps; in first step, Features are extracted on the user search image. Second step is after extracting the features of each images similarity is calculated. Third step is to Form indexing of near duplicate images based on user search image. And finally optimize the results. For indexing we use Locality Sensitive Hashing (LSH) No explicit distinction is made between these two types and simply uses the term duplicates to refer to them both.

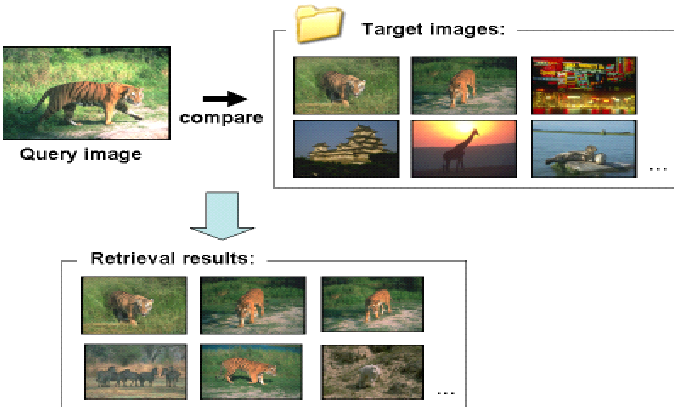

Fig. 1 Retrieved Result

2.Related Works

[1] extended the concept of Chinese information retrieval, it is easy to index a Chinese text document for retrieval. We just need to segment the text document into phrases. When the document is Chinese document image (non-ASCII file), we may first convert the document image into text file by using Chinese optical character recognition (OCR) technology, and then index the document by using information retrieval algorithm. However, OCR needs longer time, which can influence retrieval efficiency. First segment the document image to get all the Chinese character images,then calculate stroke density of each Chinese character image, and at last attain stroke density code of the character image. The index method has the advantage of speed and robustness to noise. In addition, this paper also offers retrieval method for Chinese document image based on the index technology. [2] Presents the paper provide a comparative study on how well content-based duplicate image detection methodsare able to detect the duplicates of a query image. We conduct a survey to better understand in which ways such images on the internet differ from each other and use these observations to form a realistic and challenging duplicate image detection scenario. The methods evaluate in our study are representative techniques.Our evaluations show that to obtain high accuracy it is not necessary to use a large nor computationally intensive image descriptor. We also presented results per transformation to gain further insight into the strengths and weaknesses of the methods.[3]conclude final ResuIt is the compare of classification output of both classifiers in tenns of classification efficiency. We find that ANN with dmey wavelet give highest classification efficiency with bath training and testing data set.Db4 based ANN also give good classification result for training data set but the performance of Db4 based ANN is poor for testing data as compared to Demy based ANN.Haar based Ann and KNN testing dataset.In case of KNN based classifier Dmey based KNN give batter result as compared to Db4. Dmey wavelet based ANN gives batter classification result the overall classification efficiency compared to all wavelet based KNN[5]This paper has presented a study on rock texture image classification using support vector machines (and also K-nearest neighbors and decision trees) with the aid of feature selection techniques. It has offered both unsupervised and supervised methods for feature selection, based on data reliability and information gain ranking respectively. Following this approach, the conventional classifiers which are sensitive to the dimensionality of feature patterns, become effective on classification of images whose pattern representation may otherwise involve a large number of features. Although the images encountered are complex, the resulting feature pattern dimensionality of selected features is manageable. Classifiers built using such selected features generally outperform their counterparts that employ the full set of original features which has a dimensionality several folds higher than that of the selected feature subsets. This is confirmed by systematic experimental investigations. In this paper, we have presented a new image classification and retrieval approach that is based on the concept of correlation. In this approach, images are classified through an off-line process on the basis of their cross correlation with other images in the database. Images with maximum cross relation are recursively grouped in the same class.

