A Report on paper : Comparison of Color Features for Image Retrieval

Abdullah Al Mamun (1305003)
S.Mahmudul Hasan (1305043)

Abstract

This is a report on a paper titled "Comparison of Color Features for Image Retrieval". The paper is written by S.R. Kodituwakku & S.Selvarajah. In this report, we described the content of the paper in our own language. The paper deals with image retrieval problem. We described the importance of the topic covered in the paper, the methods used, the evaluation metrics and the evaluation results given in this report.

Introduction

Content Based Image Retrieval is an application of **computer vision techniques** to a problem known as **image retrieval problem**. Image retrieval problem deals with the **searching and retrieval of digital images** from large databases.

The phrase **content based** means that the process analyzes the **contents** of the image and **not** the meta-data such as keywords, tags, associated descriptions etc. The content in this context may refer to colors, shapes, textures, or any other informations that can be derived from the image.

In this paper, images are represented by **color descriptors**. There are various color descriptors, but in this paper **color histogram, color moments and color coherent vector** are used as color descriptors. The efficiency of these color descriptors to represent visual features of images are compared in the paper.

Motivation

Content based image retrieval is important because it lacks the drawbacks meta-data based image retrieval has. Searches that rely purely on meta-data are not independent. Rather they depend on **annotation quality and completeness**. Also to manually annotating images by entering meta-data or keywords in a large database is time consuming and can have human errors like not capturing

the keywords desired to describe a particular image.

Content based image retrieval has applications in various important sectors including architectural and engineering design, art collections, crime prevention, geographical information and remote sensing systems, medical diagnosis, military, photograph archive, retail catalogs, nudity-detection filters, face recognition etc. **Commercial systems** that have been developed based on this include IBMs QBIC, Virages VIR Image Engine, Excaliburs Image RetrievalWare etc. Also various **experimental systems** have been and are being developed based on content based image retrieval.

Core Concepts

CBIR systems manages images by automatically **indexing** them by summarizing their visual features [6–9, 11–15]. A feature can capture some characteristics of images either in a global manner or in a local manner [13]. CBIR systems uses **color, textures and shapes** as features. Images features are extracted by mapping the image pixels to the *feature space*. Feature spaces are generally represented as feature vectors. Once the features are represented as feature vectors, they can be used to measure similarity between different images. Query image and database's images are represented as feature vector at retrieval stage. [12]

The paper deals with performance of different color descriptors to describe images and based on the description of images the retrieval rate of certain images from the database. For that purpose a fixed color model is selected. The **RBG color model** is used in this paper.
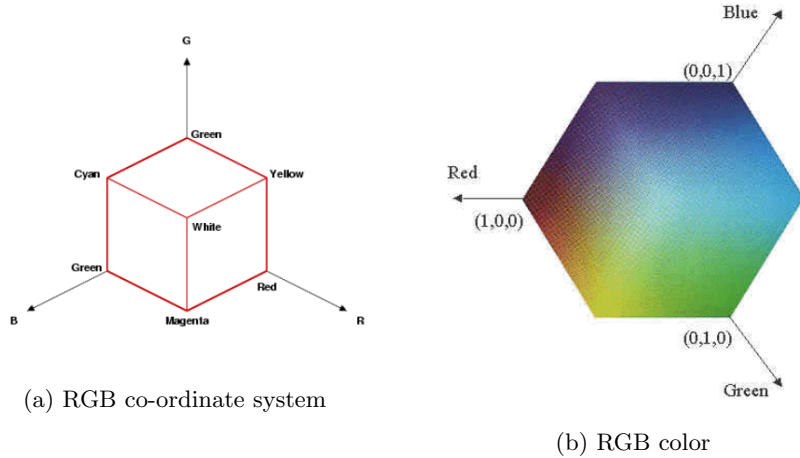

(a) RGB co-ordinate system

(b) RGB color

Figure 1: RGB color space

2

Color is a very important feature of an image. It doesnt change whether an image is **scaled, translated or rotated**. The key components of color feature extraction are **color space, color quantification and similarity measurement**. A color space is a specific organization of color which allows for reproducible representation of color when combined with physical device profiling.

