

Content Based Image Retrieval: Multiple Features Comparative Study

Nora Youssef ¹, Alsayed Albergawy ², Ibrahim F. Moawad ¹, EL-Sayed M. EL-Horbaty ¹

December 13, 2017

Abstract

Multimedia resources are growing rapidly with a huge increase of visual contents. Thus, Searching these images accurately, and efficiently for all types of datasets becomes one of the most challenging tasks. Text-based image retrieval (TBIR) methods involve searching images based on their metadata. Disadvantages of this approach they require huge labor of manual annotation, have been annotated subjectively, limit the search process to metadata, and finally have the synonymy problem. Content-based image retrieval (CBIR) could be a solution for these problems, where it retrieves images based on its visual contents. This paper presents a CBIR system that extends the LIRe open source project in biodiversity domain. The proposed system has four main contributions. Firstly, it generalizes LIRe crawler to work on ImageNet synsets. Secondly, to speed LIRe crawler parallel approach has been included. Thirdly, extends the LIRe analyzer part by implementing new image feature extractors. Finally, it enhances retrieval accuracy by combining multiple features. The proposed system has been evaluated on the ImageNet subset and the experimental results show that the crawling process has been increased by 2.9 times and the retrieval accuracy has been enhanced using the combined features.

1 Introduction

Different forms of multimedia resources (i.e. text, images, audio, and videos) are rapidly growing with a huge development of visual contents and technologies, e.g. data visualized in smart phones, 2D/3D applications, web content, telecommunication, etc. The new century has witnessed an evolution in the amount, availability, complexity, variety and importance of images in all domains. Thus, images play an essential role in a wide range of applications and fields such as education, medical care and social media. However, visual media require a considerable amount of processing and storage, which requires a highly efficient method to index, store, analyze, and retrieve visual information from image databases. Moreover, images have a little metadata by which they can be indexed and searched conveniently. Consequently, searching images rapidly, accurately, and efficiently for all types of image datasets becomes one of the most challenging tasks [1].

The traditional image retrieval techniques are text based image retrieval (TBIR). So that, users can retrieve images based on keywords or textual descriptions which are annotated by human. TBIR systems have main drawbacks (1) huge human labor for the manual annotation, as when a giant data stores are taken into consideration. (2) The subjectivity of the annotation which affects the retrieval accuracy badly, i.e. captions or any other metadata are entered manually. (3) The

¹Faculty of Computer and Information Sciences, Ain Shams University, Egypt, {nora.youssef, ibrahim.moawad, s.horbatty }@cis.asu.edu.eg

²Friedrich-Schiller University of Jena, Germany, alsayed.albergawy@uni-jena.de

search is limited to the image text metadata which could produce false negative i.e. if an image misses a part of the metadata or false positive results i.e. if an image contains a wrong metadata, both cases may probably happen by mistake. In addition, a word could have different meanings like Jaguar it may be an animal or a car, which would defiantly affect the search results [1–4].

The term content-based image retrieval (CBIR) appears to have been first used in the literature by Kato [1992] to describe his experiments in the automatic retrieval of images from a database by color and shape. [3] CBIR has been introduced as a solution for the TBIR limitations, it is considered as, the technique of effective retrieval of images from an oversized data store [1–4]. The typical CBIR system performs two major tasks. The first one is feature extraction, where a set of features is extracted to describe the content of each image in the database. The second task is the similarity measurement between the query image and each image in the database, using the feature extraction. The feature extraction values for a given image are stored in a descriptor that can be used for retrieving similar images. Image descriptors are descriptions of the visual features of the contents in images that produce such descriptions. They describe elementary characteristics, such as color, texture, shape and spatial location [3]. There are various examples of CBIR systems: LIRe [5], VisualSeek [6] and SIMPLICITY [7].

The proposed is a CBIR for biodiversity domain. It has two main objectives, in other words, two main modifications of LIRe system. On one hand, crawler modification: firstly by replacing the flickr dataset by a dataset that suits the biodiversity domain. Secondly, the dataset is fed into the system via a user input. Finally, increase the crawler speed. The second main objective is to enhance the retrieval quality by building the image descriptor by a mixture of different image features. The rest of the paper is organized as: Section 2, related work. Section 3, introduces the LIRe system in details. Section 4, proposes the methodology and how the LIRe system has been customized. Section 5, the discussion and the experimental results. In addition the system evaluation. Finally section 6, concludes the paper.

