International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 5 Issue: 12
ISSN: 2321-8169
62 – 66

Image based Search Engine for Online Shopping

Sumit Mene, Arshey Dhangekar, Abhijeet Gurav, Harshal R. Randad, Rupali Tornekar and Prof. (Dr.) Vijay Gaikwad

***Abstract*—** This paper presents a method based on principle of content based image retrieval for online shopping based on color, HSV aiming at efficient retrieval of images from the large database for online shopping specially for fashion shopping. Here, HSV modeling is used for creating our application with a huge image database, which compares image source with the destination components. In this paper, a technique is used for finding items by image search, which is convenient for buyers in order to allow them to see the products. The reason for using image search for items instead of text searches is that item searching by keywords or text has some issues such as errors in search items, expansion in search and inaccuracy in search results. This paper is an attempt to help users to choose the best options among many products and decide exactly what they want with the fast and easy search by image retrieval. This technology is providing a new search mode, searching by image, which will help buyers for finding the same or similar image retrieval in the database store. The image searching results have been made customers buy products quickly. This feature is implemented to identify and extract features of prominent object present in an image. Using different statistical measures, similarity measures are calculated and evaluated. Image retrieval based on color is a trivial task. Identifying objects of prominence in an image and retrieving image with similar features is a complex task. Finding prominent object in an image is difficult in a background image and is the challenging task in retrieving images. We calculated and change the region of interest in order to increase speed of operation as well as accuracy by masking the background content. The Implementation results proved that proposed method is effective in recalling the images of same pattern or texture.

***Index Terms*—** *Image color analysis, Web Image search, Histograms, Image retrieval, Feature extraction.*

*****

I. INTRODUCTION

Development of the Internet technology has eased the increase of in-home shopping. The Internet has made a powerful impact on marketing and created another type of retail transaction called e-commerce for online shopping.

The Electronic commerce is a term used for selling and buying on the Internet.In the last few years the business volume of online shopping is growing up and it also seems that this trend will be maintained in the next future. In particular, the clothing sector has a considerable margin of growth and therefore the number of potential buyers is very high. Nowadays an increasing number of people own mobile phones with Internet access and multimedia capabilities such as built-in camera. Modern search engines offer the state of the art in the text search of content as web pages, databases of documents etc. Regarding querying by images, the scenario is less mature, although the information contained within an image is usually much greater than the information which can be retrieved from the text. The major online search engines offer image search systems even if the greatest practical difficulty lies in sending an image of the desired object.

Obviously, search based on images is required. Nowadays some search engines have provided image search, such as image Baidu, but it can't retrieve the commodities on shopping websites. We have noted that commodities of online shopping websites have a character that there are about five images and a paragraph of text to describe a commodity. In this paper, we extract the visual features from these images without their background, and apply the features to a two-stage image search proposed, which utilizes basic features consisting of color and textural features to filter mismatching images in first stage, and further uses SIFT features for accurate search in second stage. This project is deploying a technique that uses low-level features which are extracted automatically from images and then are used for indexing and retrieval. It combines color and texture information in a histogram, and with e-commerce for online shopping. The idea of this work is using image processing to help in shopping. Our application permits the buyer to upload a photo and then return comparable products using image retrieval systems

II. RELATED WORK

In this field, researchers have done some pioneering work. Davis proposed a multi-modal shopping assistant which provides users with a service capable of reducing the time spent on grocery shopping and the stress that occurs during this activity. It helps consumers to find commodities quickly at shopping mall in reality. Anil proposed an algorithm to search commodities with trademark. Some companies have developed some image retrieval systems like QBIC of IBM Virage of company Virage and MARS of Illinois University of United States. All of them are great prototype systems for image retrieval.

III. PROPOSED METHODOLOGY

There are three types of search engines for an image search :

1. Search by Meta-data –
   Where a user performs a search on a search by meta-data system i.e. user provides a query in text format, just like in conventional text search engines, and then images that have similar labels or annotations are returned by the system during a search.

2. Search by Example –
   Search by example, depends entirely on the context of the image - no keywords are given by user. The image is scrutinized, quantified, and features are stored so that similar images are returned as results.

ISSN: 2321-8169
62 – 66

3. Hybrid –
   Hybrid is combination of both search by example and Metadata. By hybrid search method one could construct an image search engine that could take both contextual hints along with a Search by Example/label method.

A: Principle –

To building an image search engine we will have to ***index our dataset***. Indexing a dataset is the process of quantifying our dataset by utilizing an ***image descriptor*** to extract ***features*** from each image.