The resultant hierarchy is maintained as a binary tree in which each root node represents the mean of the images in its sub trees such that theleaf nodes contain maximally correlated images, thus, making the retrieval process very efficient.We are well aware of the fact that the classification process in our scheme is very expensive.[7] We address the problem of different kinds of invariance (rotation,shift, and scale) in image classification and propose a scheme to extract shift invariant wavelet features for classification of images with different sizes. The proposed wavelet energy features, which were obtained from the result of a normalization and an adaptive shift-invariant wavelet packet transform

3. PROPOSED WORK

Identification of duplicate images consists of five steps. a) Image Enhancement b)Speeded Up Robust Features(SURF) c) Sim-Hash Algorithm d)Locality Sensitive Hashing e) Particle Swarm Optimization(PSO)

Locality-sensitive hashing means focus on pairs of signatures likely to be similar.

3.1Dataset

In this paper,Actually this dataset contains real camera photos taken directly from a real user’s personal photo collection. It consists of many different types of near duplicates. Initially around 20 images were taken by the proposed system and later the entire dataset is used.

3.2 Image Enhancement

Image enhancement is the process of adjusting digital images so that the results are more suitable for display or further image analysis. For example, you can remove noise, sharpen, or brighten an image, making it easier to identify key features

3.3SPEEDED UP ROBUST FEATURES(SURF)

In SIFT for key point detection and description. But it was comparatively slow.but processing a image and videos people needed more speeded-up version. In 2006, three people, Bay, H., Tuytelaars, T. and Van Gool, L, published another paper, “SURF: Speeded Up Robust Features”.It is also called as speeded-up version of SIFT. SIFT and SURF algorithms employ slightly different ways of detecting features [9]. SIFT builds an image pyramids, filtering each layer with Gaussians of increasing sigma values and taking the difference.on the other hand, SURF creates a “stack” without 2:1 down sampling for higher levels in the pyramid resulting in images of the same resolution [9]. Due to the use of integral images, SURF filters the stack using a box filter approximation of second-order Gaussian partial derivatives, since integral images allow the computation of rectangular box filters in near constant time [6].In keypoint matching step, the nearest neighbor is defined as the keypoint with minimum Euclidean distance for the invariant descriptor vector. Lowe used a more effective measurement that obtained by comparing the distance of the closest neighbor to that second-closest neighbor [4] so the author of this paper decided to choose 0.5 as distance ratio like Lowe did in SIFT.

Local feature matching has become an increasingly used method for comparing images. Various methods have been proposed the usage of Speeded Up Robust Features (SURF)are analysed as local descriptors for image recognition. The effect of different feature extraction and viewpoint consistency constrained matching approaches is analysed. Matches between the geo-referenced database images and those captured in realtimeare locating by employing the fast SURF algorithm. The most important property of an interest point detector is its repeatability. The repeatability provides the reliability of a detector for finding the similar physical interest points under different viewing conditions.

The repeatability measurement is computed as a ratio between the number of point-to-point correspondences that can be established for detected points and the mean number of points detected in two images [4]:

$$r_{1,2}= \left(\frac{C(I_1, I_2)}{mean(m_1, m_2)} \right) \longrightarrow \quad (1)$$

3.4 SIM-HASH ALGORITHM

Locality sensitive hashing (LSH) [13] is a general framework of indexing technique, devised for capably solving the approximate nearest neighbour search problem [11]. The work of LSH largely depends on the underlying particular hashing methods. Two popular hashing algorithms are MinHash [12] and SimHash (sign normal random projections) [8]. MinHash is an LSH for its resemblance similarity which is defined over binary vectors, while SimHash is an LSH for cosine similarity which works for general real-valued data and the collision probability of SimHash is a function of cosine similarity (S).

SimHash is another popular LSH for its cosine similarity measure, which originates from the concept of sign random projections(SRP)[8]. Given a vector x and SRP utilizes a random vector w with each component generated from i.i.d. normal, i.e., wi~N(0, 1), and it only stores the sign of the projected data. Formally, SimHash is given by

$$h_w^{sim}(x) = sign(w^T x) \longrightarrow \quad (2)$$

SimHash generates a single bit output (only the signs) whereas MinHash generates an integer value.

