A Study of Color Histogram Based Image Retrieval

Conference Paper · May 2009

DOI: 10.1109/ITNG.2009.126 · Source: IEEE Xplore

2 authors:

Rishav Chakravarti
IBM, Yorktown Heights, United States
12 PUBLICATIONS **81** CITATIONS
SEE PROFILE

Xiannong Meng
Bucknell University
54 PUBLICATIONS **254** CITATIONS
SEE PROFILE

2009 Sixth International Conference on Information Technology: New Generations

A Study of Color Histogram Based Image Retrieval

Rishav Chakravarti, Xiannong Meng
Department of Computer Science
Bucknell University
Lewisburg, PA 17837

Abstract

This paper describes a project that implements and tests a simple color histogram based search and retrieve algorithm for images. The study finds the technique to be effective as shown by analysis using the RankPower measurement. The testing also highlights the weaknesses and strengths of the model, concluding that the technique would have to be augmented and modified in order for practical use.

Key Words: color-based image retrieval, content-based image retrieval, Euclidean distance, histogram-intersection, similarity.

1. Introduction

With the increasing popularity of image management tools such as Google's image search and photo album tools such as Google's Picasa project, as well as image search applications in general social networking environment, the quest for practical, effective image search in the web context becomes ever more important. The research community has seen a number of algorithms and tools that facilitate image retrieval. This paper examines one particular algorithm that is based on color-histogram for image retrieval. We implemented the algorithm in Java and compare the effectiveness of the algorithm with other popular image search tools. We conclude that while the algorithm is effective, it needs to be fine tuned before being deployed as a practical tool. We offer some thoughts how this might be done.

The rest of the paper is organized as follows. Section 2 provides a literature survey. The basic algorithm of color-histogram based image search is described in Section 3. The experiment and the data are discussed in Section 4, followed by the results and analysis in Section 5. Finally conclusions and some of our thoughts for future directions are presented in Section 6.

2. Literature reviews

Currently the most popular search engines for images rely on the comparison of metadata or textual tags associated with the images. This methodology relies on human intervention to provide an interpretation of the image content so as to produce tags associated with the image. However, the ever increasing prevalence of large image databases has resulted in the development of algorithms to augment and replace tag based image retrieval with content based image retrieval. These algorithms compare the actual content of the images rather than text which has been annotated previously by a human being.

There are a number of features that can be extracted from an image for comparisons based on their content. Indeed, the Photobook application developed at MIT allows users to perform image retrievals based on user developed models for various information extractions. The user specifies and provides algorithms for extracting certain features of an image in order that they be compared using the platform provided by Photobook. Once the specified feature has been extracted from the image, there are also a number of options for carrying out the actual comparison between images [9]. Generally similarity between two images is based on a computation involving the Euclidean distance or histogram intersection between the respective extracted features of two images. Both these methods involve an intuitive extension of the mathematical definition of a distance between two objects [5]. The three most common characteristics upon which images are compared in content based image retrieval algorithms are color, shape and texture.

2.1 Shape and texture based retrieval

Utilizing shape information for automated image comparisons requires algorithms that perform some form of edge detection or image segmentation. Segmentation refers to the identification of the major color regions in an image [10]. These regions can then be compared from one image to the next. Edge detection tends to be slightly more complicated as it attempts to identify the major contours and edges in a given image. These edges may be then compared based on their direction, with respect to image edges [4]. The advantages of this method include its applicability to black and white images. However, the

978-0-7695-3596-8/09 $25.00 © 2009 IEEE
DOI 10.1109/ITNG.2009.126

performance of the algorithm is not invariant on scale or translation manipulations of images. Information regarding the texture of images can be even harder to extract automatically during retrieval. Generally algorithms rely on the comparison of adjacent pixels to determine the contrast or similarity between pixels [10].

2.2 Color based retrieval

By far the most intuitive information that can be extracted from images for comparison is the color characteristics of an image. This paper attempts to explore and analyze such an algorithm that compares images based on their color content.

