Duplication Image And Video Detection Using Min-Hash Method

Hariharan.P[1], Dr V.Saravanan[2]
[1] Dept of Information Technology
[2] HOD & Professor, Dept of Information Technology
[1, 2] Hindusthan college of Arts and science, Coimbatore

Abstract- *Automated robust methods for duplicate detection of images & videos are getting more attention recently due to the exponential growth of multimedia content on the web. The large quantity of multimedia data makes it infeasible to monitor them manually. In addition, copyright violations and data piracy are significant issues in many areas including digital rights management and in the entertainment industry.*

My main aim is to propose a system that can detect duplicate images in very large image databases. Specifically focus on the scalability issue. The proposed approach results in a very compact image signature that is robust to many image processing operations, can be indexed to efficiently search large databases (show results on a 10 million image database), and is quite effective (about 82% precision).The method can also be extended for similar image detection and "region of interest" duplicate detection. In many practical scenarios, the duplicates are not identical replicas of the images in the database, but are digitally processed versions of the original images in the database. In these cases, standard hashing methods will not work.

I. INTRODUCTION

1.1 PROBLEM DEFINITION

The definition of a near duplicate image varies depending on what photometric and geometric variations are deemed acceptable. The application ranges from exact duplicate detection where no changes are allowed to a more general definition that requires the images to be of the same scene, but with possibly different viewpoints and illumination.

In this project, build on a Min-Hash method that addresses (through a similarity threshold parameter) a whole range of near duplicate images: from images that appear, to a human observer, to be identical or very similar to images of the same scene or object. Detection of near duplicate images in large databases imposes two challenging constraints on the methods used. Firstly, for each image only a small amount of data (a fingerprint) can be stored; secondly, queries must be very cheap to evaluate. Ideally, enumerating all the duplicates of an image should have complexity close to linear in the number of duplicates was returned.

.

II. EXISTING SYSTEM

Existing system does not consider about the images and their duplicates. It considers only about the duplicate copy of the text files and other regarding files while just comparing by two files as per the input is given to the software or regarding project and comparing that files it will display the duplicate survey existing techniques, such a scheme would be highly resistant to occlusions and cropping, both of which can destroy a significant fraction of the features.

Image could generate thousands of local features, and a single query would require the system to search for each of these features in a database containing millions or billions of features. Since features would not generate exact matches, each of the individual searches would become a similarity query in a very high dimensional feature space. Consequently, such approaches have previously been dismissed as computationally impractical.

The closest work is on NDID based on min-Hash. The project extends the method and directly compares the results. demonstrate near-duplicate detection and sub-image retrieval by using sparse features, taken from each image, coupled with a disk-based Locality Sensitive Hashing (LSH) for fast approximate search on the individual feature descriptors. They demonstrate the efficacy of their method on a synthetic database of "corrupted" images but show the system only scaling to handle 18K images with query times many times slower than the min-Hash method. It uses a parts-based representation of each scene by building Attributed Relational Graphs (ARG) between interest points.

They then compare the similarity of two images by using Stochastic Attributed Relational Graph Matching, to give impressive matching results. Unfortunately, they only demonstrate their method on a few hundred images and don't discuss any way to scale their system to larger data sets of images.

Relevant work has been published on near duplicate shot detection (NDSD). These methods typically use strong temporal constraints than are not available in NDID.

For example, use an edit distance. Represents each key frame by a set of 20D spatial-temporal descriptors computed about Harris interest points (requiring storing a large amount of data per key frame – possibly hundreds of Harris points and their descriptors).

2.1.1 Drawbacks

- large sets of image features is not detected
- Near-duplicate detection within the images returned by existing text-based image search engines.
- Digital watermarking techniques exist, these schemes are very difficult to design and there is an inherent trade-off between the robustness of the watermark and the amount of degradation induced in the image.
- Clustering tasks (such as finding all groups of near duplicated images in the database) the bit string representation is less suitable.
- Global image descriptor, which limits the approach to no geometric / viewpoint invariance.
- They demonstrate the efficacy of their method on a synthetic database of "corrupted" images but show the system only scaling to handle 18K images with query times many times slower than the min-Hash methods.

