

AN EFFICIENT/ENHANCED CONTENT BASED IMAGE RETRIEVAL FOR A COMPUTATIONAL ENGINE

¹Shriram, K.V., ²P.L.K. Priyadarsini, ³Kaushik Velusamy and ¹A. Balachandran

¹Centre for Research and Development, PRIST University, India

²Department of CSE, Temporary Faculty, National Institute of Technology Trichy, India

³Department of Computer Science Engineering, Amrita University, Coimbatore, India

Received 2013-10-19; Revised 2013-10-30; Accepted 2013-11-13

ABSTRACT

A picture or image is worth a thousand words. It is very much pertinent to the field of image processing. In the recent years, much advancement in VLSI technologies has triggered the abundant availability of powerful processors in the market. With the prices of RAM are having come down, the databases could be used to store information on the about art works, medical images like CT scan, satellite images, nature photography, album images, images of convicts i.e., criminals for security purpose, giving rise to a massive data having a diverse image set collection. This leads us to the problem of relevant image retrieval from a huge database having diverse image set collection. Web search engines are always expected to deliver flawless results in a short span of time including accuracy and speed. An image search engine also comes under the same roof. The results of an image search should match with the best available image from in the database. Content Based Image Retrieval (CBIR) has been proposed to enable these image search engines with impeccable results. In this CBIR technology, using only color and texture as parameters for zeroing in on an imagemay not help in fetching the best result. Also most of the existing systems uses keyword based search which could yield inappropriate results. All the above mentioned drawbacks in CBIR have been addressed in this research. A complete analysis of CBIR including a combination of features has been carried out, implemented and tested.

Keywords: Image Processing, Search Engines, CBIR

1. INTRODUCTION

Search engines are always expected to deliver flawless results in a short span of time. Success of a search engine depends on its accuracy and speed. Image search engines also fall under the same category. The results of an image search should match with the best available image from the database. Content Based Image Retrieval (CBIR) is augmentation to the existing existing image search strategy. CBIR might have been very expansive and prominent in the academics arena, but it has not grown up to that extent in the commercial sector. For getting clear-cut understanding about the existing search engines in the market and the underlying methodologies adopted in each of them, a literature survey has been carried out. Here is a glimpse of all the commercial tools studied.

CBIR has been evolving consistently for over a decade. The improvements in CBIR were based on properties of the image like color, texture and shape. Few recent improvements in CBIR use relevance feedback methodology. In this method, the feedback from the user is used for filtering the images. Only limited parameters such as color and texture have been taken into consideration for constructing CBIR based search engines. Color and texture are not the only parameters that can help us in fetching the perfect result. Also most of the existing systems uses keyword based search which could yield inappropriate results. All the above mentioned drawbacks in CBIR have been addressed in the research that is carried out. A complete analysis of CBIR including a combination of features has been carried out, implemented and tested.

Corresponding Author: Shriram, K.V., Centre for Research and Development, PRIST University, India

2. LITERATURE SURVEY AND SHORTCOMINGS OF THE PRESENT METHODOLOGIES IN CBIR

There are many search engines available for usage. An analysis of the way they work is shown and results do not meet the expectation. The search engines used are (1) Yahoo search, (2) Bing search, (3) Google search engine for the testing purpose. In the same order as mentioned above the searching has been done and the obtained results have been presented in **Fig. 1-3**. The search was done for the query-“Sachin Tendulkar with cooling glass”.

2.1. Google Image Search

Yahoo Image Search: **Figure 2** is also as bad as **Fig. 1** with results which are not of interest to the searching person.

2.2. Bing Search-Beta Version

From all the above three searches done in three famous image search engines, one can understand that the searching methodology has an explicit shortcoming which has to be addressed (Shriram *et al.*, 2012a). The aim of the research is to propose a methodology which overcomes the problems faced in such existing search engines.

Apart from the search engines that have been prevailing in the market, a good amount of papers have been analyzed and read through. Though many papers were read in the process of carrying out literature survey, few are found to be worth briefing which are relevant to the research being currently carried out.

Su *et al.* (2011) in this study, the authors have integrated data mining with the image processing by including clustering, transformation and navigation techniques. Query processing phase, image search phase, knowledge discovery phase, data storage phase are the phases in which they approach the problem.

Smeulders *et al.* (2000) the authors considered the description of content in two steps. First, they discussed the image processing operations that transpose the image data into another spatial data array and divided the methods over local color, the local texture, or local geometry. They considered cultural, geometric, physical, literal, perceptual, categorical features for extracting the images. They also considered the query space definition and initialization.

Stanescu and Burdescu (2005) the system the authors have discussed the insertion of new images in the database and also two ways of querying the database: Simple text-based and content-based visual query, taking

into consideration the color. In order to obtain better results they considered, the calculation of mode of the distances between the query and target image, namely: The histograms intersection, Euclidian distance between histograms and the quadratic distance between histograms.

Satya *et al.* (2007) in this study, the authors have discussed the image retrieval process in three steps, namely, feature extraction, image matching, feedback from user. Based on the feedback relevance, the query is further narrowed down for obtaining more accuracy. They considered color, texture and shape properties for retrieving the images from the database.

Khan *et al.* (2011) the authors of this study have tried to get a search engine with external query being supported. The methodologies the authors have followed are color and texture alone and nothing more than that. Also there is no justification on why the same has been selected. Matlab has been used for the implementation part.

Kondekar *et al.* (2010) this study is almost the similar the previous one with some minor changes in the approach. Here the texture, color and shape have been taken into consideration. But no justification has been made on in which order the feature comparison has to be done and statistics are also not presented. Matlab has been used for this implementation as well.