In this section, it is described how a method originally developed for text near-duplicate detection and it is adapted to near-duplicate detection of images. Two images are near duplicate if the similarity Sims is higher than a given threshold. The goal is to retrieve all documents in the database that are similar to a query images. The distance measure between two images is computed as the similarity ofsets w1 and w2, which is defined as the ratio of the number of elements in the intersection over the union:

$$sim(w1,w2)= \frac{qi \cap wi}{qi U wi} \longrightarrow \quad (3)$$

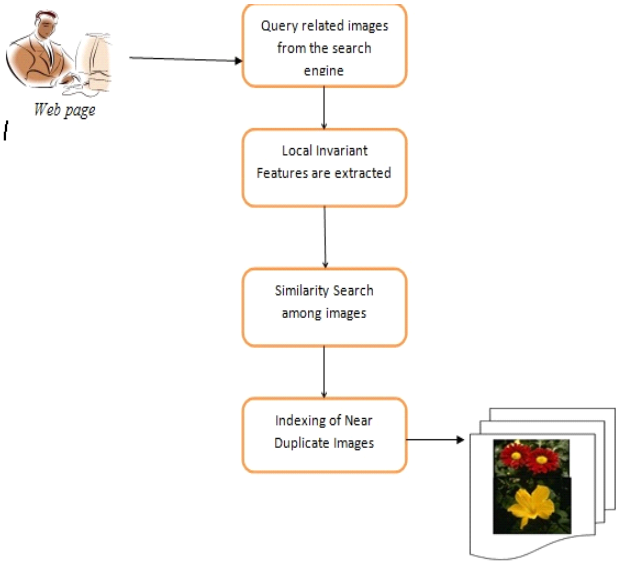
Web page
Query related images from the search engine
Local Invariant Features are extracted
Similarity Search among images
Indexing of Near Duplicate Images

Fig. 2 Intermediate Result Expectation

3.5 LOCALITY SENSITIVE HASHING

Locality-sensitive hashing (LSH), proposed by Indyk&Motwani [10], is an approximate similarity search technique that works efficiently even for high-dimensional data. Classical data structures for similarity search deteriorate from the curse of dimensionality, in that they scale poorly for data with dimensions greater than 20, where it is performed no better than an exhaustive linear search through the entire database. It has been proven that LSH out-does tree-based structures such as the Sphere/Rectangle-tree (SR-tree)by at least an order of magnitude. Given that the data consists of many, high-dimensional (36-dimensional) feature vectors, LSH becomes an attractive indexing scheme. A popular algorithm for LSH, introduced by Gioniset al. [7] conceptually transforms each point p into a binary vector by concatenating the unary representation of each (discretised) coordinate of p. The resulting bit string is a point in a high-dimensional Hamming space, where L1 orbits between points in the original space are preserved. Hash functions that usually select a subset of the bits that satisfy the craved locality-sensitive properties. The algorithm builds a set of l such hash functions, each of which selects k bits from the bit string (each function uses a different, randomly selected set of k bits). These k bits are hashed once more to index into the buckets in the hash table, and a 32-bit checksum hash value is also generated. The two parameters, k and l enable the designer to select an appropriate trade-off between accuracy and running time.

Given a metric space (X; k _ k) and a database S _ X,for any given query v 2 X, the k-nearest neighbour algorithmcomputes a set of k points I(v) _ S that are closest to v. Weassume that X is embedded in a D-dimensional Euclideanspace RD and each item is represented as a high-dimensionalvector, i.e., v = (v1,…………….,vD).The basic LSH algorithm is an approximate method tocompute nearest neighbors, which uses M (M <<D) hashfunctions h1(.),…….., hM(.) to transform RD into a lattice spaceZM and distributes each data item into one lattice cell:

H(F)=h1(F),h2(F),h3(F),……………hn(F).