A number of algorithms have been developed since the late 1980s that use color information extracted from images for retrievals [10]. A most basic form of color retrieval involves specifying color values that can be searched for in images from a database. Indeed, Google's image organization and editing software, Picasa 3.0, allows users to use an experimental tool to search for certain colors in images. Even this basic method presents challenges in implementation due to the different manners in which computers and human 'see' colors. Computers represent all visible colors with a combination of some set of base color components, generally Red, Green and Blue (RGB). Thus, images perceived by a computer to contain a large component of red may not necessarily appear 'reddish' as perceived by a human eye. Indeed, Picasa's experimental tool suffers from this and returns certain unintuitive results.

Other image retrieval methodologies rely on specifying more precisely the nature of the color that is to be retrieved. These methodologies range from improving upon the nature of the color query to associating spatial conditions to the color information. For example, when searching for country flags, one could specify a search for different regions of color and their positions relative to each other [10].

A more common approach to comparing the color content of a query image to that of database images is that of comparing color histograms. The methodology relies on the fact that images are generally represented as a series of pixel values, each corresponding to a visible color. Color histograms are computed for each image so as to identify relative proportions of pixels within certain values. The idea is that similar images would contain similar proportions of certain colors [4]. The method offers numerous benefits with only a few limitations.

Firstly, image retrieval based on this concept should accurately retrieve images despite the manipulation of orientation, size and position of a certain image. Also, retrieval based on color histograms is fairly efficient and easy in terms of processing content information. A limitation of this algorithm is its inability to easily incorporate the spatial characteristics of the colors in an image. This is particularly true for images stored in Meta or Vector formats that contain more information than simply an array of pixels. Researchers from Stanford University have explored possible solutions by implementing vector quantization strategies that incorporate the distribution of colors in an image [6]. Also, black and white images can necessarily not be compared using an algorithm based solely on color comparisons.

In their paper [4], Jain and Vailaya further analyze this methodology. The image retrieval utilized during their experimentation computes similarity based on the similarity of three different histograms, one for each component of a RGB pixel. The similarity is computed using a Euclidean distance function comparing each 'bin' of the histograms. Retrieval was then carried out by searching for images with the minimum distance to a query image. The experiment carried out testing using an image database consisting of trademarks. The results demonstrate that even with this relatively simple implementation, over 90% of the time an image query is matched accurately with an image in the database. The experiment also goes on to incorporate shape features from the images in the retrieval algorithm.

The paper by Jeong [5] performs a more comprehensive analysis of the histogram approach by studying some of the specific algorithms utilized to carry out the comparisons and retrievals. The manner in which Jeong's paper carries out the actual histogram creation is slightly different from the method proposed in the paper by Jain and Vailaya [4]. Instead of computing histograms separately for each of the red, green or blue components of a pixel, the Jeong experiment attempts to create bins that contain information from all three. Thus, the number of bins is much larger. Additionally, the experiment also tests the retrieval mechanism utilizing HSV (hue, saturation and value) pixels that require different bin sizes.

Three different implementations of the similarity computation were carried out. The first involved a Euclidean distance calculation which computed differences between the number of a certain set of pixels found in one image versus another for each bin in the histogram. The second utilized a histogram intersection method in which colors not present in either one of the images were not used to compare the images. The histogram values were normalized by dividing the number of pixels in each histogram bin by the number of pixel values used in the comparison. This allows image

comparisons to be unaffected by transformations of image size. In addition, ignoring a pixel ranges not present in one of the images can reduce the impact of background color on the result.

Their experiment was carried out on 500 images taken from the SIMPLIcity content based image retrieval database using various implementations of the color histogram. The results from querying these databases with images were analyzed using precision versus recall graphs. Recall signifies the number of relevant images in the database that are retrieved in response to a query. Precision refers to the proportion of the retrieved images that are relevant to the query. Thus, if precision can be increased without sacrificing recall the algorithm is performing well. The experiments analyses showed that the HSV model in conjunction with a histogram intersection method produced the most successful query responses.

2.3 Current content based image retrieval systems

Most existing platforms for retrieving images based on image content implement algorithms that extract a combination of shape, texture and shape features from an image. Then weights are generally assigned to each piece of information extracted from the images and an overall similarity is computed. Images can then be ranked based on this similarity computation. A number of both closed and open source software products can be found [2]. A popular system that has been implemented is IBM's QBIC system [8]. The system has been implemented by the Hermitage Museum website which allows users to search through their digital library of artwork using QBIC's color and layout comparison tools [8].

