International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
NCICCT' 14 Conference Proceedings

Efficient Content Based Image Search and Retrieval (CBIR) Using Sift Based On Multi Sort Indexing

A.Rajasri,
Department of CSE,
University College of Engineering (BIT Campus),
Anna University Chennai,
Trichirappalli-650624
shrirajasri17@gmail.com

J. B. Shriram,
Department of CSE,
University College of Engineering (BIT Campus),
Anna University Chennai,
Trichirappalli-650624
shriram.jb@gmail.com

***Abstract*- A common approach to content-based image search and retrieval has been widely used to describe the process of retrieving desired image from a large collection on the basis features such as color, texture and shape. For efficient content search and retrieval many methods have been proposed but accuracy and search time has limiting factors in those methods such as hashing and tree based methods etc.. To overcome this drawback, the proposed system has been designed which includes extraction SIFT features on multi sort indexing. By analyzing the high dimensional value cardinalities and inherent character of descriptor quantization and normalization have been used in the extraction process. Since dimension with unique value cardinality have more discriminative power a multiple sort algorithms is used. Multi sort indexing algorithm reduce the dimension based on cardinality so that two similar images lie within a close constant range from the experiments conducted with INRIA holidays datasets.**

Keywords: SIFT, CBIR, multisort Indexing

I. Introduction

The accessibility of images over the internet become obvious to the address of the challenge of the content based searching, in turn to find visually similar content. Content-based image retrieval (CBIR), a technique for retrieving images on the basis of automatically-derived features such as color, texture and shape. CBIR operates on a totally different principle from keyword indexing. Primitive features characterizing image content, such as color, textures, and shape, are computed for both stored and query images, and used to identify the stored images most closely matching the query.

Content-based means contents of the image rather than the metadata such as tags, keywords, and/or descriptions associated with the image. The term content in this context might refer to textures, shapes, colors, or any other information that can be derived from the image. The process of digitization does not in itself make image collections easier to manage. Some form of classification and indexing is still essential – the only difference being that much of the required information can now potentially be derived automatically from the images themselves. It can be grouped under three captions – image compression, query specification and metadata description. CBIR is enviable because most web based image search engines depend purely on metadata and this produces a lot of garbage in the results.

Since keywords manually entered by humans for images in a large database may be inefficient, expensive and sometimes do not capture every keyword that describes the image [1]. Thus a system that can filter images based on their content would provide better indexing and return more accurate results. Access to a desired image from a repository might thus involve a search for images depicting specific types of object or scene, evoking a particular mood, or simply containing a specific texture or pattern. User wants to search for, say, many rose images. He/she submits an existing rose picture as query. He/she submits his own sketch of rose as query. The system will extract image features for this query. It will compare these features with that of other images in a database. Relevant results will be displayed to the user.

II. Related work

In this section we present literatures of several content based image retrieval techniques and similarity searching mechanisms.

Renato O. Stehling [10] et al.Proposed BIC (Border/Interior pixel Classification) a compact and efficient CBIR approach is provided which is suitable for broad image realm. It has three main components: (1) a simple and powerful image analysis algorithm that classifies image pixels as either border or interior, (2) a new logarithmic distance (dLog) for comparing histograms, and (3) a compact representation for the visual features extracted from images. After the image pixels are classified, color histogram is computed one for border pixels, and another color histogram for interior pixels. After the quantization step, image pixels are classified as border or interior pixels. A pixel is classified as border if it is at the border of the image itself or if at least one of its 4-neighbors (top, bottom, left and right) has a different quantized color. Limitations of Pixel classification sometimes they obtained regions are part of a real object, i.e., a user would likely identify by looking at the image. The criterion of homogeneous visual properties usually leads to a super segmentation of the image.

Van De Sande [5] et al. Presents Image category recognition, which is important to access visual information on the level of objects and scene types. In both image retrieval and video retrieval use machine learning based on image descriptions to distinguish object and scene categories. Changes in the illumination of a scene can greatly affect the performance of object recognition if the descriptors used are not robust to these changes. The invariance properties and the distinctiveness of color descriptors are studied in a structured way. The taxonomy is resultant of the diagonal model of illumination change. Then, the distinctiveness of color descriptors is analyzed experimentally using two benchmarks from the image domain and the video domain. Limitations of color Descriptors. The moments and histograms are not very distinct when compared to SIFT – based descriptors: it contains too little relevant information to be competitive with SIFT. The performance gets affected, when no prior knowledge about the dataset and object and scene categories is available.