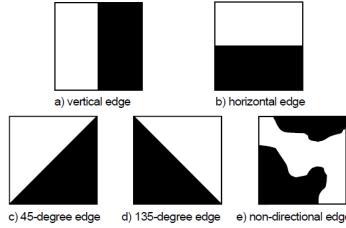


Figure 1: Five edges types

2 Image Descriptors

Visual image descriptors or image low level features are descriptions of the visual contents in an image or video. They are algorithms or applications that produce such descriptions. They describe elementary characteristics such as the shape, the color, the texture, edge or spatial location [8]. Thus, feature extraction and representation is an important step for multimedia processing [9]. Histogram is the most commonly used structure to represent any global feature composition of an image. It is invariant to image translation and rotation, it is also considered a very useful for indexing and retrieving images [10–12]. In this paper we will focus on edge, color and fuzzy color texture histogram and the combination among them.

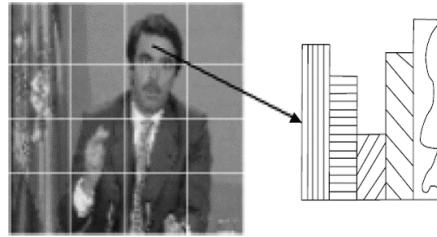


Figure 2: Five edges types

2.1 Edge Histogram Descriptor (EHD)

Edge in image is an important low-level feature. It can describe both shape and texture features, which are essential elements for content-based image analysis [12, 13]. The Edge Histogram Descriptor (EHD) describes ve edge types in the image,namely horizontal,vertical,two diagonal and non-directional edge types as shown in figure 1. Population for these ve types of edges in a local image region of a sub-image is represented by a histogram with ve bins. Specifically, the image space is divided into 16 (44) non-overlapping sub-images and for each sub-image a histogram with ve edge bins is generated as in figure 2, yielding a combined edge histogram with 80 (165) bins for the entire image. The histogram with 80 bins is the standardized edge descriptor for MPEG-7, figure visualizes the EHD 3 [2,11,13,14].

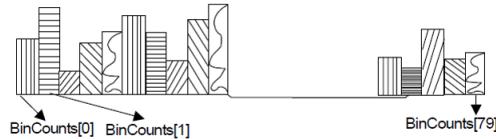


Figure 3: Edge Descriptor Histogram

However, using the local histogram bins only may not be sufficient to represent global features of the edge distribution. Thus, to improve the retrieval performance, we need global edge distribution as well. On one hand, the global edge histogram is a 5 bin histogram and it can be calculated by the summation of the same type of edge in the corresponding bins in local edge histogram as shown in figure 4 [2,11,13,14].



Figure 4: Global Edge Descriptor Histogram

On the other hand, the semi-global histogram could be generated as shown in figure 5 and it will yield 65 bin histogram. By using the combination of the three types edge, we would have a 150 ($80 + 5 + 65$) bin edge histogram. Finally the distance between 2 combined edge histograms

would be given by equation 1.

$$D(A, B) = \sum_{k=0}^{79} |Local_{A[i]} - Local_{B[i]}| + 5 \sum_{k=0}^4 |Global_{A[i]} - Global_{B[i]}| + \sum_{k=0}^{64} |Semi_Global_{A[i]} - Semi_Global_{B[i]}| \quad (1)$$

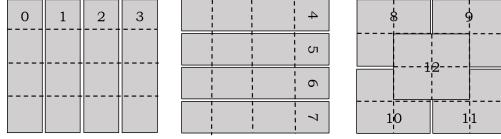


Figure 5: Semi-Global Edge Descriptor Histogram

2.2 Color Layout Descriptor (CLD)

Color is one of the most important features of images. Color features are defined subject to a particular color space or model. A number of color spaces have been used in literature, such as RGB and HSV [8]. For instance, the CLD is used in representation of the spatial distribution of colors in images. It is a compact descriptor that consists of representative colors and it is encoded using a non-overlap 8 x 8 sliding window over an color image followed by performing a Discrete Cosine Transform (DCT) [12, 15]. Figure ?? visualizes an example for color layout.