In addition, there are a number of other solely online application offering services that perform some form of content based image retrieval. Some of these applications were used in the testing process for color histogram technique. SIMPLIcity (also developed as ALIPR) is one such online tool developed by Li and Wang at Penn State [13]. The online tool combines aspects of smart tagging and content based image search [13].

CIRES (Content Based Image REtrieval System) is another such online tool developed by Qasim Iqbal at the University of Texas [3]. The tool extracts information from images to determine structural information, color histograms and textural information. In addition, the website offers an image database that one can search within using images as queries. Also, the tool allows the user to decide the weights assigned to each type of information extracted from the image.

The final such system utilized during our testing is AIRS (Advanced Image Retrieval) developed by the ImageClick Corporation. The system augments simple keyword searching with the beta version of its visual/texture based search engine. Currently, the website allows users to search within thumbnails provided from its image database [1].

3. Algorithm and methodology

The algorithm utilized in our testing of color-histogram approach to content based image retrieval is based on the paper written by Jain and Vailaya [4]. The following is an outline of the method.

1. Read images in database and extract RGB format pixel information from images.
2. Create 48 bin normalized histograms for each of the RGB components of each image read from database. Thus, each image will have 3 histograms associated with it.
3. Read in a query image and extract RGB format pixel information
4. Create histograms for each of the RGB components of the query image.
5. Compute a Euclidean distance by comparing the query image histograms to that of each image in the database.
6. Sort images in database in order of ascending Euclidean distance to query image and return as result.

3.1 Extraction of RGB information

The algorithm was implemented in Java and, thus, the built-in methods provided by Java's image class were utilized to retrieve an array containing pixel values in RGB format. As a result, only image formats compatible with Java's built-in methods were utilized. These consist of the most common formats including, JPEG, BMP, GIF and PNG.

3.2 Histograms

Once extracted, the bits representing each component of the RGB pixel are used to create a histogram. The histogram consists of 48 bins where each bin defines a small range of pixel values. The value stored in each bin is the number of pixels in the image that are within the range. These ranges represent different levels of intensity for each RGB component. The values in each bin are normalized by dividing with the total number of pixels in the image.

3.3 Comparison

Once the histograms have been created, Euclidean distances are calculated. Differences are calculated for each bin by comparing the proportion of pixels of a certain intensity level in each level and then these differences are squared. The squared distances are summed together. The square root of this value is taken. This process is carried out for each histogram after which the average of the three values is taken.

4. Image collection and experiment set-up

All images used in our experiments are available online at: http://www.students.bucknell.edu/rc036/csci378/

The database utilized during program implementation underwent different stages. The preliminary database consisted of images of comic superheroes due to their easily identifiable color schemes. Also, the initial database contained images that were used to the resilience of the image to transformations. Thus, images were taken and put through rotations, flips, resizing, brightening and darkening.

The next stage database was expanded to include five different categories of images collected from various places on the internet as well as my personal collection of images: animals, colors, landscapes, structures, and superheroes. The animals group consisted of different photographs ranging from animals in the savannah to aquarium. Colors were simply monochromic pictures of different colors. Landscapes consisted of photographs of scenery such as desserts, lakes and rocks. Structures consisted of photographs and of pictures taken from the internet. The superhero group of images consisted of the same one as the stage 1 implementation. The total number of images was 50 with a near even distribution in each category.

The third stage database consisted of images retrieved from online Content Based Image Retrieval services found on the internet. Fifty images were taken from 5 different categories found on the CIRES (Content Based Image REtrieval System) project [3]. An additional four images from one category were taken from the AIRS [1]. Thus, in total there were 6 different categories: Lions, Flowers, Orchids, Horses, Aircrafts and Snowboarding. It is important to note that some of these pictures were only taken as thumbnails, thus, the results do not exactly match those found on the online services.

5. Results and analysis

5.1 Stage 1 Results

During the first stage, queries were tested in a limited database against images in the database that were different, duplicates, or transformations. The analysis was simply performed on whether or not the appropriate image was returned as the top most result. The algorithm performed as expected in that rotations, expansions and contractions do not affect the result of the query. Brightening and dimming do have an effect, although when the brightening or darkening is less than 15% the results are still accurate.