Color Moments

Color moments are measures that can be used differentiate images based on their *features of color*. These moments provide a measurement for color similarity between images. These values of similarity can be compared to the values of images in a database for image retrieval. Color moments assume that the *distribution of color* in an image can be interpreted as a *probability distribution*. As probability distributions are characterized by a number of unique moments, the color in an image follows a certain probability distribution and the moments of that distribution can be used as features to identify the image based on color.

Mean, variance and standard deviation are used as color moments [10]. A color is defined by *three or more* values. Generally we use **hue, saturation and brightness** to define a color. In our RGB color space, there are red, blue and green color. For each of the color there are three color moments and so in *total nine* color moments are computed.

Let a image be of size $n \times m$ and X_{ij} be the pixel at i^{th} row and j^{th} column, then,

$$mean = \sum_{i=1}^{n} \sum_{j=1}^{m} X_{ij}/(m \times n) \tag{1}$$

$$variance = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} (X_{ij} - mean)^2 \tag{2}$$

$$stddev = \sqrt{variance} \tag{3}$$

Color Histogram

A color histogram is a representation of the **distribution** of colors in an image [6]. It represents the number of pixels that have colors in each of a fixed list of color ranges that span the images color space. The color histogram of a certain image can be built for any kind of color space but in this paper, histogram is built for three-dimensional RGB color space. Color histogram is a *statistic* that can be viewed as an *approximation* of an underlying continuous distribution of color values.It is a graphical representation of the number of pixels in an image. It is a **bar graph** whose X-axis represents **tonal scale** and Y-axis represents

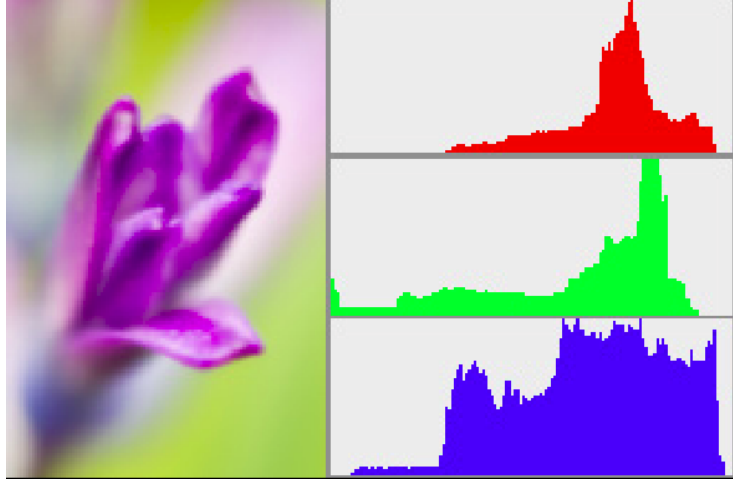the **number of pixels** in an image in a certain area of the tonal scale.

Figure 2: Histogram of an image

The color histogram for an image is constructed by **counting the number of pixels of each color**. There can be a number of elements in the histogram of an image. This number depends on the **number of bits** in each pixel of an image. If an image has a pixel depth on n bit, then the pixel values of the image will be between 0 and $2^n - 1$. In this case, the histogram will have 2^n elements.

Let, color histogram be h. Then it is defined by,

$$h_{A,B,C}(a, b, c) = N * Prob(A = a, B = b, C = c) \quad (4)$$

where A, B and C represent the three color channels (R, G, B) and N is the number of pixels in the image.