Figure 6: Visualization for color layout descriptor

Color layout (CLD) could be combined with other features for better retrieval accuracy. It could be combined with Gabor filter in wavelet domain for texture description [3]. Another idea proposed by [15], they combined a mixture of three color features at the same time for image retrieval, these features are dominant color (DCD), SCD that is like the traditional color histogram but in HSV color space and CLD. One more idea by [12] is to merge the CLD with EHD. In this paper, CLD have integrated with the combined edge histogram (Local + Global + Semi-Globa) for image retrieval and the similarity is calculated as in equation ??

$$D(A, B) = \sum_{k=0}^{79} |Local_{A[i]} - Local_{B[i]}| + 5 \sum_{k=0}^4 |Global_{A[i]} - Global_{B[i]}| + \sum_{k=0}^{64} |Semi_Global_{A[i]} - Semi_Global_{B[i]}| + \sum_{k=0}^{63} |YCoef_{A,B} - YCoef_{B,A}| + |CbCoef_{A,B} - CbCoef_{B,A}| + |CrCoef_{A,B} - CrCoef_{B,A}| \quad (2)$$

Such that $Local_{A,B}$, $Global_{A,B}$ and $Semi_Global_{A,B}$ are for combined edge histogram. On the other hand, the $YCoef_{A,B}$, $CbCoef_{A,B}$ and $CrCoef_{A,B}$ are for the CLD.

2.3 Fuzzy Color Texture Histogram (FCTH)

3 Experimental Evaluation

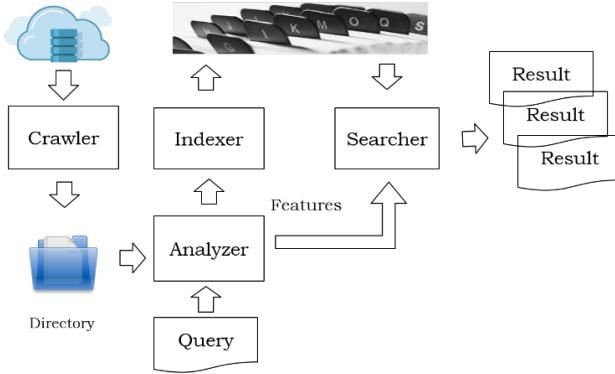


Figure 7: LIRE Block Diagram

3.1 LIRE System

LIRE (Lucene Image Retrieval) is an open source library for content based image retrieval, which means you can search for images that look similar. Besides providing multiple common and state of the art retrieval mechanisms LIRE allows for easy use on multiple platforms [5]. LIRE with its default architecture does not suit the biodiversity domain very well. This is due to the very general data-set which is just a collection of random images from the flickr data-set and the downloaded images don't relate to the biodiversity field. In addition we found the crawler is slow to some extent and this would be an issue at the biodiversity field, because we are usually dealing with huge data-sets. The analyzer extracts only one image feature at a time and the searcher measure the distance between the query image and the data-set against only one feature value, but as shown at [2, 11, 13, 16] we may mix multiple features at a time for better image description. Thus, we can enhance the retrieval accuracy.

The main components of the LIRE system is given by figure 7 which represents its block diagram. These components are: crawler, indexer, analyzer and searcher. Such that, the crawler craps a random number of images from the flickr dataset and save these downloaded images into a local directory/folder. The analyzer component extracts the images features using the state-of-the-art methods as mentioned above. The third component is the indexer component which takes the extracted image features and build a Lucene based index. The last component searcher or matcher takes the query image and calculate the distance or the similarity between the input and the images that are stored in the index, it then displays the most relevant images.

3.2 Evaluation Criteria

Precision and recall, F measure and mean average precision mAP are the basic measures that are used in evaluating search strategies. Assumes that there are two sets as in figure 8. The one on the right gives the set of items retrieved and the set on the left gives the set of relevant items for a query in the database. This gives the next three notations.

- A: Relevant items that were retrieved.

- B: Relevant items that not retrieved.
- C: Irrelevant items that were retrieved.

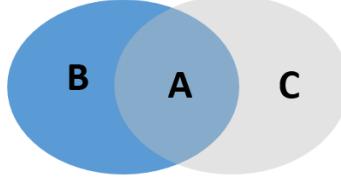


Figure 8: Precision and recall measurements illustration.