Considering all these observations and analysis, the research is aimed at creating a better methodology for CBIR with better accuracy and speed. Also we have been considering several other features for the research. We referred many more papers in this area. All of them are indicated clearly in the references section.

3. PROBLEM STATEMENT

“The problem involves entering an image as a query into a software application that is equipped to employ CBIR techniques to extract visual properties and match them. This is done in order to retrieve the images in database that are “visually similar” to the input image”.

Retrieving images in text based image search has been very inaccurate. CBIR has proven to be very accurate and efficiency rate has been tremendously increased. So, researchers have been continuously working on making CBIR a more efficient one. The proposed research also has aimed towards a better working model for CBIR. Focus has been given to spatial domain and better search results are obtained. Spatial meaning, analysis of the pixels in the image which contain the color, position and the count. With these parameters taken into consideration one can get a better search engine created effectively and our thesis also aims at the same.

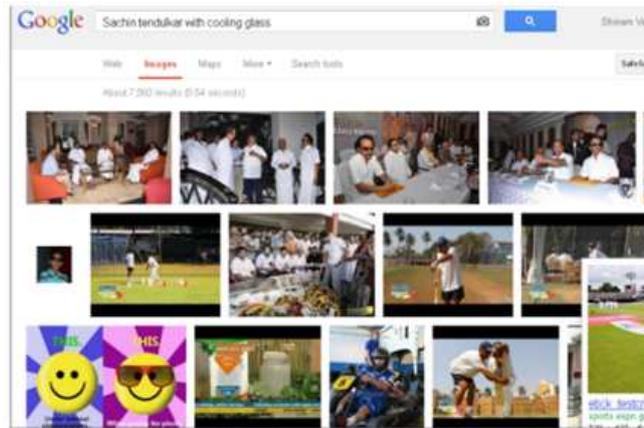


Fig. 1. Results obtained from Google image search



Fig. 2. Results obtained from Yahoo image search

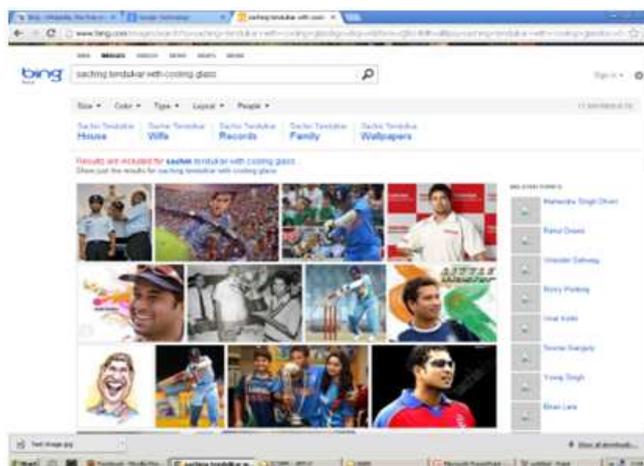


Fig. 3. Results obtained from Bing image search

4. PROPOSED METHODOLOGY

Since the legacy method which is followed for building up CBIR is proven to be lacking perfection, new architecture incorporating methodologies discussed above is presented in **Fig. 4**. The following are the steps involved in the image matching and retrieval. This provides a basic idea on how retrieval is carried out. This is not the final model to be followed for a perfect retrieval. There needs to be a lot research done in detecting order of features to be used and the same has been presented in detail at the later stages of the paper. But this has taken a lot of refinement and changes to arrive at the final sequence of the methodologies to be followed for efficient CBIR. This sequence is just a template over which the system has been constructed.

Following are the methodologies involved and the same has been diagrammatically drafted in **Fig. 4**.

The color histogram represents the color distribution in an image. This histogram represents the number of pixels that have color values that span the image's color space, i.e., the set of all possible colors. One such histogram generated for a color image and is presented below in **Fig. 5**.

The color histogram can be calculated from the RGB planes of the image. Pixel values in all the 3 planes are calculated individually and plotted (Shriram *et al.*,

2012b). The histogram calculation is similar to the grey scale image calculation; the difference is that the calculation is to be performed in all the three planes. The different plane histogram is as shown in **Fig. 6**.

4.1. Entropy

Entropy is defined as the amount of information contained in the image. It used to identify the redundancy in an image. Entropy is a facilitator to get the set of images which are closer to the fed input image as mentioned earlier. In simple terms, the number of different colors present in the image can be found out with entropy calculation:

$$\text{Entropy} = x (\log x)$$

The information from an image can be modeled as a probabilistic process. A random event (pixel value) 'E' that occurs with probability P(E) (the probability of occurrence of the pixel values) is said to contain information $I(E) = \log(1/P(E))$ where I(E) is called the self-information of the image. If $I(E) = 0$ then there is no information. If the logarithm is considered as base 2, then the information is expressed as bits. The entropy can also be called as the uncertainty in the information present.