Computationally, color histogram is made by **discretizing the colors** within an image and counting the number of pixels of each color. If the data sets are too large, color moments can be computed by using color histograms.

$$mean = \sum_{i=0}^{255} i * h(i) / \sum_{i=0}^{255} h(i) \quad (5)$$

4

$$variance = \sqrt{\frac{\sum_{i=0}^{255} h(i) * (i - mean)^2}{\sum_{i=0}^{255} h(i)}} \qquad (6)$$

Here $h =$ histogram of the image

In this experiment two techniques of histogram based image retrieval are used. The first one is **Global Color Histogram (GCH)** and the next one is **Local Color Histogram (LCH)**. In GCH, the whole image is represented by a single histogram. In LCH, the image is divided into blocks of 8×8 and for each of the blocks a histogram is computed.

In both cases, the histograms of query image and databases images are compared and relevant images are retrieved. Let h and g represent two color histograms. Then the *euclidean distance* between the color histograms h and g can be computed as:

$$d^2(h, g) = \sum_{A} \sum_{B} \sum_{C} (h(a, b, c) - g(a, b, c))^2 \qquad (7)$$

In this case the comparison deals only with identical bins in the respective histograms.

Color Coherent Vector

Color coherent vector is similar to color histogram. Color coherence is the degree to which pixels of some color are members of large similarly colored regions. We refer this regions as coherent regions. Coherent regions are of significant importance in characterizing images. Depending on the size of coherent region, coherent pixels are selected and using those color coherence vector for the images are computed. These CCVs are used to compare the differences and similarities between different images.

Color coherent vectors takes the spatial informations of images into count. Color histogram doesnt take this into count. As a result two or more images can have the same color histogram.

  The images shown in Figure 3 have similar color histogram although their appearances are clearly different. The color red appears in both images in approximately the same quantities. In left image the red pixels (from the flowers) are widely scattered while in the right image the red pixels (from the golfers shirt) form a single coherent region.

Our coherence measure classifies pixels as either coherent or incoherent. Coherent pixels are a part of sizable contiguous region while incoherent pixels are not. A
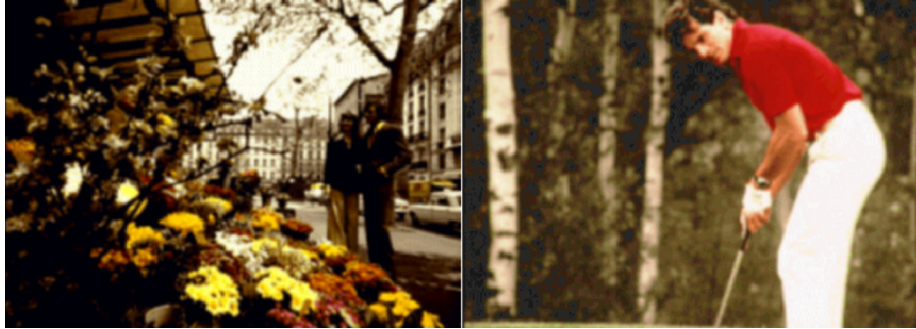
Figure 3: Two images with similar color histogram

color coherent vector represents this classification for each color in the image. CCVs prevent coherent pixels in one image from matching incoherent pixels in another. This allows fine distinctions that cant be achieved with color histograms.

To classify the pixels within a given color bucket as either coherent or incoherent, some steps are taken. A coherent pixel is part of a **large group of pixel of the same color**, while an incoherent pixel is not. To determine coherent pixels pixel groups are calculated. We determine pixel groups by computing connected components.

A connected component C is a maximal set of points such that for any two points $p, p' \in C$, there is a path between p and p'. We only compute connected components within a given discretized color bucket.

Figure 4: Connected components

The connected component can be computed in linear time. When it is completed, each pixel will belong to exactly one connected component. We classify each

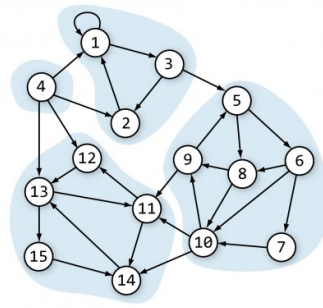pixel as coherent or incoherent based on the size of its connected component. A pixel is considered coherent if its connected components size exceeds a fixed value T. All other pixels are considered incoherent.