Precision: Equation 3 gives the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved. It is usually expressed as a percentage.

$$Pr = \frac{A}{A + C} \quad (3)$$

Recall: Equation 4 gives the ratio of the number of relevant items retrieved to the total number of relevant items in the database. It is usually expressed as a percentage.

$$R = \frac{A}{A + B} \quad (4)$$

F allows to trade off precision against recall. In this paper the balanced F with $\alpha = 0.5$ has been used.

$$F = \frac{1}{\alpha \frac{1}{Pr} + (1 - \alpha) \frac{1}{R}} \quad (5)$$

Moreover, the mean average precision mAP is given by equation 6 shows the average of precision value of all tested queries.

$$mAP = \frac{\sum Pr}{\#queries} \quad (6)$$

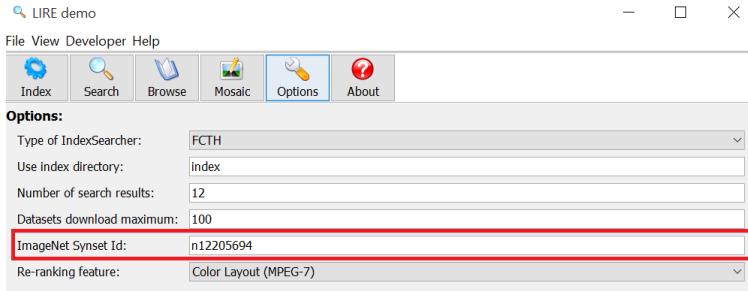


Figure 9: LIRE settings page.

Table 1: Selected synsets details

Synset	Id	#Images	Sample
Animal	n00015388	400	
Planet flora, planet life	n00017222	400	
Owl	n01621127	100	
Elephant	n02503517	100	

3.3 Dataset and Queries

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images [17]. In this paper, experiments were done on four different synsets of ImageNet, these synsets are given by table 1, that it gives an overview about the synsets used in terms of name, synset Id, number or downloaded images and a sample for each of them. The public API for downloading these synsets is given by <http://www.image-net.org/api/text/imagenet.synset.geturls?wnid=>. while the synset id or wnid is a user input as in figure 9.

Queries for the experiments are randomly selected that covers all the fours synsets figure 10 shows a sample for the selected queries.



Figure 10: A sample of the selected queries.

3.4 Results

The experiment includes 13 different queries, each of them has been tested against 6 algorithms. On one hand, the state-of-the-art: Color layout, local edge histogram, fuzzy color texture histogram (FCTH). On the other hand: Combined edge histogram, edge histogram and color layout (Edge + Color) and finally, edge histogram and FCTH (Edge + FCTH).

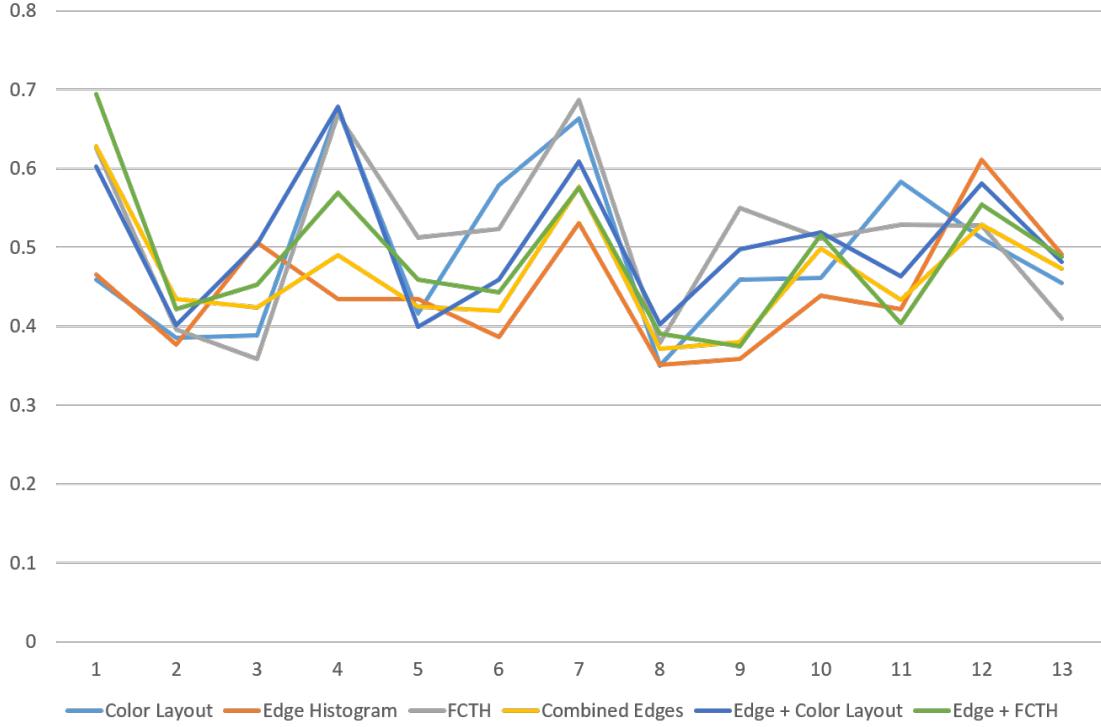


Figure 11: Precision (pr) graph for 6 algorithms and 13 queries.