III. PROPOSED SYSTEM

This system considers images and their pixels and Meta data of an image. It uses Hausdorff distance using hash function to parse image and find the duplicate copy of the images and survey. Here there is no requirement of an input file.

Her it requires where is located ex folders or drivers it will display all the duplicate copies of n number images and m number duplicate copies by comparing both the pixels and metadata and the binary values and the exact similarity and image quality by using the hash function for the Hausdorff distance.

Define Image Near-Duplicate (IND) as a pair of images in which one is close to the exact duplicate of the other, but differs slightly due to variations of capturing conditions (camera, camera parameter, view angle, etc), acquisition times, rendering conditions, or editing operations. Detecting IND can be used in a variety of applications, for example, linking multimedia content, identifying copyright infringement, managing photo albums, etc. image similarity measures for fast indexing via locality sensitive hashing.

The similarity measures are applied and evaluated in the context of near duplicate image detection. The proposed method uses a visual vocabulary of vector quantized local feature descriptors and for retrieval exploits enhanced min-Hash techniques. Standard min-Hash uses an approximate set intersection between documents descriptors was used as a similarity measure.

The choice of an image representation and a distance measure affects both the amount of data stored per image and the time complexity of the database search. The amount of stored data ranges from a constant (small) amount of data per image to storing large sets of image features, whose size often far exceeds the size of the images themselves. When searching the database for relevant images, algorithms of different time complexity are used, the most naive approach being computing the similarity between every image pair in the database.

3.1 ADVANTAGES

Near duplicate image detection (NDID), specifically in the min-Hash algorithm.

- The min-Hash method stores only a large constant amount of data per image. Image representation and the similarity measure is that it enables very efficient retrieval.
- Efficiently compute the Hausdorff Distance between all relative positions of a model and an image. The similarity measures do not require extra computational effort compared to the original measure.

VI. DESCRIPTION OF MODELS

4.1 IMAGE FEATURE EXTRACTION

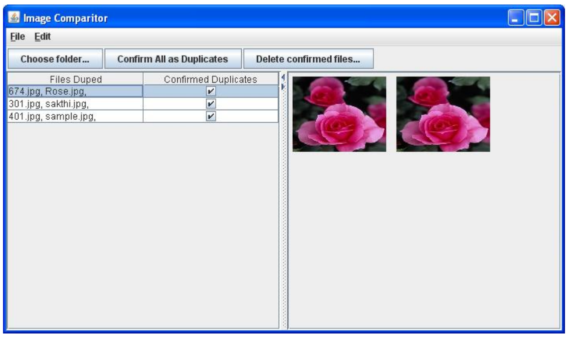
- **Feature Extraction** - method of capturing visual content of Feature Extraction - method of capturing visual content of images. The features should carry enough information about the image and should not require any domain-specific knowledge for their extraction. They should be easy to compute in order for the approach to be feasible for a large image collection. Color feature is one of the most widely used features in Image Retrieval. Color Histogram is the most used in color feature representation. Closely related to human visual perception RGB color model.

- **Textures Histogram** can be rough or smooth, vertical or horizontal. spatial arrangement of color or intensities
- **Shape Feature** Extraction is image part separating.
- **Curve Feature** Extraction is angel based extraction.

4.2 META DATA

Automatically extracts preservation-related metadata from digital files. EXIF reader techniques used to extract the meta data. Extracts high-quality metadata. The term **metadata** is an ambiguous term which is used for two fundamentally different concepts (types). Although the expression "data about data" is often used, it does not apply to both in the same way. Structural metadata, the design and specification of data structures, cannot be about data, because at design time the application contains no data. In this case the correct description would be "data about the containers of data". Descriptive metadata, on the other hand, is about individual instances of application data, the data content. In this case, a useful description (resulting in a disambiguating neologism) would be "data about data content" or "content about content" thus meta content. Descriptive, Guide and the National Information Standards Organization concept of administrative metadata are all subtypes of Meta content.

