Original Research Paper Engineering

Re- Indexing and Identifying Near-Duplicate Images Using Optimizing Technique

S.Thaiyalnayaki Assistant Professor, Computer Science and Engineering, DhanalaksmiSrinivasan College Of Engineering and Technology, Deparment of computer science and Engineering, Anna University, Chennai, India

J.Sasikala Assistant Professor, Departmentof computer science and Engineering, AnnamalaiUniversity,Chidambaram,India

ABSTRACT

Image tracking technique is being applied more broadly in fields like robotics, human-computer interaction, security surveillance and other areas in recent years [7-9], it has attracted more and more researchers. As the most important segment of image tracking, the image matching becomes one hot topic in the current field of computer vision. The image matching is that the key point could be recognised between two ro more images by some matching algorithm. The images can use in matching often in different time, different perspectives, different scale. Near duplicates can be a similar copy 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 then indexing the near duplicate images on web and Re-indexing the near duplicate images. First step is to get the search image from the user and enhance the search image and then Features are extracted from search image as well as using SURF (Speeded up Robust Features) that is to extract the local invariant features of search image. After this calculate the similarity among the features extracted images using same SURF algorithm and then indexing Near duplicate images based on user's search image using Locality Sensitive Hashing (LSH). And finally optimizing result by Re-indexing the near duplicate images. 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-duplicate detection, Image Enhancement,Re-indexing

1. INTRODUCTION

World Wide Web having billions of images and videos. User browsing the internet will quickly encounter duplicate images as well as near duplicates in multiple locations. Duplicate image detection is important for reducing storage space, understanding behaviour and interest of user and for copyrights. Duplicates can be exact duplicates, or near duplicates. Exact duplicate images have exactly the similar appearance that is images with identical contents. Near duplicate means little changes are present in the original image such as rotation, cropping, and transforming, adding, deleting and altering image content. 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 to 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. After indexing, first image of near duplicate images is taken and extract the features then compare those features and Re-indexing the near duplicate images. This process is achieved by four steps; in first step, Features are extracted on the user search image and web images. Second step is after extracting the features similarity is calculated between each web images and search image. Third step is to Form indexing of near duplicate images and exact duplicate image based on user search image. For indexing we use Locality Sensitive Hashing (LSH) And finally optimizing result by Re-indexing the near duplicate images. No explicit distinction is made between these two types and simply uses the term duplicates to refer to them both.

2. RELATED WORKS

[1]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 features selection, based on data reliability and information gain ranking respectively.[2]Image Retrieval is one of the ongoing research areas, the main reason for this evolution is that the amount of images in internet domain is increasing exponentially. So those techniques which had been proposed earlier cannot be implemented to this amount of data. In this paper we are proposing a work of automatic image classification of two different groups of images such as flower image and sports image. Our major research work is on retrieval of image by giving both image and text as input. For which we use Multi-Modal Ontology searching technique where the Domain Ontology of selected domain say Flower and Sports image ontology is matched with its Feature ontology. For this a Preprocessing of general classification of two kind of image is required which is been proposed in this paper. In the forthcoming research we would try to integrate this Ontology for better Image Retrieval system.[3]In this paper we have introduced and tested the matching algorithm with descriptor length 36 as the matching algorithm for VBN depending on a lower number of interest point matches between real-time captured images and those from a database. Additionally, the samples count in the sub-divisions with the different descriptor length (36, 64, and 128) was changed to test the effect of the number of samples in each subdivision on the accuracy of the matching algorithm. Results showed that a number of samples are effective in the matching algorithm, which had previously not been investigated.[4]Our final Results is the compare of classification output of both classifiers in tens of classification efficiency. We find that ANN vvith dmey wavelet give highest classification efficiency vvith bath training and testing data set.Db4 based ANN also gave good classification results for training data set but the performance ofDb4 based ANN is poor for testing data as compared to Demy based ANN.Haar based Ann and KNN gives very poor classification result for both training and 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 than the overall classification efficiency compared to all wavelet based KNN. [5] Extended the concept of Chinese information retrieval, it is easy to index a Chinese text document for retrieval, as 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 a text file by using Chinese optical character recognition (OCR) technology, and then index the document by using information retrieval algorithm. [6] The methods evaluate in our study are representative techniques. And our evaluations show that to obtain high accuracy it is not necessary to use a large nor computationally intensive image descriptor. We have also presented results per transformation to gain further insight into the strengths and weaknesses of the methods.