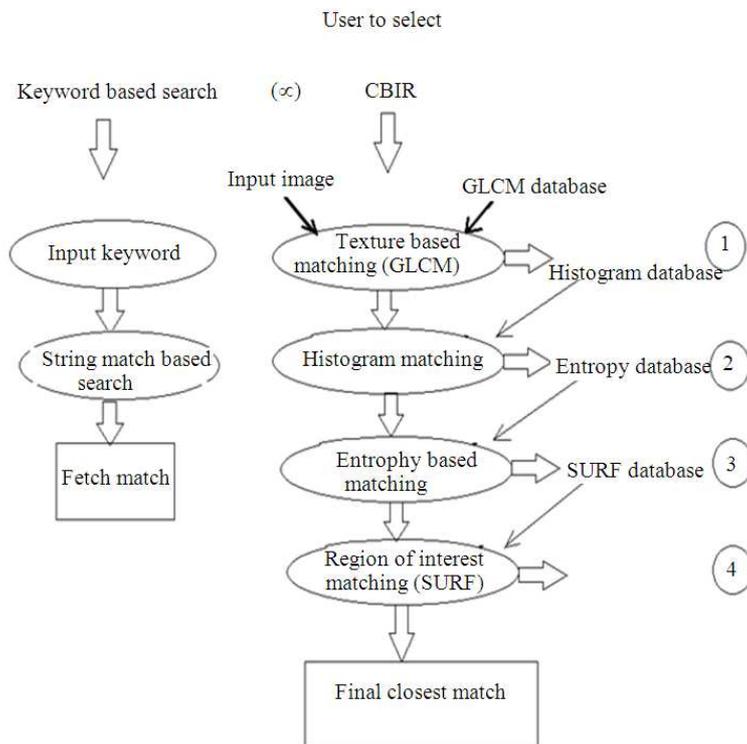


Fig. 4. Proposed architecture for search engine

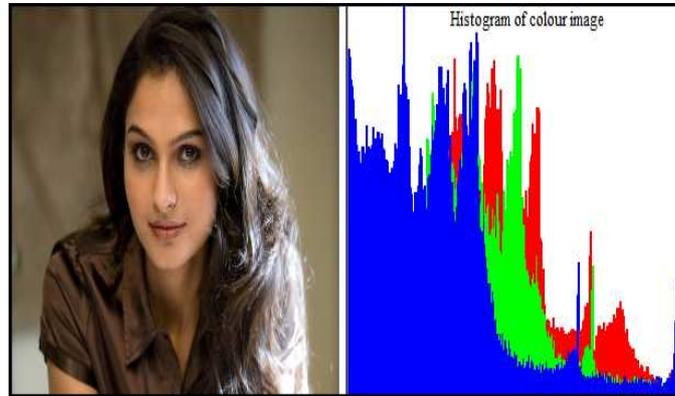


Fig. 5. Histogram of a color image

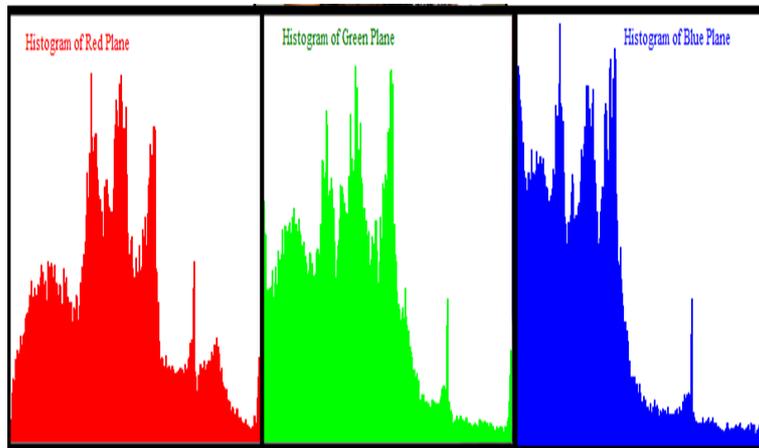


Fig. 6. RGB planes of color image shown in Fig. 5

The entropy is given by the expression:

$$\sum_{l=1}^L p(al) \log(p(al))$$

Using the entropy information of an image we can get the total distribution of the information in the image. In the proposed system the entropy of the entire database images is retrieved and stored in a text file once the query image entropy is obtained it is compared using the Norm 1 or the Euclidean distance formula:

$$d(p, q) = d(q, p) = \frac{\sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2}}{\sqrt{\sum_{i=1}^n (q_i - p_i)^2}}$$

Once the distance is calculated for the compared images the least distance measure gives us the

appropriate match. Those images are the closest match to the given query image in term of the probabilistic distribution of the pixels.

4.2. Texture (GLCM) Grey Level Co-Occurrence of an Image

GLCM is the method used to obtain the second order statistical texture features; this approach has been used in various applications. Another simple example is presented in Fig. 7.

A GLCM matrix rows and columns indicate the no. of grey levels in the image or in other words the no. of grey levels is the size of the matrix. The matrix elements $P(i, i|dx, dy)$ is the relative frequency with which two pixels are separated by a pixel distance (dx, dy) , occur within a given neighborhood. I and j indicate the intensity of the 2 pixels considered for the frequency measure.

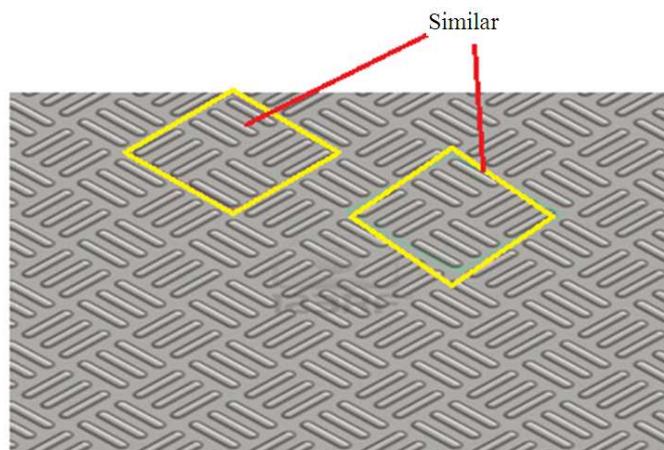


Fig. 7. Texture

Once the GLCM matrix has been obtained then the various features can be obtained like the energy, entropy, homogeneity, contrast and various other features. Energy of the GLCM matrix is given by:

$$\sum_i \sum_j P^2(i, j)$$

Energy is 1 for a constant image. The energy gives the squared sum of the probabilities.