Metadata (metacontent) is traditionally found in the card catalogs of libraries. As information has become increasingly digital, metadata is also used to describe digital data using metadata standards specific to a particular discipline. By describing the contents and context of data files, the quality of the original data/files is greatly increased. For example, a webpage may include metadata specifying what language it's written in, what tools were used to create it, and where to go for more on the subject, allowing browsers to automatically improve the experience of users.



Files Duped	Confirmed Duplicates
674.jpg, Rose.jpg,	☑
301.jpg, sakthi.jpg,	☑
401.jpg, sample.jpg,	☑

Figure 1.finding duplicate image

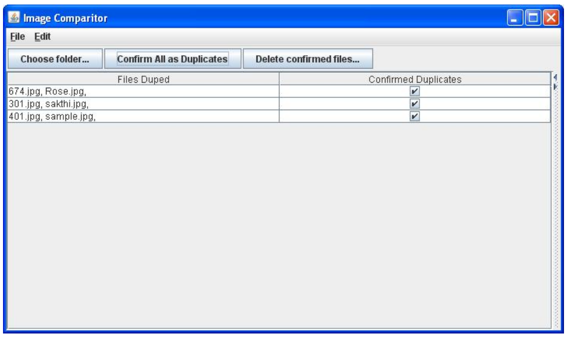
4.3 PARSING AND COMPARISON

Natural images consist of an overwhelming number of visual patterns generated by very diverse stochastic processes in nature. The objective of image understanding is to parse an input image into its constituent patterns. Define image parsing to be the task of decomposing an image **I** into its constituent visual patterns.

A typical example where a football scene is first divided into three parts at a coarse level: a person in the foreground, a sports field, and the spectators. These three parts are further decomposed into nine visual patterns in the second level: a face, three texture regions, some text, a point process (the band on the field), a curve process (the markings on the field), a color region, and a region for nearby people.

4.4 DISPLAYS AND DELETING THE DUPLICATES

- Near duplicate image retrieval, especially on large data sets.Present an extensive comparison of the original min-H.Introduced a framework for image parsing by defining generative models for the processes that create images including specific objects and generic regions such as shading and texture. Here using hash map to find the duplicate images.



Files Duped	Confirmed Duplicates
674.jpg, Rose.jpg,	☑
301.jpg, sakthi.jpg,	☑
401.jpg, sample.jpg,	☑

Figure 2. Files conformed to Delete

V. SYSTEM IMPLEMENTATION

The modules in this project has been tested thoroughly and found to be accurate which can meet the needs on the user. The tested modules are finally combined together into a complete one. The user acceptance testing is found to be little difficult and in this project user satisfaction is obtained.

A principal activity of the development phase is coding and testing until the user requirement specification fulfilled by the each modules and component of the overall system. Other important activities include implementation planning, equipment acquisition and system testing. The development phase concludes with a development phase report and user review.

A software application in generally implemented after navigating the complete life cycle method of a project. Various life cycle processes such as requirement analysis, design phase, verification, testing and finally followed by the implementation phase results in a successful project management. The software application which is basically a web based application has been successfully implemented after passing various life cycle processes mentioned above.

As the software is to be implemented in a high standard industrial sector, various factors such as application environment, user management, security, reliability and finally performance are taken as key factors throughout the design phase. These factors are analyzed step by step and the positive as well as negative outcomes are noted down before the final implementation.

The application's validations are made, taken into account of the entry levels available in various modules. Possible restrictions like number formatting, date formatting and confirmations for both save and update options ensures the correct data to be fed into the database. Thus all the aspects are charted out and the complete project study is practically implemented successfully for the end users.

Two documents are near duplicate if the similarity is higher than a given threshold r. the goal is to retrieve all documents in the database that are similar to a query document. This section reviews an efficient randomized hashing based procedure that retrieves near duplicate documents in time proportional to the number of near duplicate documents. The outline of the algorithm is as follows: First a list of min-Hashes is extracted from each document.

A min-Hash is a single number having the property that two sets A1 and A2 have the same value of min-Hash with probability equal to their similarity sims(A1,A2). For efficient retrieval the min-Hashes are grouped into n-tuples called sketches.