Results are shown in terms of precision, recall, F measure graph and mean average precision mAP . figure 11 shows the average precision values of 10 users for each one of the 13 queries.

Query 1, Query 4 and Query 7 belong to Herbs and Elephant Synset. We have many images that matches the exact color and with duplicates, that's why the color feature exceeded either in Color or (Edge + Color) combined. FCTH appears as a second rank which color based feature but it takes the texture into consideration as well. In other words, adding something extra feature like edge was not the best option while we have sufficient amount of images that it could be retrieved by single feature. Thus, it actually included some false positives just cause it matches the query's edges.

Query 2 and Query 8 which are 2 queries for a cat and giraffe respectively, as noticed the 6 algorithms gave their minimum Pr values, this is due the lack of images that have either giraffe or a cat there is only 1 and 2 available matches in the downloaded synsets. In other words, this means most of the retrieved results are irrelevant.

Figure 13 shows the average balanced F measure of the 13 queries, such that $\alpha = 0.5$. Moreover, Figure 14 shows the mAP for them. It is shown that using multiple features like combined edges exceeded the local edge histogram and edge + color layout exceeded both of color layout and local edge as separate features. However, the combination of edge and FCTH exceeded local edge histogram but retreated the FCTH.

4 Conclusion and Future Work

Acknowledgment

This work was done at the SWEP course as a part of the BioDialog project which is funded

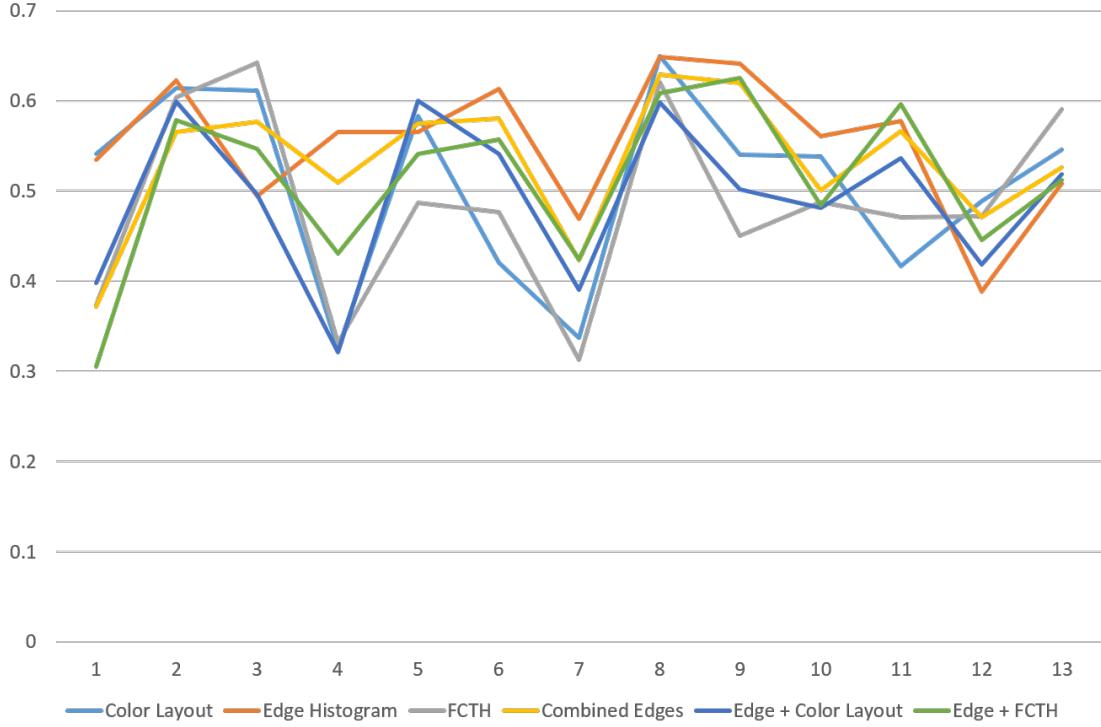


Figure 12: Recall (R) graph for 6 algorithms and 13 queries.

by DAAD at Friedrich Schiller University of Jena. Special thanks Prof. Birgitta Knig-Ries for her support.