The lattice space is usually implemented as a hash table,since many of the cells may be empty. LSH algorithms havebeen developed for several distance measures, such as lpdistance.Forlp space

This algorithm includes following steps:

3.6 Algorithm For Simhash and LSH

Procedure calculating Resemblance

1.Input:Features of Search image qi like sf1,sf2,sf3,………………,sfn

Features of Web image wi like w11,w12,w13,………………,w1n,
w21,w22,w23,………………..w2n,
w31,w32,w33,………………..w3n,
wi1,wi2,wi3,………………..win

```
3.for all images F=1,……………K do
            If(sfF==wiF) then
            Increment the rem[i]
            Increment the Features
        End
End
4.Ifsim[i]==k then
        EDI = Wif
  Else
         NDI=Wif
End
5.Increment the image I;
```

Efficient Disk Access

Locality-sensitive hashing was originally designed to work efficiently in memory, where random access is fast. For large datasets, one must store the database on disk, and a naive implementation of LSH fails badly. This is because random access on disk is extravagant, on the order of 10ms per seek. Multiple queries into a hash table, by definition, requires random seeks on disk. Initial experiments revealed that querying the database for the key points from just one image took several minutes, indicating that the standard LSH implementation could never be practical for the problem. The key difference between the system and other systems that use LSH for other applications is that all of the queries occur in batches of hundreds or thousands (corresponding to all of the key points in the query image).key points are extracted from the query image, and search on the entire set of to determine if any of them match the key points in the database. An earlier disk-based implementation of LSH by Gioniset al was designed for efficient single point queries rather than the batch queries required by the system. Since disk seek times are the bottleneck, our approach relies on organizing the batch queries so as to minimize the motion of the disk heads. Hence it is done by pre computing all of the hash bins that are needed to access, sort them, and access them in sequential order. Reducing the disk head motion in this manner translates to a dramaticimprovement in effective seek time — cutting it to approximately 1ms per seek. Gionisetal.Also suggested in lining the data in the hash table instead of storing only the pointers as one would for an in-memory implementation. The goal was to halve the number of seeks because one would not need to follow a pointer to the actual data. However, for the application, in lined data led to a massive increase in required disk space (20xfor our dataset) and actually slowed our search. Since the searches do not require random seeks, better performance can be achieved by employing a small hash table with an auxiliary key point database (and scanning both in-order) rather than a large hash table with in lined data. All of these components are required to make the system practical. The use of robust interest point detectors and distinctive local descriptors enables us to query images with high recall and precision. By using locality-sensitive hashing and optimizing the data layout on disk, interactive response times for queries are achieved.

3.7 Overall Proposed System

The proposed system is used to identify and indexing the near duplicate images and similar duplicate images corresponding to the user search image;
Overall Block Diagram of the Proposed System is