For example:

- The mean and standard deviation of each Red, Green, and Blue channel, respectively,
- The statistical moments of the image to characterize shape.
- The gradient magnitude and orientation to describe both shape and texture.

Features, on the other hand, are the output of an ***image descriptor***. When user puts a images from dataset into an color descriptor ***features*** will be extracted from images. ***Features*** (or***feature vectors***) are piece of information i.e. just a list of numbers used to abstractly represent context of images.

Feature vectors can be used to compare similarity of images by using a ***distance metric*** or ***similarity function***. Distance metrics and similarity functions take two feature vectors as inputs and then gives output a number that represents how “similar” the two feature vectors are.

B: Four Algorithms of CBIR System –

Content-Based Image Retrieval System can be split down into 4 distinct algorithms:

1. Defining your image descriptor
2. Indexing your dataset
3. Defining your similarity metric
4. Searching

The figure below details Algorithm 1 and 2:

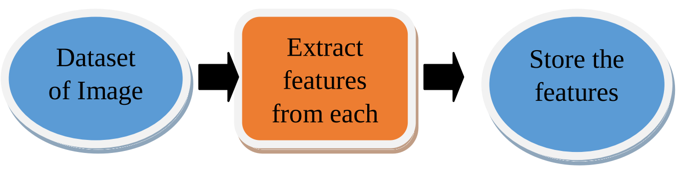


Fig.1.1A flowchart representing the process of extracting features from each image in the dataset.

We start by taking our dataset of images, extracting features from each image, and then storing these features in a database.

We can then move on to performing a search (Algorithm 3 and 4):

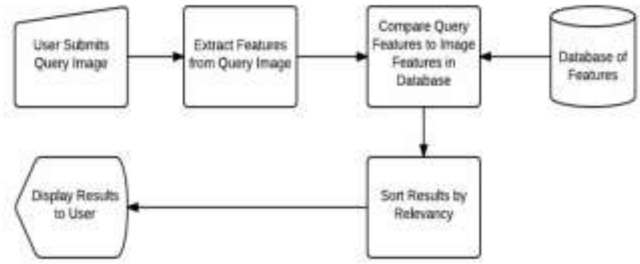


Fig 1.2 Performing a search on a CBIR system. A user submits a query, the query image is described, and the query features are compared to existing features in the database, and results are sorted by relevancy and then presented to the user.

A user must upload a query image to search engine. Then features are extracted from query features. Then these features of a query image are then compared to the features of the images we already indexed in our dataset. Then, the outputs are then ranked by relevancy and displayed as result.

Inorder to construct this system, we’ll be using effective image descriptor method named as **color histogram**.

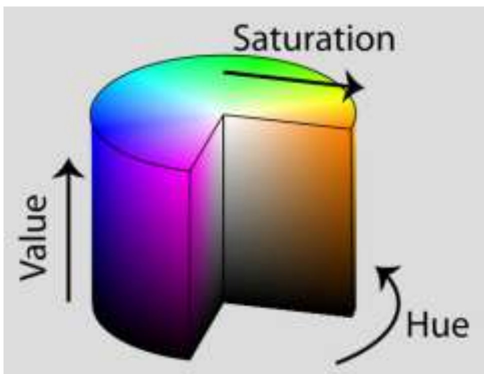
By utilizing a color histogram as image descriptor, we’ll be relying on the *color dissemination* of the image. Because of this, we have to make some important assumptions regarding image search engine:

Assumption: Images that have similar color disseminations will be considered relevant to each other. Even if images have dramatically different contents, they will still be considered “similar” provided that their color distributions are similar as well.

Step 1: Defining Image Descriptor –

Instead of using a conventional approach of color histogram, we are going to apply a few methods to make it a little more robust.

Our image descriptor have three dimensional color histogram in the HSV color space (Hue, Saturation, and Value) as shown in figure:



ISSN: 2321-8169
62 – 66

Fig 2.1 HSV model

We have selected a color space. Also we need to define the number of bins for our histogram. Histograms shows graphical distribution of pixel intensities in an image. Our histogram will estimate the probability density of the underlying function, or in this case, the probability *P* of a pixel color *C* occurring in our image.

If we select *too few bins*, then histogram will have less components and unable to disambiguate between images with substantially different color distributions. Likewise, if you use *too many bins* your histogram will have many components and images with very similar contents may be regarded and "not similar" when in reality they are. Smaller the dataset, the less bins use and vice versa.

Instead of computing a three dimensional HSV color histogram for the ***entire*** image, it is better to compute a 3D HSV color histogram for different ***regions*** of the image. ***Global***-histograms allows us to simulate locality in a color distribution over ***regions***-based histograms . For example, take a look at this image below:

Fig 3.1 Division of region of interest

With a global histogram we would be unable to determine *where* in the image the "blue" occurs and where the "brown" sand occurs. Instead, we would just know that there exists some percentage of blue and some percentage of brown.