K. Mikolajczyk [6]et al. propose a novel approach for detecting interest points invariant to scale and affine transformations. Scale and affine invariant detectors are based on the following results: Interest points extracted with the Harris detector can be adapted to affine transformations. The characteristic scale of a local structure is indicated by a local extremum over scale of normalized derivatives. The affine shape of a point neighborhood is estimated based on the second moment matrix. Limitations of the detector are the invariance to geometric and photometric affine transformations removes some of the information. The unreasonably high number of pints increases the probability of mismatches and the complexity.

L. Pauleve [7] et al. The hashing methods have proven to b e suitable for approximate similarity search, since they support efficient indexing and data compression. The basic idea of the hashing methods is (a) to encode the distances between the data into the form of compressed sequences of bits by using hash functions, and (b) to store the encoding distances into buckets, in order to ensure that the probability of collision is much higher for data that are close to each other than those that are far apart. Then, they approximate exact similarity measures by comparing has codes, using a hamming distance on binary codes or other measures.. Diverse strategies are followed during the preprocessing for the generation of the binary codes.

The existing hashing methods can be broadly categorized as data-independent hashing methods are Local sensitive Hashing (LSH), spherical Hashing, Multi-Probe LSH, Posteriori Multi-Probe LSH and Shift-Invariant Kernels Hashing. Limitations of Hashing are failing to achieve accuracy when the hashing functions are drawn independently from the data. Requires additional Memory usage. More Processing time is required for long binary code lengths.

J.Song [4]et al. Instead of indexing the data into the original high-dimensional space, dimensionality reduction methods aim at mapping the data into a lower-dimensional subspace. The main idea is to make such a transformation without losing much information and build an index on the subspace. Global dimensionality reduction methods map the whole dataset into a much-lower dimensional subspace. For example, the Locally Linear Embedding method projects the data to a low-dimensional space, while preserving local geometric properties.

Dimensionality reduction methods can be used either for exact or approximate similarity search. In the first case, the similarity search is performed only into the transformed subspace. In the second case, first the similarity search is performed into the transformed space, where lower bounds on the distances are used for filtering, then a resulting set of candidates is returned, and finally the candidates are refined in the original space with exact search. Limitations of Dimensionality reduction methods is some of the limitations of the existing dimensionality reduction methods are the scatter gets maximized by the between-class scatter that is useful for classification, and also by the within-class scatters that, for classification purposes. Much of the variation from one image to the next is due to illumination changes. Consequently, the points in the projected space will not be well clustered, and worse, the classes may be smeared together. Preprocessing cost of the transformation is high.

The main strategy for all tree-based indexing methods is to prune tree branches on the established bounding distances in order to reduce the node accesses. Tree-based indexing organizes the terms into a single tree. Each path into the tree represents

ISSN: 2278-0181
NCICCT' 14 Conference Proceedings

general properties of the indexed terms, similar to classification trees or decision trees. The basic tree-based indexing method is discrimination tree indexing. The tree reflects precisely the structure of terms. A more complex tree based method is abstraction tree indexing. The nodes are tagged with lists of terms, in a manner that reveal the substitution of variables from one term to another: the domain of variable substitutions in a node is the co-domain of the substitutions in a sub node. A relatively tree-based method was substitution tree indexing. Substitution tree indexing reveals retrieval and deletion times faster than other tree-based indexing methods. Limitations of Tree based Indexing methods in high dimensional spaces tree based indexing methods become inefficient. Requires additional time for insertion and deletion. Sometimes requires more substitution which takes additional processing time.

To overcome this drawback we designed efficient content based image search and retrieval using scale invariant feature transform based on multisort indexing.

III. Proposed system

The main objective of our proposed system is providing efficient content based image search and retrieval has become a difficult task due to the limiting factors such as searching time and accuracy. To overcome this drawback proposed system has been designed with scale invariant feature extraction and Multisort indexing. SIFT is a method for detecting and extracting local feature descriptors. Multisort Indexing rearrange the dimensions of the descriptor vectors according to the cardinalities. By these results are combined and it produced from the above-mentioned approaches efficient search time and accurate retrieval is achieved.