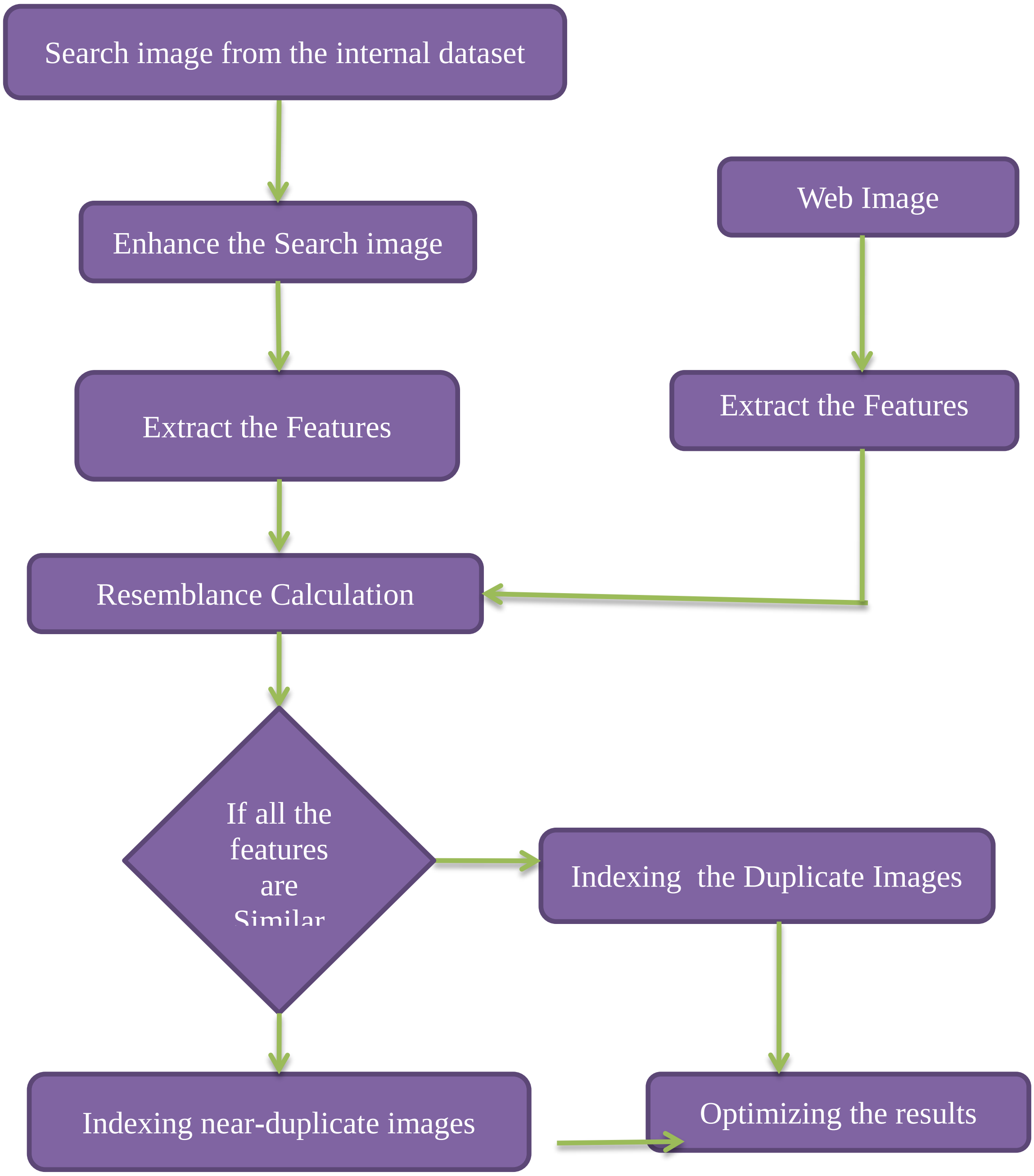
Search image from the internal dataset
Enhance the Search image
Extract the Features
Resemblance Calculation
If all the features are Similar
Indexing near-duplicate images
Web Image
Extract the Features
Indexing the Duplicate Images
Optimizing the results