Identical sketches are then efficiently found using a hash table. Documents with at least h identical sketches (sketch hits) are considered as possible near duplicate candidates and their similarity is then estimated using all available min-Hashes.

VI. CONCLUSION

Two novel similarity measures whose retrieval performance is approaching the well established tf-idf weighting scheme for image / particular object retrieval. Show that pairs of images with high values of similarity can be efficiently (in time proportional to the number of retrieved images) retrieved using the min-Hash algorithm.

Have shown experimental evidence that the idf word weighting improves both the search efficiency and the quality of the results. The weighted histogram intersection is the best similarity measure (out of the three examined) in both retrieval quality and search efficiency. Promising results on the retrieval database encourage the use of the hashing scheme beyond near duplicate detection, for example in clustering of large database of images.

VII. FUTURE ENHANCEMENT

Further, make system practical through careful data placement and batched disk accesses to minimize random seeks. Show experimentally that the system has **near perfect** accuracy (99.85% recall and 100% precision) on a standard test set. For future work, plan to further optimize the data structures to gain additional query performance and further improve accuracy.

REFERENCES

[1] G. Padmavathi, D. Shanmugapriya and M. Kalaivani, *"A Study on Vehicle Detection and Tracking Using Wireless Sensor Networks,"* *Wireless Sensor Network*, Vol. 2 No. 2, 2010, pp. 173-185. doi: 10.4236/wsn.2010.22023

[2] Y.Zhang, *A multilayer IP security protocol for TCP performance enhancement in wireless networks.* IEEE Journal on Selected Areas in Communications, Vol. 22, n. 4, pp. 767-776, May 2004. NS-2 Network Simulator (Vers. 2.27),URL: http://www.isi.edu/nsnam/ns/nsbuild.html

[3] M. Luglio, A. Saitto, “Security of Satellite Networks”, chapter in H. Bidgoli (Ed), “The Handbook of Information Security”, John Wiley & Sons, Inc., 2006, Hoboken, N.J., Vol. 1, pp. 754-771.

[4] M. P. Howarth, S. Iyengar, Z. Sun and H. Cruickshank, “Dynamics of key management in secure satellite multicast”, IEEE Journal on Selected Areas in Communications, Vol. 22, n. 2, pp. 308-318.

[5] C. Partridge, and T. Shepard, *TCP Performance over Satellite Links*.IEEE Network, vol. 11, n. 5, 1997, pp. 44-49.

[6] W. Stevens, *TCP/IP illustrated, Volume 1*. Addison Wesley, 1994.

[7] ETSI TS 102 292, Broadband Satellite Multimedia (BSM); Functional Architecture

[8] Caini, C., et al.: PEPsal: A Performance Enhancing Proxy for TCP Satellite Connections. IEEE A&E Systems Magazine (August 2007)

[9] I-PEP specifications, Issue 1a. Satlabs group recommendations (October 2005), http://www.satlabs.org

[10] ETSI TS 102 463: Broadband Satellite Multimedia (BSM); Interworking with IntServQoS

[11] ETSI TS 102 464: Broadband Satellite Multimedia (BSM); Interworking with DiffServQoS

[12] Obanaik, V.: Secure performance enhancing proxy: To ensure end-to-end security and enhance TCP performance over IPv6 wireless networks. Elsevier Computer Networks 50, 2225–2238 (2006)

[13] Bellovin, S.: Probable plaintext cryptanalysis of the IPSecurity protocols. In: Proceedings of the Symposium on Network and Distributed System Security (February1997)

[14] M. Annoni*et al.*, “Interworking between multi-layer IPSEC and secure multicast services over GEO satellites,” presented at the COST-272 Symp., Thessaloniki, Greece, June, 20–21 2002. Doc. TD-02-016-P.

[15] J. Arrko*et al.*, “MIKEY: Multimedia Internet Keying,” IETF Internet Draft, work-in-progress, draft-ietf-msec-mikey-06.txt, Feb. 2003 , expires Aug. 2003.

[16] N. Assaf*et al.*, “Interworking between IP security and performance enhancing proxies for mobile networks,” *IEEE Commun. Mag.*, vol. 40, pp. 138–144, May 2000.