Once the initial algorithm validation was completed, groups of images were added to the database. Then each query image was matched visually with groups of images from the database and the database images were ranked according to how similar they were perceived to be to the query. These groups served as the expected relevant results during testing. This method is very subjective to the experimenter as it is difficult to visually compare images based solely on their color similarities. Regardless, the human determine results were compared to those provided by the algorithm using a measurement called *RankPower* [7]. *RankPower* computes a number representing the average rank of relevant results with respect to the total number of relevant results in the database. It sums the positions at which the program returned relevant results and divides with the number of relevant results: $\frac{\sum_{i=1}^{n} s_i}{n^2}$. Clearly, the lower this value, the more likely it is for relevant images to be returned at the top of the results. The optimal value this measure can take is 0.5. There is, however, no upper bound on the number making it a little difficult to interpret. Due to this lack of on upper bound an alternate representation was also utilized for rank power. This representation, proposed by Tang *et. al.*[12], utilizes the formula: $\frac{n(n+1)}{2\sum_{i=1}^{n} s_i}$ where n represents the number of relevant images in the database. This new number is bounded between 0 and 1, where one is the optimal value. Thus, this measure provides a relative measure of the retrieval's effectiveness.

5.2 Stage 2 results

Table 1: RankPower Results Stage 2

Query Image	Returned Rankings	Relevant Images In DB	Rank Power	Alternate Rank Power
1	1	1	1.00	1.00
2	1, 24	2	6.25	0.12
3	1, 2, 5, 7, 10, 17, 20, 23	8	1.33	0.42
4	4, 5, 6, 14, 19	5	1.92	0.31
5	2, 3	2	1.25	0.60
6	1, 5, 7, 14, 15, 16, 21	7	1.61	0.35
7	1, 4, 6, 7, 8, 9, 11, 13	8	0.92	0.61
8	1, 3, 4, 6, 7	5	0.84	0.62
9	1, 2	2	0.75	1.00
10	3, 14, 31	3	5.33	0.13
11	3, 12, 14, 16	4	2.81	0.22
12	2, 5, 6, 18, 21	4	3.25	0.19
Avg		**4.25**	**2.27**	**0.46**

As is apparent there is a fair degree of variation in the rank power of the results. However, the average demonstrates that the average rank with respect to the number of relevant images is fairly low using the traditional rank power methodology. The bounded representation, however, seems to provide mixed results. While the retrieval is not providing poor results, it is not providing consistently accurate results either. This is partially due to the presence of certain results that have extremely high traditional rank power values. Indeed, if two of the results (those with rank power values 5.33 and 6.25) are dropped, the alternate rank power developed by Tang *et. al.* increases past 0.5.

This method of assessing the effectiveness of the retrieval algorithm is highly dependent upon the determination of the 'expected relevant results'. The process of defining these expected results is highly subjective and difficult as one must determine the visual similarity between almost 50 images in the database with the query. These concerns led to the construction of a second stage database where images were taken from previous content based image retrieval platform databases. These images were already placed in groups by their similarity thus allowing the rank power analysis to be carried out much more efficiently.

5.3 Stage 3 results

Table 2: RankPower Results Stage 3

Query Image	Returned Rankings	Relevant Images in DB	Rank Power	Alternate Rank Power
13	1, 2, 4, 7, 12, 16, 19, 20, 22, 23	10	1.26	0.44
14	1, 2, 3, 4, 5, 6, 7, 8, 9, 10	10	0.55	1.00
15	1, 2, 3, 7, 22, 26, 27, 28, 29, 30	10	1.75	0.31
16	1, 2, 3, 4, 5, 6, 7, 12, 19, 21	10	0.80	0.69
17	1, 2, 3, 4, 5, 6, 7, 14, 28, 35	10	1.05	0.52
18	1, 2, 3, 4	4	0.63	1.00
19	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 16	12	0.59	0.92
Avg		**9.4**	**0.95**	**0.70**

The stage 3 results demonstrate a much lower level of variation in rank power than in the previous test results. The results are also much improved in that the average Rank Power is less than 1 and the alternate rank power measure is much closer to 1 than in the previous test runs. Also, if one simply notes the ranks of the relevant images as determined by the algorithm, it is apparent that the majority of the relevant images are returned in the top ten results.