Fig. 3 Overall Block Diagram of Proposed System

The steps in Proposed Work can be depicted using the flow chart –

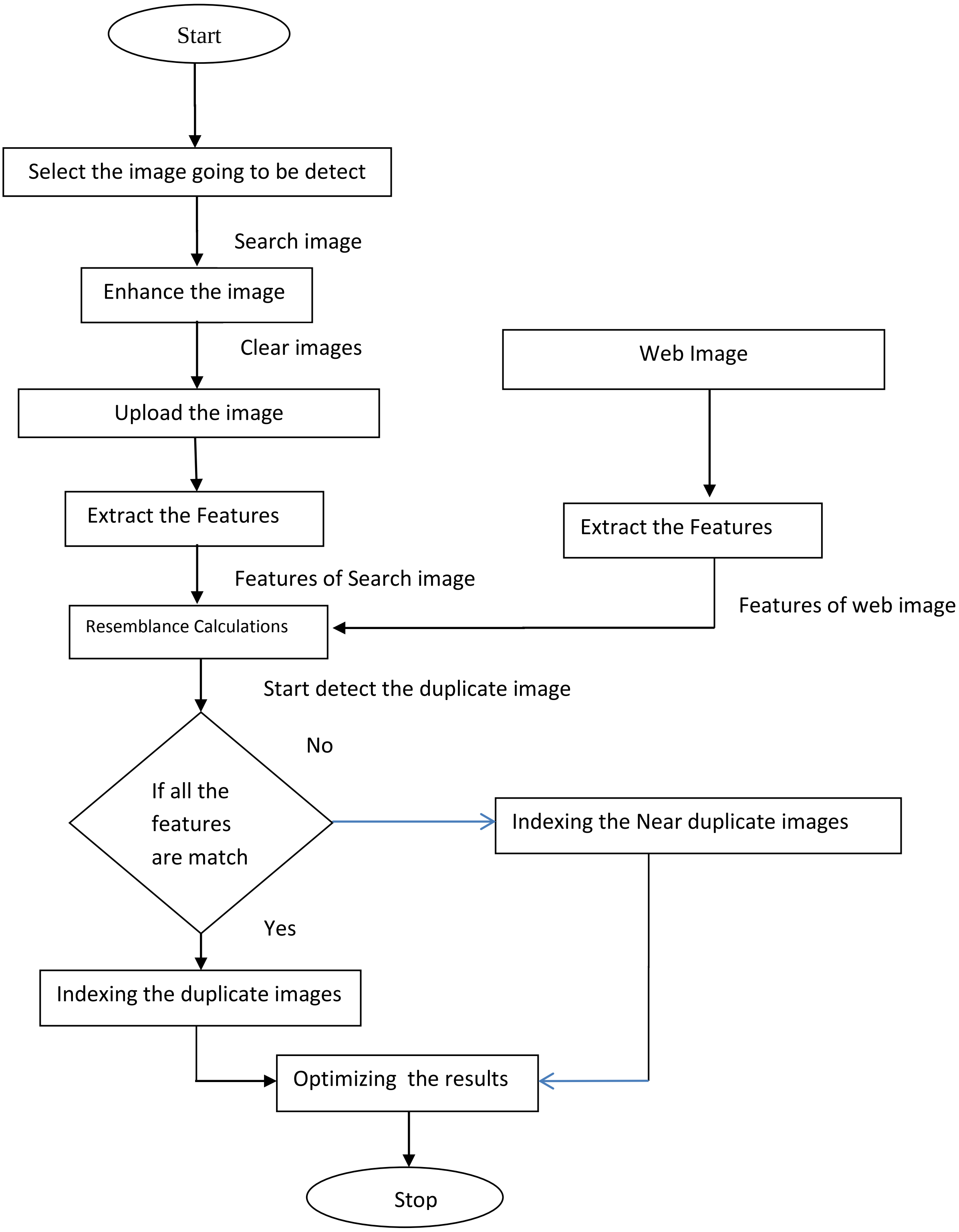

Fig. 4 Flowchart of Proposed System

3.8 Particle Swarm Optimization(PSO)

Particle Swarm Optimization(PSO) is a swarm intelligence based met heuristic algorithm proposed by Kennedy and Eberhart [14] which takes its inspiration from the cooperation and communication of a swarm of birds.The intelligence which emerges from such behaviour causes the swarm to mimic complex global patterns. Below we describe general concepts of PSO. In PSO,each individual in the swarm,called a particle,behaves like an agent of a highly decentralized and intelligent environment. Each particle of the swarm contributes to the environment by following very simple rules,thus cooperating and communicating with other particle so the swarm.A complex global collective behaviour merges in the swarm.This complex global behavior is exploited to solve a complex optimization problem. High decen- tralization,cooperation amongs the particles and simple implementation make PSO efficiently applicable to optimization problems [17,15,16]. PSO has three main components,particles,social and cognitive components of the particles, and the velocity of the particles.In a problem space where there may be more than one possible solution and the optimal solution of the problem is required,a particle represents an individual solution to the problem.The learning of the particles comes from two sources,one is from a particle's own experience called cognitive learning and the other source of learning is the combined learning of the entire swarm called social learning.Cognitive learning is represented by personal best(pBest) and social learning is represented by the global best (gBest)value.The pBest solution is the best solution the particle has ever achieve dinits history.The gBest value is the best position the swarm has ever achieved.The swarm guides the particle using parameter gBest. Together cognitive and social learning are used to calculate the velocity of particles to their next position. When applied to optimization problems,a typical PSO algorithm starts with the initialization of a number of parameters.One of the important initializations is selecting the initial swarm.The number of particles in the swarm depends upon the complexity of the problem. An initial choice of solutions is normally made randomly.