For our image descriptor, we are going to divide our image into five different regions: (1) the top-left corner, (2) the top-right corner, (3) the bottom-right corner, (4) the bottom-left corner, and finally (5) the center of the image.

Then, we'll need to construct an ellipse to represent the center region of the image by using masking concept because weneed the mask to instruct the OpenCV histogram function where to extract the color histogram from.

Remember, our goal is to describe each of these segments *individually*. The most efficient way of representing each of these segments is to use a *mask*. Only *(x, y)* coordinates in the image that has a corresponding *(x, y)* location in the mask with a white (255) pixel value will be included in the histogram calculation. If the pixel value for an *(x, y)* coordinate in the mask has a value of black (0), it will be ignored as shown below.

Fig 3.2 Masking of other data except region of interest

This means that if we computed a color histogram for two identical images, except that one was 50% larger than the other, our color histograms would be (nearly) identical.

Step 2: Extracting Features from Our Dataset –

The process of extracting features and storing them on persistent storage is commonly called "indexing".

For each of the images we'll extract an imageID, which is simply the filename of the image. For this example search engine, we'll assume that all filenames are unique, but we could just as easily generate a UUID for each image.The describe methodof our Color Descriptor returns a list of floating point values used to represent and quantify our image.

This list of numbers, or *feature vector* contains representations for each of the 5 image regions we described in Step 1. Each section is represented by a histogram with $8 \times 12 \times 3 = 288$ entries. Given 5 entries, our overall feature vector is $5 \times 288 = 1440$ dimensionality. Thus each image is quantified and represented using 1,440 numbers.

Step 3: The Searcher –

Now that we've extracted features from our dataset, we need a method to *compare* these features for similarity. That's where Step 3 comes in — we are now ready to create a class that will define the actual similarity metric between two images.

This method will take two parameters, the query Features extracted from the query image (i.e. the image we'll be submitting to our CBIR system and asking for similar images to), and limit which is the maximum number of results to return.

Lastly, all we have to do is sort the results dictionary according to the similarity value in ascending order.

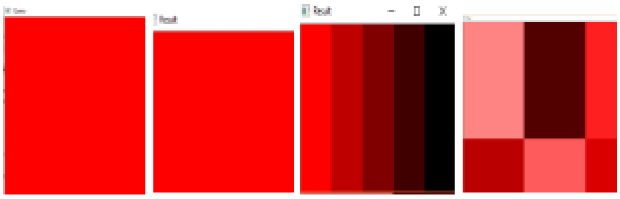
Step 4: Performing a Search –

We'll also need a --query, which is the path to our query image. This image will be compared to each image in our index. The goal will be to find images in the index that are similar to our query image.

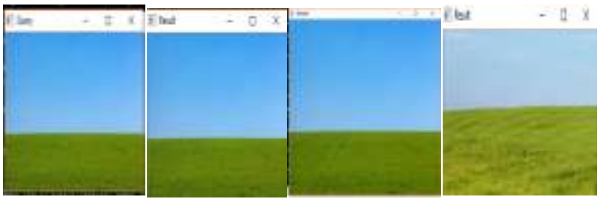
RESULT:

1 Query Result1 Result2 Result3

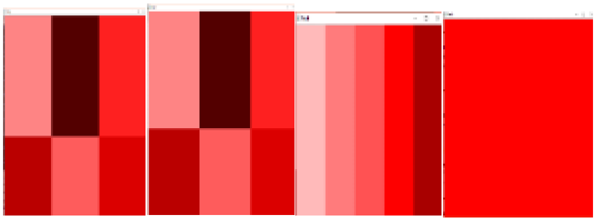
ISSN: 2321-8169
62 – 66



2 Query Result1 Result2 Result3



3 Query Result1 Result2 Result3



4 Query Result1 Result2 Result3

IV. RESULTS

Table1: Reduction in time

Sr no.	No of images	Previous interest of region	Region of interest
1	50	1.3	0.9
2	100	1.89	1.35
3	150	2.65	1.69
4	200	2.74	2.15
5	250	3.2	2.60
6	300	4.07	3.18
7	350	4.41	4.49
8	400	5.25	4.87
6	450	5.85	5.03
7	500	6.17	5.79