Query 13 and 15 resulted in a much lower alternate rank power. This lack of performance may be attributed to the fact that when collecting images from the online service, only thumbnails were downloaded. Thus, the comparisons in these thumbnails may not correspond perfectly to comparisons between the original images. Despite these outliers, however, the performance of the algorithm is effective even with a simple and highly intuitive implementation of color histogram based image retrieval.

6. Conclusions

The study's implementation of a color histogram based image retrieval system identified numerous strengths in the algorithm performance as an image retrieval system. Firstly, the algorithm is relatively easy and intuitive to implement from a coding stand point. In addition, the method allows retrieval of images that have been transformed in terms of their size as well as translated through rotations and flips. During testing, the basic algorithm produced some good results in that it was able to retrieve many relevant images. This was particularly after other computerized online services were used to identify which images in the database were relevant.

The main weakness of this method was also, however, identified during the testing process in that this particular study's implementation of color histograms does not necessarily allow the relevant images as seen by the algorithm to be the same as the relevant images as seen by a human. The algorithm attempts to analyze color information by studying each RGB component histogram separately. Thus, the color analysis does not necessarily relate to similarity of colors as identified visually by the human eye. It is possible that this shortcoming may be addressed using the implementation by Jeong [5] in which the RGB pixel is treated as a single component forming a 3d histogram with a much larger number of bins. Thus, colors are identified by the algorithm in a manner more akin to a visual identification of the colors.

Lastly, as demonstrated by the various implementations of content based image retrieval systems, color histogram based comparisons can be easily combined using weights with techniques that extract other information from the image. Thus, this easy to implement technique of comparing images is an effective tool for accurate content based image retrieval.

7. References

[1] *AIRS - Advanced Image Retrieval Service.* http://www.imageclick.com/airs/sub/aboutAIRs.html (accessed October 2008).

[2] *Content-based image retrieval - Wikipedia.* 5 November 2008. http://en.wikipedia.org/wiki/CBIR (accessed November 8, 2008).

[3] Iqbal, Qasim. *CIRES: Content Based Image REtrieval System.* August 2007. http://amazon.ece.utexas.edu/~qasim/research.htm (accessed November 2008).

[4] Jain, Anil K., and Aditya Vailaya. *Image Retrieval Using Color And Shape.* Great Britain: Elsevier Science Ltd, 1995.

[5] Jeong, Sangoh. *Histogram-Based Color Image Retrieval.* Psych221/EE362 Project Report, Stanford University, 2001.

[6] Jeong, Sangoh, Chee Sun Won, and Robert M Gray. *Image Retrieval using color histograms generated by Gauss mixture vector quantization.* Standford University and Dongguk University, Seoul: El Sevier, 2003.

[7] Meng, Xiannong. "A Comparative Study of Performance Measures for Information Retrieval Systems." *Third International Conference on Information Technology: New Generations (ITNG'06)*. 2006. 578-579.

[8] *QBIC - IBM's Query By Image Content* . http://wwwqbic.almaden.ibm.com/ (accessed November 2008).

[9] Sclaroff, S, A. Pentland, and R Picard. "Photobook: Tools for Content-Based Manipulation of Image Databases ." *SPIE Storage and Retrieval of Image & Video Databases II.* February 1994. http://vismod.media.mit.edu/vismod/demos/photobook/ (accessed November 5, 2008).

[10] Smith, John R, and Shi-Fu Chang. "Tools and Techniques for Color Retrieval." *Electronic Imaging: Science and Technology-Storage~ Retrieval for Image and Video Database IV*. San Jose: IS&T/SPIE, 1996. 1-12.

[11] Smith, John R., and Shih-Fu Chang. *Automated Image Retrieval Using Color and Texture.* Columbia University Technical Report , New York: Columbia University, 1995.

[12] Tang, J., Z. Chen, A.W. Fu, and D.W. Cheung, "Capabilities of Outlier Detection Schemes in Large Databases, Framework and Methodologies", *Knowledge and Information Systems*, 11(1): 45-84.

[13] Wang, James Z. *ALIPR/SIMPLIcity/ALIP:Object Concept Recognition/Content Based Image Retrieval/Annotation.* 1995. http://wang.ist.psu.edu/IMAGE/ (accessed September 2008).