References

- [1] A. Alzubi, A. Amira, and N. Ramzan, “Semantic content-based image retrieval: A comprehensive study,” *Journal of Visual Communication and Image Representation*, vol. 32, pp. 20–54, 2015.
- [2] C. S. Won, D. K. Park, and S.-J. Park, “Efficient use of mpeg-7 edge histogram descriptor,” *ETRI journal*, vol. 24, no. 1, pp. 23–30, 2002.
- [3] H. A. Jalab, “Image retrieval system based on color layout descriptor and gabor filters,” in *Open Systems (ICOS), 2011 IEEE Conference on*. IEEE, 2011, pp. 32–36.
- [4] M. Srinivas, R. R. Naidu, C. S. Sastry, and C. K. Mohan, “Content based medical image retrieval using dictionary learning,” *Neurocomputing*, vol. 168, pp. 880–895, 2015.
- [5] M. Lux, “Content based image retrieval with lire,” in *Proceedings of the 19th ACM international conference on Multimedia*. ACM, 2011, pp. 735–738.
- [6] J. R. Smith and S.-F. Chang, “Visalseek: a fully automated content-based image query system,” in *Proceedings of the fourth ACM international conference on Multimedia*. ACM, 1997, pp. 87–98.

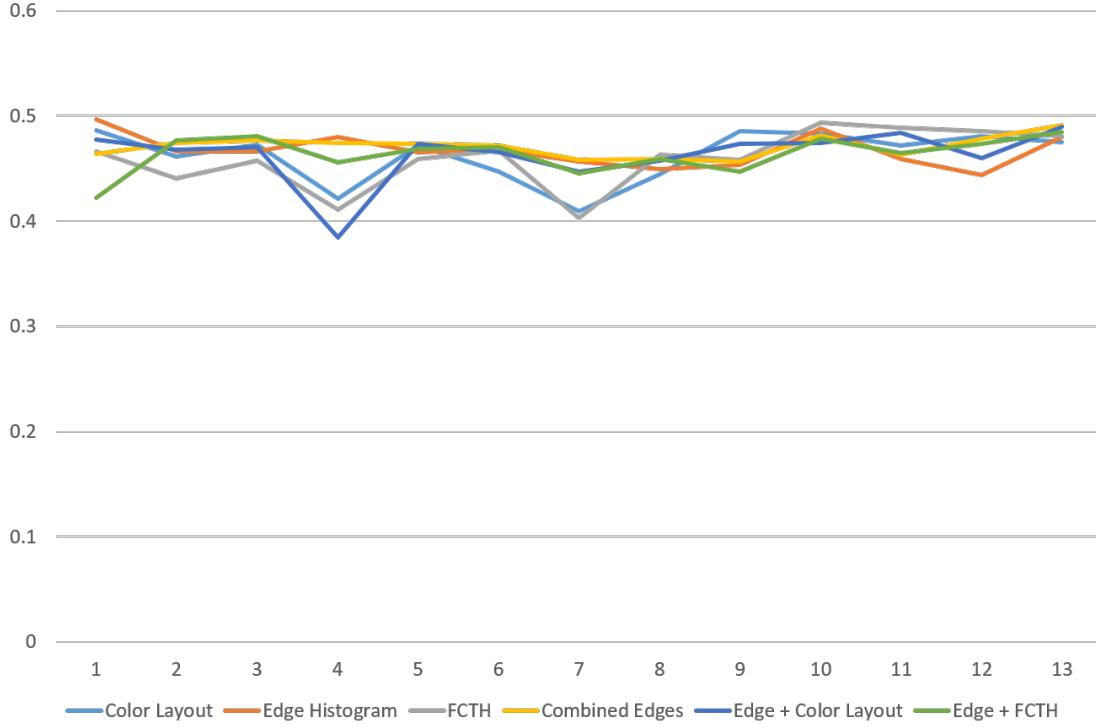


Figure 13: F Measure.