A. *Sift Feature Extraction*

This method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of a scene or object. The features are invariant to image scale and image rotation, are exposed to provide vital matching across a significant choice of affine distortion, and change in illumination, change in 3D viewpoint and addition of noise. The features are extremely distinctive, for that, a single feature can be properly matched with high possibility against a large database of features from many images. It also describes an approach to using these features for object recognition. The identification takings by similar individual features to a database of features from known objects using a fast nearest-neighbor algorithm, followed by a Hough transform to identify clusters belonging to a single object, and at last performing verification through least-squares solution for consistent cause parameters. Scale Invariant Feature Transform is a method for detecting and extracting local feature descriptors which are invariant to changes in image noise, illumination, scaling, rotation, and small changes in viewpoint.

Stages detection for SIFT features: Scale-space extreme detection, Key point localization, Orientation assignment, Generation of key point descriptors.

B. *Scale space extreme Detection*

An important issue is to determine the frequency of sampling in the image and scale domains that are needed to reliably detect the extreme. Inadequately, it turns out that there is no minimum spacing of samples that will detect all extreme, as the extreme can be impulsively close together. This can be seen by allow for a white circle on a black background, which would have a single scale space maximum where the circular positive central region of the difference-of-Gaussian function matches the size and location of the circle. Interest points for SIFT features correspond to local extreme of difference-of-Gaussian filters at different scales, Given a Gaussian-blurred image.

$$\mathbf{L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)}$$

Where

$$\mathbf{G(x, y, \sigma) = 1/(2\pi\sigma^2)exp^{-(x2+y2)/\sigma2}}$$

Is a variable scale Gaussian, and then the image is convolved with a difference-of-Gaussian filter.

The first step toward the detection of interest points is the convolution of the image with Gaussian filters at different Scales, in which the difference-of-Gaussian images are obtained from the difference of adjacent blurred images. The convolved images are grouped by octave (an octave corresponds to doubling-up the value of σ), and the value of k is selected so that we obtain a fixed number of blurred images per octave. This also makes certain that we obtain the same number of difference-of-Gaussian images per octave.

C. *Key point Localization*

Interest pints (called key points in the SIFT framework) are identified as local maxima or minima of the DoG images across scales.

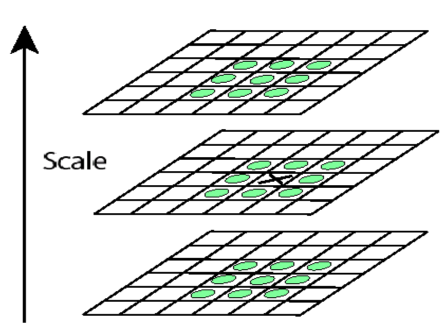

Fig: Local Extreme Detection

Each pixel in the DoG images is compared to its 8 neighbors by the same scale, plus the 9 corresponding neighbors by the side of neighboring scales. If the pixel is a local minimum or maximum, it is preferred as a candidate key point.

For each candidate key point:

1. Interpolation of nearby data is used to accurately determine its position.
2. Key points with low contrast are removed
3. Responses along edges are eliminated.

D. Orientation Assignment

A gradient orientation histogram is computed to determine the key point orientation, in the neighborhood of the key point (using the Gaussian image at the closest scale to the key point's scale). Peaks in the histogram correspond to dominant orientations. key point was created for the direction corresponding to the histogram maximum, and any other direction within 80% of the maximum value. All the properties of the key point are calculated corresponding to the key point orientation, which give invariance to rotation.

E. Feature Descriptor

Once a key point orientation has been selected, the feature descriptors are computed as a set of orientation histograms based on 4*4 pixel neighborhoods. The orientation histograms are virtual to the key point orientation, in which the orientation information comes from the Gaussian image which is closest in scale to the key point's scale. The contribution of each pixel is weighted by the gradient magnitude, and by a Gaussian with **σ 1:5** times the scale of the key point.

F. Feature Matching

Find nearest neighbor in a database of SIFT features from training images. For robustness, use ratio of nearest neighbor to ratio of second nearest neighbor. The matched image will be the neighbor with minimum Euclidean distance.

IV. Multisort Indexing

Examines the value cardinalities of the SIFT descriptors' dimensions. Reorder the dimensions of the descriptor vectors according to their value cardinalities. Supports dynamic indexing and storing of the new image content. Multi-Sort Indexing is an indexing method used to reorder the storage positions of images' descriptors according to value cardinalities of their dimensions, by performing a multiple sort algorithm, in order to increase the probability of having two similar images in storage positions that do not differ more than a specific global constant range, denoted by a parameter.