4. Experimental Results

This paper is proposed mainly for identity and detect the near duplicate images by using SURF and Simhash algorithm. In this paper, the near-duplicate images are detected based on user query image and Retrieving the near duplicate image based on indexing. This process is achieved by four steps, First Features are extracted on the user search image. Second is after extracting the features of each images similarity is calculated.Third,Form indexing of near duplicate images based on user search image.Finally,optimizing the results. For indexing we use Locality Sensitive Hashing (LSH) No explicit distinction is made between these two types and simply use the term duplicates to refer to them both

The below figure show upload the enhanced users search image for web search

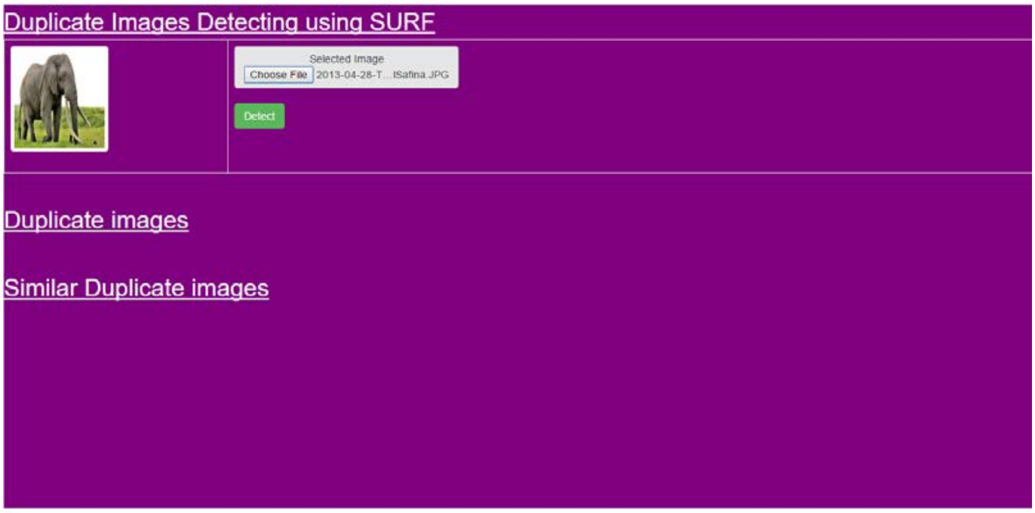

Fig. 5.Uploading the users search image

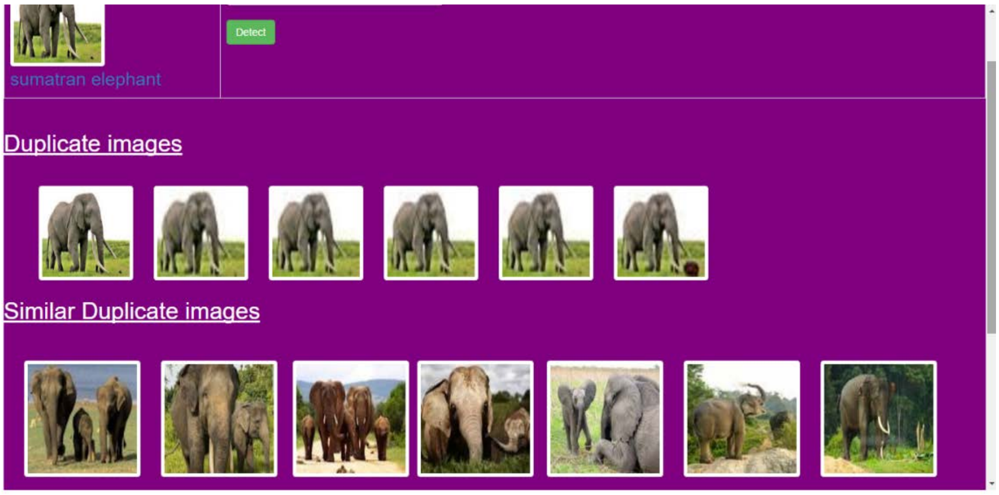

Fig. 6 Indexing the duplicate images and similar duplicate images

5.CONCLUSION

The overall work here is identifying near duplicate images and indexing those images from a collection of dataset. In this paper, a methodology is presented for identifying and indexing of near-duplicate images. Initially, the search image is passed by the user to the search engine and the search engine results in set of query related images. These images contain duplicate as well as near-duplicate images. Here we concentrate in detecting near-duplicate images and index those images. This is done using following steps – initially enhance the user query image and then extract the feature. After features are extracted Similarity is measured and finally indexing the near duplicate images and also optimize the results. This results in indexing of images. We conclude that our indexing approach is highly effective for collections of up to a few hundred thousand images.

REFERENCES

[1]Yaodong He, Zao Jiang, Bing Liu and Hong Zhao**,** content-Based Indexing and Retrieval Method of Chinese Document Images,Shenyang,China

[2] Bart Thomee, Mark J. Huiskes, Erwin M. Bakker, Michael S. LewAN EVALUATION OF CONTENT-BASED DUPLICATE IMAGE DETECTION METHODS FORWEB SEARCH