- [7] J. Z. Wang, J. Li, and G. Wiederhold, “Simplicity: Semantics-sensitive integrated matching for picture libraries,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 9, pp. 947–963, 2001.
- [8] A. A. Goshtasby, “Image registration methods,” in *Image Registration*. Springer, 2012, pp. 415–434.
- [9] D. Tian, “A review on image feature extraction and representation techniques,” 2013.
- [10] B. R. V. Dr and Kannan, “Efficient use of mpeg - 7 color layout and edge histogram descriptors in cbir systems,” 2010.
- [11] D. K. Park, Y. S. Jeon, and C. S. Won, “Efficient use of local edge histogram descriptor,” in *Proceedings of the 2000 ACM workshops on Multimedia*. ACM, 2000, pp. 51–54.
- [12] R. Balasubramani and D. V. Kannan, “Efficient use of mpeg-7 color layout and edge histogram descriptors in cbir systems,” *Global Journal of Computer Science and Technology*, vol. 9, no. 4, 2009.
- [13] C. S. Won, “Feature extraction and evaluation using edge histogram descriptor in mpeg-7,” in *Pacific-Rim Conference on Multimedia*. Springer, 2004, pp. 583–590.
- [14] D. A. Lisin, M. A. Mattar, M. B. Blaschko, E. G. Learned-Miller, and M. C. Benfield, “Combining local and global image features for object class recognition,” in *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*. IEEE, 2005, pp. 47–47.

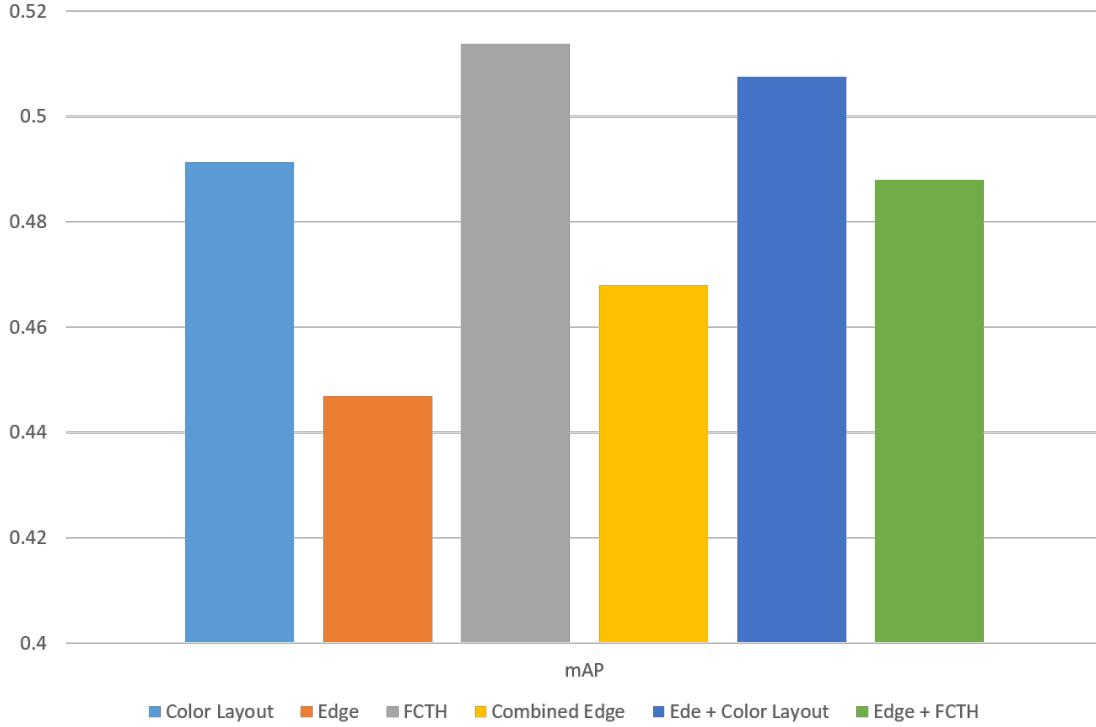


Figure 14: *mAP* chart.

- [15] H.-H. Tsai, B.-M. Chang, P.-S. Lo, and J.-Y. Peng, “On the design of a color image retrieval method based on combined color descriptors and features,” in *Computer Communication and the Internet (ICCCI), 2016 IEEE International Conference on*. IEEE, 2016, pp. 392–395.
- [16] D. ping Tian *et al.*, “A review on image feature extraction and representation techniques,” *International Journal of Multimedia and Ubiquitous Engineering*, vol. 8, no. 4, pp. 385–396, 2013.
- [17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.