4.3. Homogeneity

It is a measure of closeness of the distribution for elements in the GLCM. Its range is from 0 to 1:

$$\sum_i \sum_j \frac{P(i, j)}{(i + |i - j|)}$$

4.4. Correlation

It is the measure of how correlated the pixel to its neighbor over the whole image.

Range is between -1 and 1:

$$\sum_{i,j} \frac{(i - \mu_i)(j - \mu_j)P(i, j)}{(\sigma_i \sigma_j)}$$

4.5. Contrast

It is the measure of intensity of contrast with respect to its neighbors over the entire image:

$$\sum_{i,j} |i - j|^2 p(i, j)$$

It ranges between 0 and $(\text{size of GLCM} - 1)^2$.

4.6. Region of Interest (ROI)

ROI (Speed up Robust Feature) is one of the most frequently used interest point detectors and descriptors which were developed by Bay *et al.* (2006). The Speeded up robust features algorithm is a scale and rotation-invariant interest point detector and descriptor which is computationally fast. To make it computationally fast, it uses Integral images. The key points are detected by using a Fast-Hessian matrix. The descriptor describes a distribution of Haar-wavelet.

5. ASSUMPTIONS AND ENVIRONMENT USED

Images ranging from 5000 to 25000 have been used as database for the offline CBIR technique proposed. The images are heterogeneous in nature with different file formats. The images are of the formats a.JPG b.BMP c.PNG and c.TIFF. The content of the images are Fruits and vegetables, Butterflies, Birds and Natural sceneries. The images are not stored in a grouped manner; instead all the images are randomly stored. This will help in understanding how efficient the search engine designed will work.

For carrying out the complete research Desktop machines with 2 GB RAM has been used. Intel Core 2 Duo is the processor that was in place in the machine. Efficiency has been tested later in a downgraded setup with having Intel Pentium P4 with 1 GB RAM. There was no significant performance degradation in this second case. The Ubuntu (Linux) is the operating system used. Initially for understanding the histogram MATLAB was used. But later on for a better result a navigation has been made to OpenCV, which is a freeware.

Following **Fig. 8** shows the basic design of our CBIR system followed in Matlab. This has been used as a basic design which has been taken further to get a better design using OpenCV. Here as shown in the design, the retrieval of images happen in three stages, in the initial stage, we retrieved images with respect to the color, if the user is satisfied with the retrieval results, we stopped there itself, there by not giving much work to the system. If not, the retrieval was carried out based on the texture of the images. If the user is still not satisfied with the results, we calculated the entropy of images before comparing them. The results obtained were good but not excellent which motivated us to include the region of interest as well into

the search design which has been done with OpenCV which gave a better result when comparing.

Note: One very important point to note is, before uploading an image into the image database, we have to check its specifications. If the image doesn't correspond to our requirement of 256bit, we had to convert it into an image of bit depth equal to 8. We used `rgb2ind()` method for achieving this. One more important prerequisite to deal with images in MatLab is the image size. We chose the dimension 160*120. We used the built-in function `imresize()` to automatically convert the images before uploading into the database.

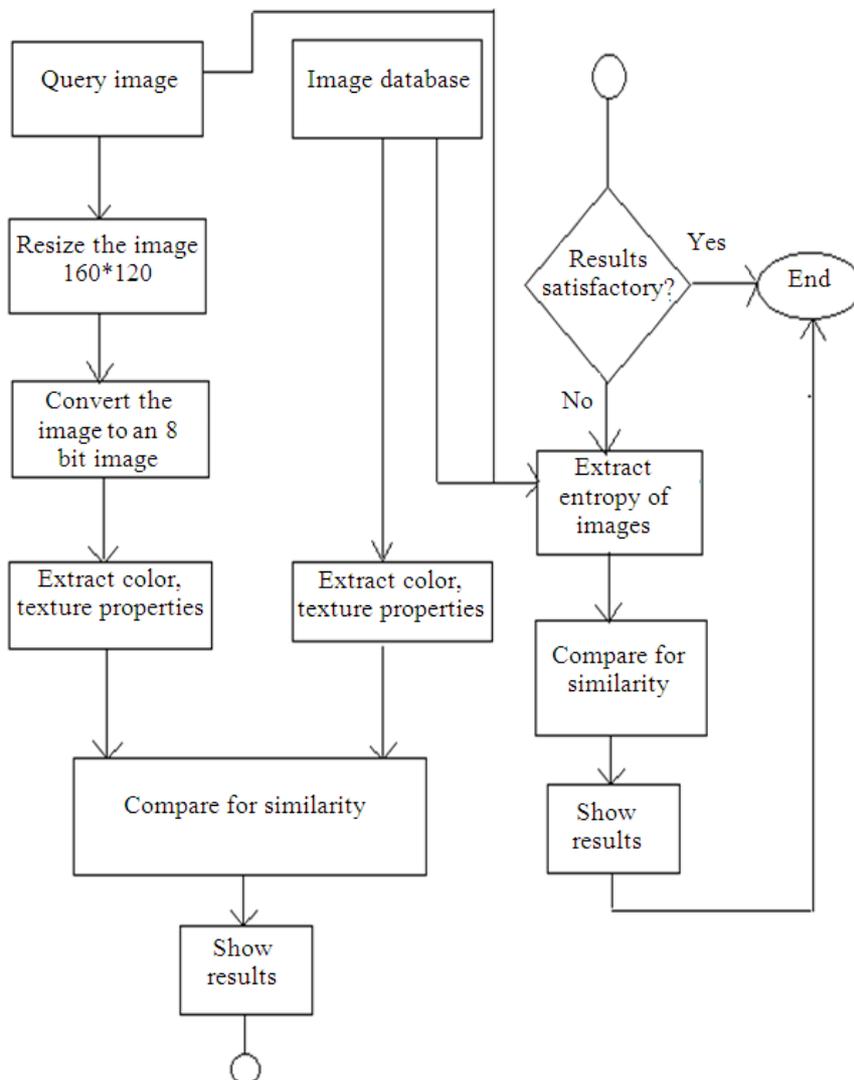


Fig. 8. Design diagram

After including Entropy, it is proven to be worthy and search was definitely better (Vasudevan *et al.*, 2012). Since there will be a better perfection with OpenCV later we have navigated to OpenCV to get the complete research carried out there and the details of the same have been provided in the following sections. Following are the features used in CBIR construction which is very novel (Shriram *et al.*, 2012a):