[3] DharmendraPatidar, Nitin Jain, Ashish ParikhPerformance Analysis of Artificial Neural Networkand K Nearest Neighbors Image ClassificationTechniques with Wavelet features, 2014 IEEE International Conference on Computer Communication and Systems(ICCCS '14), Feb 20-21, 2014, Chennai, ThlDIA

[4] D. Lowe.*"Distinctive Image Features from Scale-Invariant Keypoints", IJCV*,60(2):91–110, 2004.

[5]Changjing Shang and Dave Barnes,Support Vector Machine-BasedClassification of Rock Texture ImagesAided by Efficient Feature Selection, WCCI 2012 IEEE World Congress on Computational IntelligenceJune, 10-15, 2012 - Brisbane, Australia

[6] Bay,H,. Tuytelaars, T., &Van Gool, L.(2006). "*SURF: Speeded Up Robust Features*", *9th European Conference on Computer Vision*.

[7] Chi-Man Pun and Moon-ChuenLee,Extraction of Shift Invariant Wavelet Featuresfor Classification of Images with Different Sizes, 2009 International Conference on Environmental Science and Information Application Technology

[8] Moses S. Charikar. Similarity estimation techniques from rounding algorithms. In STOC, pages 380–388, Montreal, Quebec, Canada, 2002.

[9] Yang zhan-long and Guo bao-long. "*Image Mosaic Based On SIFT",*International Conference on Intelligent Information Hiding and Multimedia Signal Processing,pp:1422-1425,2008.

[10] OndrejChum,JamesPhilbin,MichaelIsard,AndrewZisserman,Scalable Near Identical Image and Shot Detection University of Oxford,Silicon Valley

[11] Jerome H. Friedman, F. Baskett, and L. Shustek.An algorithm for finding nearest neighbors. IEEE Transactions on Computers, 24:1000–1006, 1975.

[12] Andrei Z. Broder. On the resemblance and containment of documents. In the Compression and Complexity of Sequences, pages 21–29, Positano, Italy, 1997.

[13]PiotrIndyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In STOC, pages 604–613, Dallas, TX, 1998

[14] J.Kennedy,R.Eberhart,Particleswarmoptimization,in:Proceedingsofthe IEEE InternationalConferenceonNeuralNetworks,vol.4,IEEE,1995, pp. 1942–1948.

[15] J.F.Kennedy,J.Kennedy,R.C.Eberhart,SwarmIntelligence,MorganKaufmann, San Francisco,CA.USA,2001.

[16] R. Poli,J.Kennedy,T.Blackwell,Particleswarmoptimization,SwarmIntell.1 (1) (2007)33–57.

[17] A.P.Engelbrecht,FundamentalsofComputationalSwarmIntelligence,vol.1, Wiley, Chichester,2005.