Steps in Multi-Sort Indexing are Preprocessing, Insertion, query Processing and Deletion.

A. Preprocessing

For indexing and storing of a pre-existing image dataset in the form of high dimensional descriptor vectors. Initially cardinality values are computed from the descriptor values for each dimension where three cases such as Integer values, Normalized Real values and real values are identified based on the descriptor value type. Since dimensions with high value cardinalities correspond to dimensions with high discriminative power, the value cardinalities are sorted in a descending order. Finally priority index is created based on the cardinality values.

B. Insertion

Provides real-time indexing and storing of a new (non-existing) image in the form of descriptor vector. Allocate the storage position for the new image D-dimensional descriptor vectors. Compare the dimensional vectors of new image (v_i) and existing image(y_i).

C. Query Processing

Query processing provides searching of the top-similar images to a posed image query. Query image is inserted into the datasets. Allocate the position of the query image. Compute the distance between the query image and the database image using a distance measure d. The calculated distances are inserted into a minimum heap structure Finally remove and retrieve the top-k results from the heap constituting the result set R.

D. Deletion

Deletion of the image is performed by retrieving the physical memory address and clearing its content. Removes already indexed images.

V. Conclusion

The Proposed System provides the estimated method for efficient Content-Based Image search and Retrieval by using Scale Invariant feature descriptors and Multi-Sort Indexing. The content of the image is analyzed according to the value cardinalities that appear on the dimensions of the descriptive vectors. Able to performing the content-based search and retrieval in low time and handles dynamic operations of insertion and deletions. Experimental results show that the efficiency and effectiveness of the proposed method gets improved in terms of search time and retrieval accuracy. Further to improve the retrieval accuracy and efficiency, the image search and retrieval can be performed by using various types of feature descriptors in a reduced dimensional subspace.

VI. References

[1] Eleftherios Tiakas, Dimitrios Rafailidis, Anastasios Dimou, And Petros Daras, Msidx : Multi-sort Indexing For Efficient Content-based Image Search And Retrieval IEEE Transactions On Multimedia, Vol. 15, No. 6, October 2013.

[2] D. Lowe, "Distinctive image features from scale-invariant keypoints ,"Int. J. Comput. Vision, vol. 60, pp. 91–110, 2004.

[3] Greg Pass, Ramin Zabih,Justin Miller, "Comparing Images Using Color Coherence Vectors"

[4] J. Song, Y. Yang, Z. Huang, H. T. Shen, and R. Hong, "Multiple feature hashing for real-time large scale near-duplicate video retrieval", in Proc. of ACM Multimedia, pp. 423-432, 2011.

[5] K. E. A. Van De Sande, T. Gevers, and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 9, pp. 1582–1596, 2010.

[6] K. Mikolajczyk and C. Schmid, "Scale and Affine invariant interest point detectors", Int. Journal in Computer Vision, vol. 60, no. 1, pp. 63-86,2004

[7] L. Pauleve, H. Jegou, and L. Amsaleg, "Locality sensitive hashing: A comparison of hash function types and querying mechanisms," Pattern Recognit. Lett., vol. 31, no. 11, pp. 1348–1358, 2010.

[8] O. Chum, J. Philbin, and A. Zisserman, "Near duplicate image detection: Min-hash and tf-idf weighting," in Proc. British Machine Vision Conf., 2008.

[9] Q. Lv, W. Josephson, Z. Wang, M. Charikar, and K. Li, "Multi-probe LSH: Efficient indexing for high-dimensional similarity search," in Proc. Int. Conf. Very Large Data Bases (VLDB), 2007, pp. 950–961.

[10] R. O. Stehling, M. A. Nascimento, and A. X. Falcao, "A compact and efficient image retrieval approach based on border/interior pixel classification,"in Proc. CIKM, 2002.

[11] T. Chiueh, "Content-based image indexing," in Proc. Int. Conf. Very Large Databases (VLDB), 1994, pp. 582–593.

[12] Y. Gong, S. Lazebnik, A. Gordo, and F. Ferronnin,Iterative quantization: A procrustean approach to learning binary codes for large-image retrieval," IEEE Trans. Pattern Anal. Mach. Intell.

[13 Yixin Chen, James Z. Wang, Robert Krovetz, "Content-Based Image Retrieval by Clustering",MIR'03, November 7, 2003, Berkeley, California, USA.