- Keyword based search
- Histogram
- Entropy
- Texture
- ROI (Region of Interest)

6. EXPERIMENTAL SETUP AND RESULTS

Following are the combinations (Table 1) of the methods which are found to be successful. Rest all the

methods are found to be inefficient or inaccurate. One such combination is also tested which comprised of Histogram, Entropy, ROI and Texture. It is also covered in the result analysis.

Only the sixth combination is explained here. The Flow diagram in Fig. 9 clearly helps us to visualize the schema of how input from one stage is passed on to the other which can be seen from the Fig. 10-13.

All the above tested combinations seem to be very effective and here is the summary of the successful combinations presented:

- Histogram, Texture, Entropy and ROI
- Histogram, Entropy, Texture and ROI
- Entropy, Histogram, Texture and ROI
- Entropy Texture Histogram and ROI
- Texture Entropy Histogram and ROI
- Texture Histogram Entropy and ROI

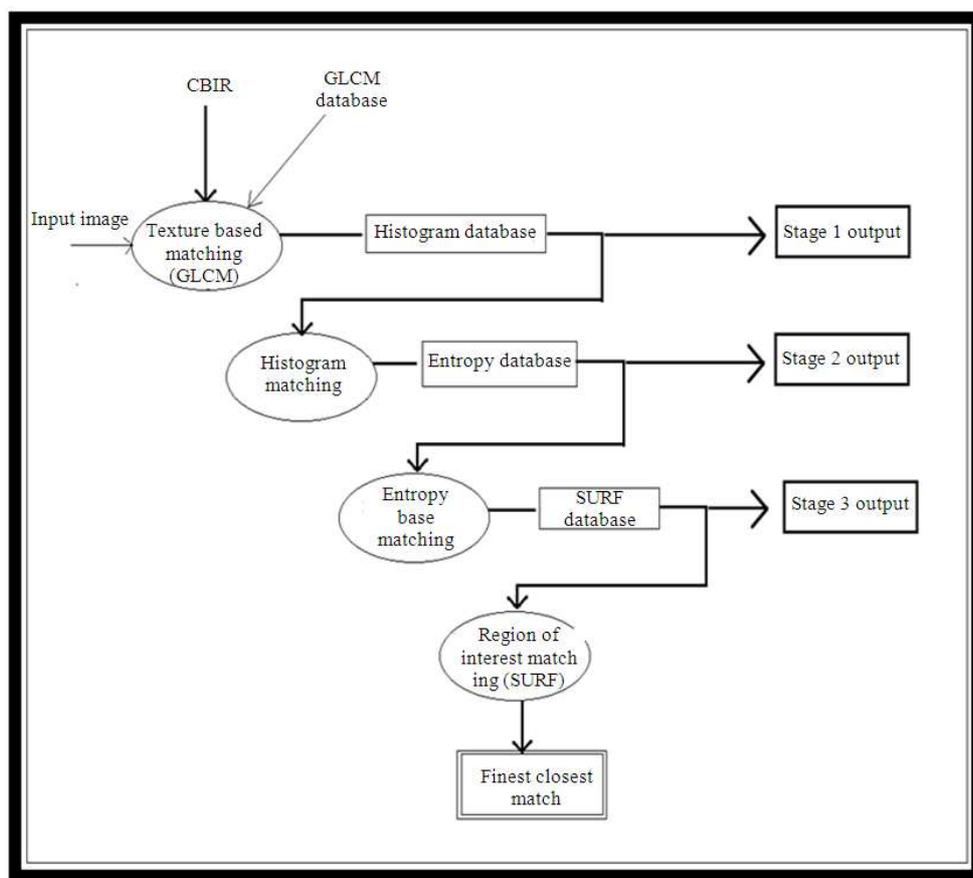


Fig. 9. Flow diagram with combinations



Fig. 10. Level 1 filtered results

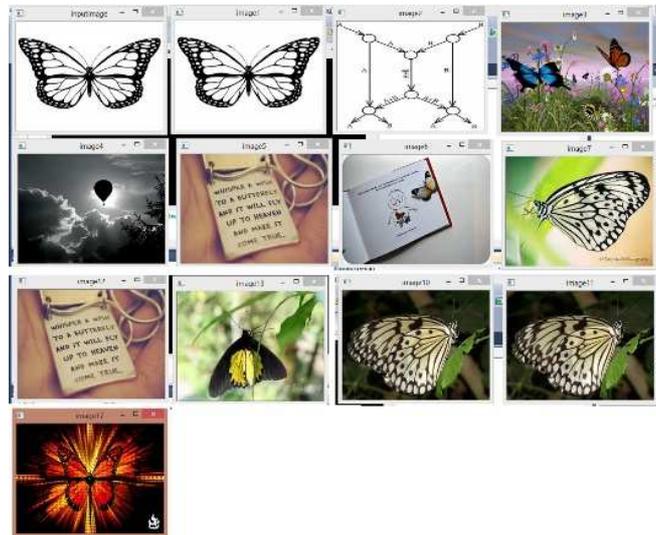


Fig. 11. Level 2 filtered results

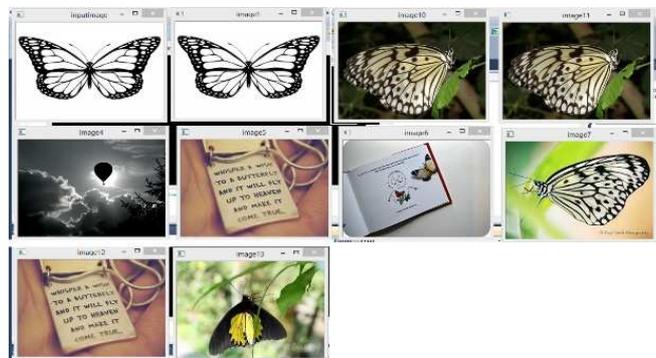


Fig. 12. Level 3 filtered results

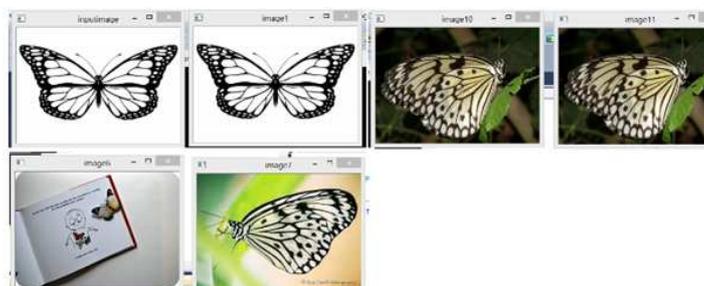


Fig. 13. Level 4 filtered results

Table 1. Combination of methodologies used

| |
|--|
| (Attempt-1 with Histogram, Texture, Entropy and ROI) |
| (Attempt-2 with Histogram, Entropy, Texture and ROI) |
| (Attempt-3 with Entropy, Histogram, Texture and ROI) |
| (Attempt-4 with Entropy, Texture, Histogram and ROI) |
| (Attempt-5 with Texture, Entropy, Histogram and ROI) |
| (Attempt-6 with Texture, Histogram, Entropy and ROI) |

If ROI is used first, the time taken to get the best match for so many images, say about 20,000 will be very huge. The core aim of any search engine is to get the searching done at the earliest with the best possible result i.e. the match. In the event of using ROI as the level-1 filtering option it would be computationally very much intensive to filter best images from all the available images. Eventually it would take more time which is not the expected outcome. That is the reason, the research did not use ROI as the first option in the searching methodologies. Also, the search engine should give first best match, next best match and should go on. In the event of using ROI in the first level and rest of the methods subsequently, the results obtained are poor after the first closest match. All other combinations other than above mentioned are found to be failing to get best results.

7. TESTING, ANALYSIS AND INFERENCES

After an intensive research being carried out with the methodologies for CBIR like Histogram Texture, Entropy and ROI. It is observed that the following combinations are found to be effective in producing effective results:

- Histogram, Texture, Entropy and ROI
- Histogram, Entropy, Texture and ROI
- Entropy, Histogram, Texture and ROI
- Entropy Texture Histogram and ROI
- Texture Entropy Histogram and ROI
- Texture Histogram Entropy and ROI

All the below mentioned results are with a PC with OpenCV in place. To get an understanding about the performance, the test cases were used, the following graphs