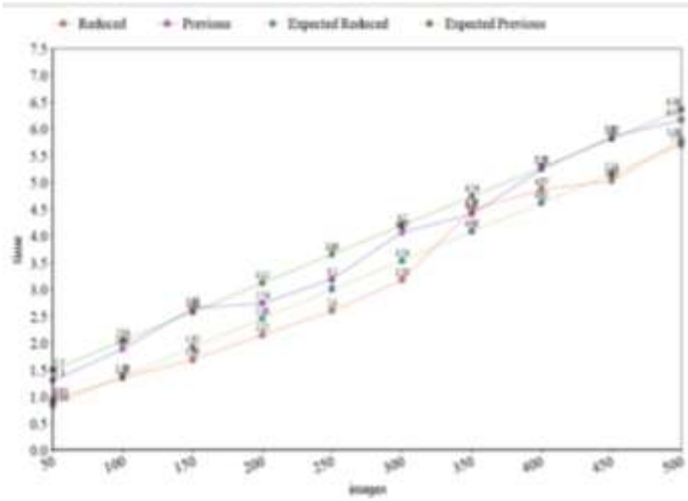


Fig : graph of time reduction

Table2: Accuracy

Srno.	Accuracy
1.(Single color)	100%
2.(Multicolor)	92%
3.(Tricolor)	86%
2.(Bicolor)	80%

V. RESULT ANALYSIS

In this paper we experimented the content based image retrieval on different types of images such as single color, bicolor, tricolor and multicolor. Also we tried same images with different orientation. Accuracy is found to be greater with single color images and least with bicolor images. The time complexity is found to be reduced after reducing the region of interest.

VI. CONCLUSION

In this paper we proposed a methodology for image search engine for online shopping by optimising the region of interest

This method helps user in finding the object/material available on online shoppingsites.And also showsthings/objects/materials which are closely related to an query image.By reducing the region of interest accuracy of image search is increased and also the speed of operation is reduced

VII. REFERENCES

[1] Lee J-E, Jin R, Jain AK, Tong W. Image retrieval in forensics: tattoo image database application. IEEE Multimedia. 2012;19(1):40–49.

[2] X.Wang,X.tang et.tl in IEEE transaction.patternananlysis and machine intelligence,{vol.36,no 4,pp.810-823}, paper

ISSN: 2321-8169
62 – 66

title {Web Image reranking using query specific semanic signatures}, in 2014.

[3] A.W.M Smwulders et.tl "Content-based image retrieval" IEEE tran.pattern analysis and machine Intelligence,Vol.22,no.12,pp.1349- 1380,dec 2000. 4.

[4] "Survey on web-scale Image search and re-ranking with semantic signatures" in IJIRCCE,Volume 3(3),PP 1864-1867,March 2015 by DarshnaC.Chaudhari and Prof.pritiSubramanium.

[5] Datta R, Joshi D, Li J, Wang JZ. Image retrieval: ideas, influences, and trends of the new age. ACM Computing Surveys. 2008;40(2, article 5)

[6] Li Y. Research on content-based image retrieval [Ph.D. thesis] JI Lin University; June 2009.

[7] Niblack W, Barber R, Equitz W, et al. QBIC project: querying images by content, using color, texture, and shape. Storage and Retrieval for Image and Video Databases; February 1993; San Jose, Calif, USA. pp. 173–187.

[8] Hsu, Esther, C.Paz, and S.Shen, "Clothing image retrieval for smarter shopping, " EE368, Department of Electrical and Engineering, Stanford University (2011).

[9] Smith JR, Chang S-F. VisualSEEk: a fully automated content-based image query system. Proceedings of the 4th ACM International Multimedia Conference; November 1996; Boston, Mass, USA. pp. 87–98. 10.

[10] Bach JR, Fuller C, Gupta A, et al. Virage image search engine: an open framework for image management. Storage and Retrieval for Still Image and Video Databases IV; February 1996; San Jose, Calif, USA. pp. 76–87. 11.

[11] J.Cai,Z.Zha,W.Zhou,and Q.Tian,"Attritube-Assisted reranking for web Image Retrievel",in Proc 20th ACM Intconf.Multimedia 2012.

[12] Proc.ACM Multimedia 2008 for Interactive online image search reranking.

[13] Lin X, Gokturk B, Sumengen B, Vu D. Visual search engine for product images. Multimedia Content Access: Algorithms and Systems II; January 2008.

[14] Taotaosou, 2011, http://www.taotaosou.com/ 15.

[15] Schmid C, Mohr R, Bauckhage C. Evaluation of interest point detectors. International Journal of Computer Vision. 2000;37(2):151–172.

[16] Muller H, Muller W, Squire DM, Marchand-Maillet S, Pun T. Performance evaluation in content-based image retrieval: overview and proposals. Pattern Recognition Letters. 2001;22(5):593–601.