3. PROPOSED WORK

Identification of duplicate images consists of five steps. a) Image Enhancement b)Speeded Up Robust Features(SURF) c)Feature matching using SURF d)Locality Sensitive Hashing e) Optimizing by Re-indexing

3.1 Enhancement of Search image

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

3.2 Speeded Up Robust Features (SURF)

Feature detection is the key step of the multi-view image registration process. Here, any salient and distinctive objects or features like closed boundary regions, edges, contours, line intersections or comers are detected. Speeded up Robust Feature (SURF) is a scale and rotation invariant interest point detector and descriptor. It approximates or even outperforms earlier proposed methods with respect to repeatability, distinctiveness, and robustness, and can be computed and compared much faster [12]. SURF feature detector works based on the second order Hessian matrix. It utilizes the integral images and box filter to speed up the computation. The integral image is a cumulative image in which a single point p(x,y) corresponds to the summed values of all points above and to the left of p(x,y) [13].

Hessian matrix H(x,) for a point X at scale is defined as follows

H(x,) $_{=[] (1)}$Where, L $_{xx}$(x,) is the convolution of the Gaussian second order derivative with the image at point X.

3.3 Feature Matching using SURF

This step establishes the correspondence between the features detected in the first step. Feature matching using SURF is based on the distance matrix. Generated 64 dimensional feature vectors for detected interest points in both the images are used for feature matching. First calculate the SSD distance matrix of the both the images. The interest point will be matched when the second nearest distance is larger some ration than first nearest distance.

3.4 Locality Sensitive Hashing

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 dramatic improvement in effective seek time — cutting it to approximately 1ms per seek. Gioniset al. 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.

3.5 Algorithm for Similarity calculations

Procedure calculating Resemblance

Input: Features of first web image.

Output: similarity of features of first web image and Near duplicate images

1.Features of first web imagew1 like wf1,wf2,wf3,……,wfn

Features of near duplicate images ni like nd11,nd12,nd13,……,nd1n,

nd21,nd22,nd23,……..nd2n,

nd31,nd32,nd33,……..nd3n,

ndi1,ndi2,ndi3,……..ndin

2.set rem[i]=0;

3.for all images I=1,……,n do

for all images F=1,……,K do

If(wfF==ndiF) then

Increment the rem[i]

Increment the Features

End

Increment the images

End

4.If (rem[i]==k) then

Goto Re-indexing algorithm.

3.6 Algorithm for Re-indexing

Input:features of Near duplicate image.

Output:Re-indexing of near duplicate image

1.for all images i=1,.....,n

If(rem[i]==rem[i+1])

Pos[i]=rem[i];

End

2.Set J=i+1;

3.If(rem[j]<rem[j+1])

Pos[j]=rem[j+1];

Pos[j+1]=rem[j];

else

Pos[j]=rem[j];

Pos[j+1]=rem[j+1];

End

4.Display of pos[i];

5.stop.

3.7 Overall Proposed System

The proposed system is used to identify and Re-indexing the near duplicate images and similar duplicate images corresponding to the user search image;