are constructed. One can understand from the graphs drafted below that, above prescribed 6 combinations would yield better results when compared to the 7th combination, which is not the best choice. The below result is from a database with 20,000 images. The results obtained from all the graphs are all self-explanatory.

Above graph **Fig. 14** has the accuracy comparison done with all the combinations.

Figure 15 shows the time taken by each combination and **Fig. 16** deals with results with different database sizes.

Figure 17 shows the result obtained with different image contents.

Figure 18 shows a comparative result between performance in embedded and PC.

The following graph has images ranging count from 5000, 10000, 15000 20000 and 25000 images in the database.

The next level of analysis with different images with following contents in the image has been tested and results are presented in the graph shown below.

Since it is always a best practice to test the software or hardware that is being developed or designed has to be tested under various tough operating conditions, this research has also been subjected to same kind of testing. The following table has the details of the test cases and testing carried out in PC as well as embedded platform along with the results:

- PC with 1000 images: It was tested with 1000 images in the database and the result was pass and the behaviour was consistent and precise. Time elapsed: 22 sec
- PC with 10000 images: It was tested with 10000 images for the image search and the result is pass and the behaviour was consistent and successful. Time elapsed : 23.5 sec
- PC >10000 images and <25000 images: It was tested with about 20000 images and the result was pass. Time elapsed varied between 25 to 28 seconds to fetch the best results

- No image: With the no image in the database the search failed to find any match and the result is hence pass
- PC with 2GB RAM: 15000 images were tested and the result is pass with 22 sec
- PC with 4GB RAM: 15000 images were tested and the result is pass with 20 sec
- Immediate check on system boot: Result is pass with 22 sec and the search was tested with 15000 images
- Repeated search for the same query. It was tested for 15000 images and the search settles down in 19 sec
- Stress testing by using the different input images for search: The images were tested and the results were obtained between 20-25 sec
- System forced to sleep/hibernate and on waking up image search to be carried out: It was tested with 20000 images and the result is pass with 22 sec
- Testing with multiple applications being launched in parallel: It was tested with 20000 mages where there was no impact in the performance and the result was pass with time elapsed being 20 sec
- Using keyword based search with CBIR: The result was pass
- Using irrelevant keyword based search with CBIR: The result was fail

8. FUTURE PROSPECTS

The necessity of having a better search engine for image with more precision is obvious. Especially with the trend towards smart phones, faster ways of image search would benefit the mobile industry to a great extent. One such search engine is crafted which can be used with both the smaller computational engine and PC as well.

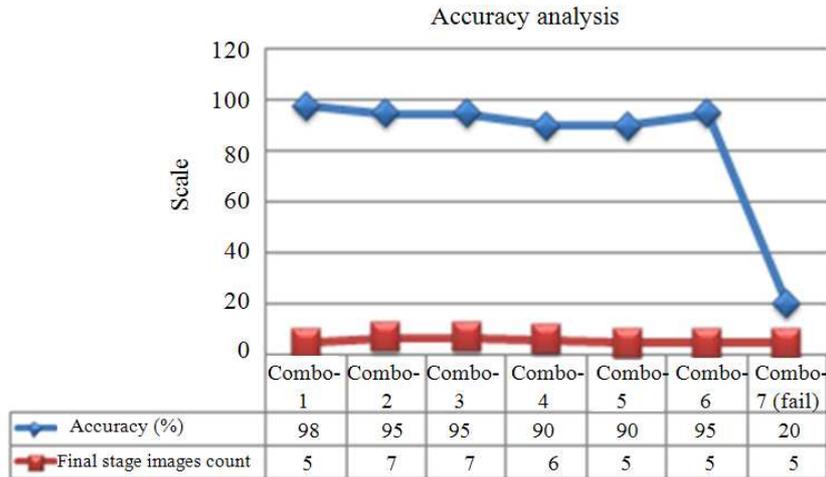


Fig. 14. Accuracy comparison-all the combinations

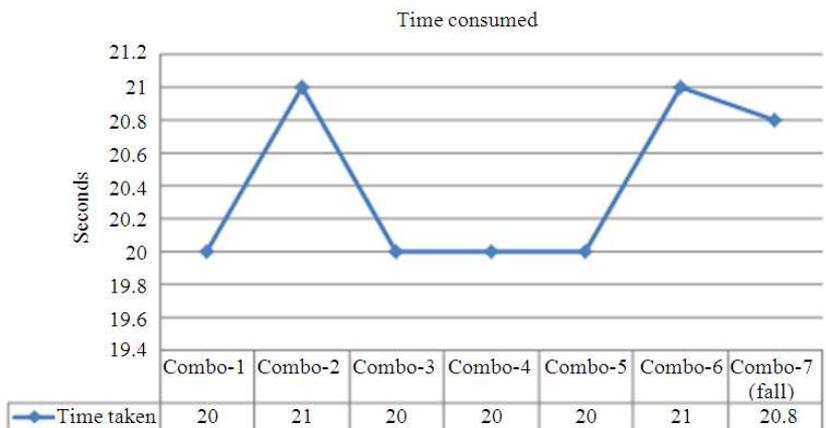


Fig. 15. Time taken-all the combinations

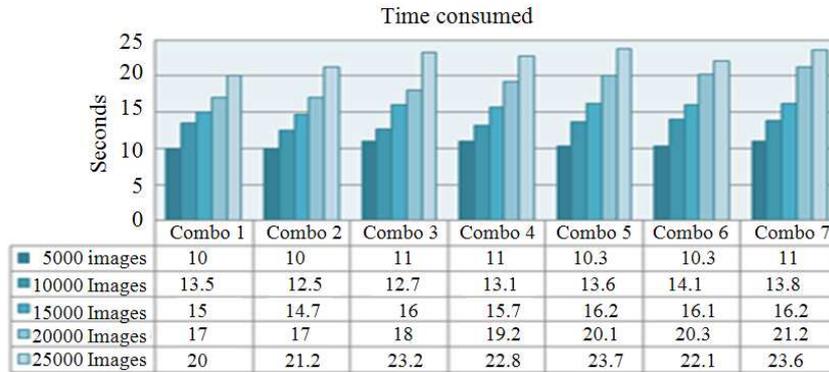


Fig. 16. Time consumed-different database sizes

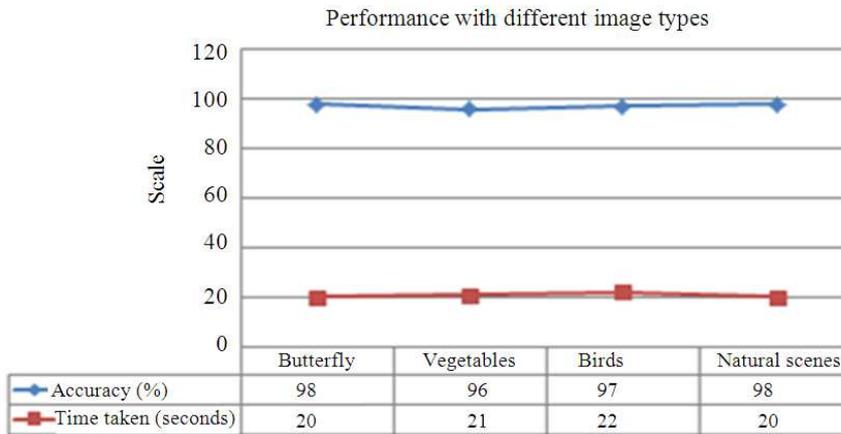


Fig. 17. Comparison with different image content

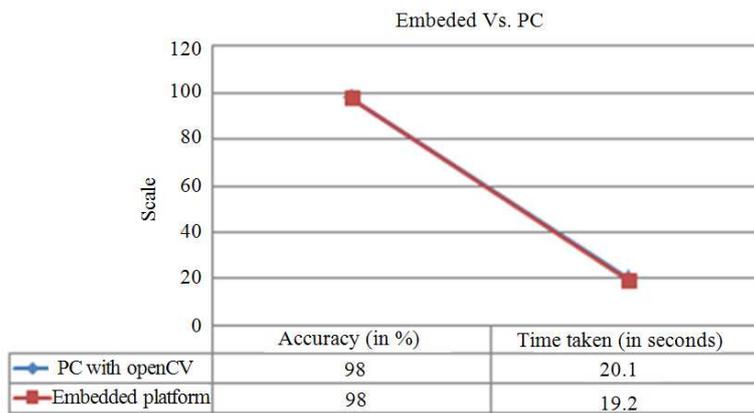


Fig. 18. Performance comparison on embedded Vs. PC

The searching mechanism includes Keyword based search, Histogram, Texture, Entropy and Region of Interest (ROI) methodologies which itself is a novel approach and proved to be worthy in using with CBIR. In addition, the research has been extended to deciding the best order of invocation