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

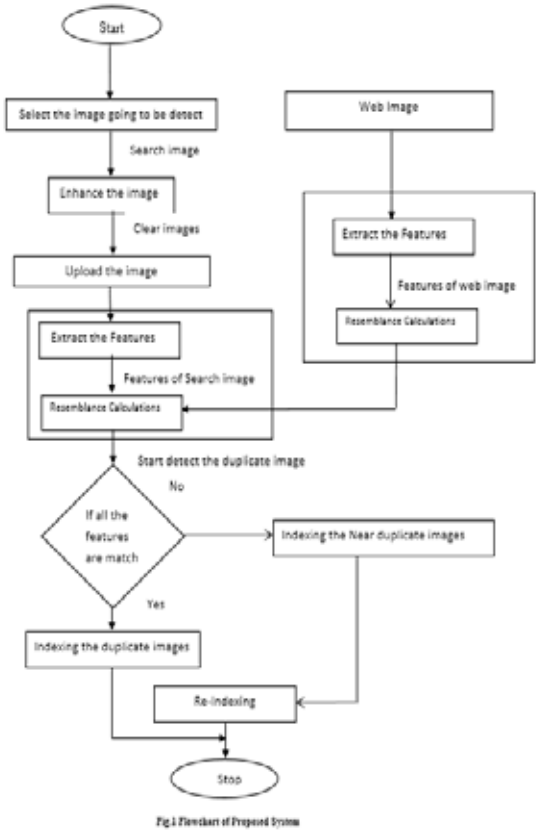

Fig.1 Flowchart of Proposed System

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 by Re-indexing. 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. 2.Uploading the users search image

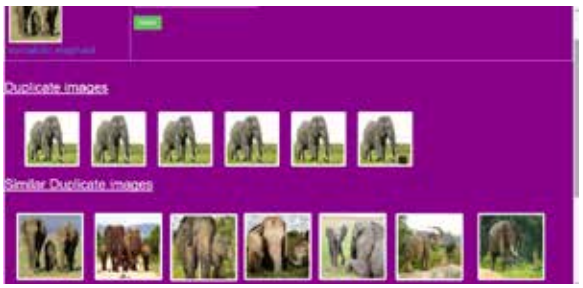

Fig. 3 Indexing the duplicate images and similar duplicate images

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 Re-indexing of near-duplicate images. Initially, the search image is given by the user and features are extracted and similarity is calculated between the search image and also web image. 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 search image and then extract the feature. After features are extracted Similarity is measured and then indexing the near duplicate images and also optimize the result by Re-indexing the near duplicates. This results in indexing of images. We conclude that our Re-indexing approach is highly effective for collections of up to a few hundred thousand images.

REFERENCES

[1] Changjing Shang and Dave Barnes ,"Support Vector Machine-Based Classification of Rock Texture Images Aided by Efficient Feature Selection", WCCI 2012 IEEE World Congress on Computational Intelligence June, 10-15, 2012 - Brisbane, Australia

[2] R.I.Minu and Dr.K.K.Thyagharajan,"Automatic Image Classification Using SVM Classifier",CiiT International Journal of Data Mining and Knowledge Engineering,Vol 3,No 9,July 2011

[3] Bassem Sheta1, Mohamed Elhabiby1, 2, and Naser El-Sheimy,"ASSESSMENTS OF DIFFERENT SPEEDED UP ROBUST FEATURES (SURF) ALGORITHM RESOLUTION FOR POSE ESTIMATION OF UAV", International Journal of Computer Science & Engineering Survey (IJCSES) Vol.3, No.5, October 2012

[4] DharmendraPatidar,Nitin jain, Ashish Parikh, Performance Analysis ofArtificial Neural Network and K NearestNeighbors Image Classification TechniqueswithWaveletfeatures", 2014 IEEEInternational Conference on Computer Communication andSystems(ICCCS '14),Feb20-21,2014, Chennai,ThIDIA

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

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

[7] Bradski GR. Computer vision face tracking as a component of a perceptual user interface[C]. IEEE Workshop Application of Computer Vision, 1998: 214-219.

[8] Tai J, Tsang S, Lin C et.al. Real-time image tracking for automatic traffic monitoring and enforcement application[J]. Image and Vision Computing, 2004, 22 (6): 485-501.

[9] Desouza GN, Kak AC. Vision for mobile robot navigation: A survey[J]. IEEE Transaction on Pattern Analysis and Machine Intelligence, 2002, 24 (2): 237-267.

[10] 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

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

[12] Herbert Bay, Andreas Ess , Tinne Tuytelaars ,and Luc Van Gool , " Speeded-Up Robust Features (SURF) ", Elsevier , 2008.

[13]. R. Karthik, A. Annis Fathima, V. Vaidehi, Panoramic View Creation using Invariant Moments and SURF Features ", International Conference on Recent Trends in Information Technology (ICRTIT) , IEEE ,2013.