of methodologies. Results reveal that six of the four methodologies fetch results with good accuracy and within an appreciable timeframe as well.

The keyword based search even though not very accurate it's presence to a great extent helps in

narrowing down the images to be searched as most of the times the image we need is named appropriately. A user can avail or waive this feature. Further refining the output we start identifying and matching textures. By identifying analysing and matching repeating patterns and by analysing its relative position to each other we are able to further increase the accuracy.

Also the research has driven us to the conclusion that with Region of Interest (ROI) based matching done in the first level, the engine responds very slowly and hence it is not a wise option to have the ROI at the level-1. Once done with the PC (Linux/Ubuntu), the journey was towards porting the CBIR to a smaller, mobile computing engine. Immediate, obvious choice was Beagle board which supports OpenCV. The code ported to Beagle board worked fine and the results were a repeat of the PC with OpenCV.

Coming to the improvement prospects and future enhancement opportunities, the above methods have great potential and serve as the forerunner of many methods to come. The same CBIR methodology can be moved to video searching. Selective information retrieval from videos can be done i.e., say a particular car image from the movie is to be retrieved it can be accomplished using the same. Also it can be used in applications in military projects like advanced face recognition, image decoding and morphing. As a milestone in CBIR based search engine, it can also be moved to online image searching which again if adopted in a mobile platform can take us one step closer to making mobile operating systems equal to ones in PC's. It would be another interesting area of research in search engines.

9. CONCLUSION

Most of the previous researches have taken Histogram and Texture alone for getting the best match. As a matter of fact, in this research the CBIR has been tested with smaller computational engine and it was proven to be worthy as well where we have proved using an optimized set of attributes for the image in the database is enough to produce the best results where we have taken proved that the size of the database also is not exponentially large. A software based search engine which is highly capable of retrieving images based on the shape, texture, entropy and region of interest has been developed considering that these are the only set of attributes that need to be taken into consideration. Complete information and details on how all of the above have been performed is presented in this study. All the scenarios have been tested and results have been analysed as well.

10. ACKNOWLEDGMENT

We thank PRIST University, Centre for research and development for providing us continuous support and

guidance in all possible way for this research. We thank all the reviewers of this study for their helpful comments and suggestions.

11. REFERENCES

- Bay, H., T. Tuytelaars and L.V. Gool, 2006. SURF: Speeded up robust features. *Comput. Vis.*, 3951: 404-417. DOI: 10.1007/11744023_32
- Khan, W., S. Kumar, N. Gupta and N. Khan, 2011. A proposed method for image retrieval using histogram values and texture descriptor analysis. *Int. J. Soft Comput. Eng.*, 1: 33-36.
- Kondekar, V., V. Kolkure, G. Sodal and J. Mudegaonkar, 2010. Image retrieval techniques based on image features: A state of art approach for CBIR. *Proceedings of the International Conference and Workshop on Emerging Trends in Technology*, Feb. 26-27, ACM Press, Mumbai, India, pp: 998-999. DOI: 10.1145/1741906.1742145
- Satya, K., S. Prakash and R.M.D. Sundaram, 2007. Combining novel features for content based image retrieval. *Proceedings of the 6th EURASIP Conference Focused on Speech and Image Processing, Multimedia Communications and Services*, Jun. 27-30, Maribor, pp: 373-376. DOI: 10.1109/IWSSIP.2007.4381119
- Shriram, K.V., P.L.K. Priyadarsini and V. Subashri, 2012a. CBIR-An analysis and suggestions for improvement. *Int. J. Comput. Applic.*, 42: 41-44. DOI: 10.5120/5764-7967
- Shriram, K.V., P.L.K. Priyadarsini and V. Subashri, 2012b. An Efficient and Generalized approach for content based image retrieval in MatLab. *Int. J. Image, Graphics Signal Process.*, 42-48. DOI: 10.5815/ijigsp.2012.04.06
- Smeulders, A.W.M., M. Worring, S. Santini, A. Gupta and R. Jain, 2000. Content-based image retrieval at the end of the early years. *IEEE Trans. Patt. Anal. Mach. Intell.*, 22: 1349-1380. DOI: 10.1109/34.895972
- Stanescu, L. and D.D. Burdescu, 2005. On-line software system for content-based visual query of a color medical imagery. *Proceedings of the AC IADIS International Conference on Applied Computing*, Algarve, Portugal, Feb. 22-25.
- Su, J.H., W.J. Huang, P.S. Yu and V.S. Tseng, 2011. Efficient relevance feedback for content-based image retrieval by mining user navigation patterns. *IEEE Trans. Knowl. Data Eng.*, 23: 360-372. DOI: 10.1109/TKDE.2010.124
- Vasudevan, S.K., P.L.K. Priyadarsini and S. Vasudevan, 2012. Content Based Image Retrieval (CBIR): A deeper insight. Lambert Academic Publishing, Germany, ISBN-10: 3659171727, pp: 56.