Things To Evaluate

There are eight similarity measurements which can be used to measure the similarities between images [11]. But in this paper, Sum of Squared Differences (SSD) and Sum of Absolute Differences (SAD) are used. These two methods are reported to have the most effectiveness and efficiency [14].

$$SSD(f_q, f_t) = \sum_{i=0}^{n-1} (f_q[i] - f_t[i])^2 \tag{8}$$

$$SAD(f_q, f_t) = \sum_{i=0}^{n-1} (f_q[i] - f_t[i]) \tag{9}$$

Here,
f_q = query feature vector
f_t = database feature vectors
n = the number of features in each vector

We used a general purpose database containing 14500 images drawn from different categories to evaluate the effectiveness and efficiency of the selected color descriptors [1–5]. In the database, the images have dimension of 384X256. We calculated the color features of different images in the database and stored them in a database.

The color features are evaluated by means of two ratios: precision and recall. Precision is the ratio of relevant retrieved images to the total retrieved images. Recall is the ratio of relevant retrieved images to the total relevant images in the database.

$$Precision = \frac{Number\ of\ relevant\ retrived\ images}{Number\ of\ total\ retrived\ images}$$

$$Recall = \frac{Number\ of\ relevant\ retrived\ images}{Number\ of\ total\ relevant\ images\ in\ the\ database}$$

Evaluation Process

Normal images are generally of true color. These images are 24-bits. So there can a total of 2^{24} or 16777216 color variations. The images both in the database and in the query lists, are quantized to reduce the number of colors from 2^{24} to 256. So the images will be of 8-bit colors.

(a) True Color Picture   (b) 8-bit Color Picture

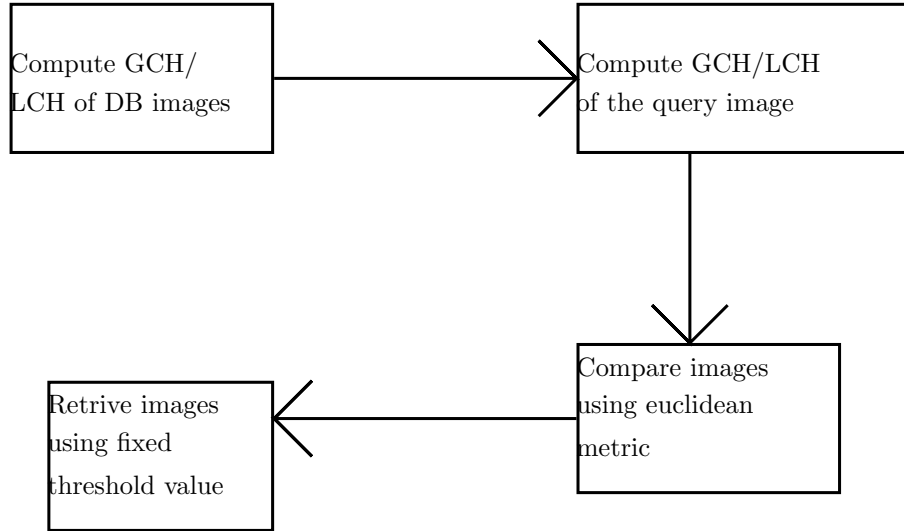Figure 5: True color & 8-bit representation of an image

```
┌─────────────────┐                    ┌─────────────────┐
│ Compute GCH/    │───────────────────▶│ Compute GCH/LCH │
│ LCH of DB images│                    │ of the query    │
│                 │                    │ image           │
└─────────────────┘                    └─────────────────┘
                                                │
                                                │
                                                ▼
┌─────────────────┐                    ┌─────────────────┐
│ Retrive images  │                    │ Compare images  │
│ using fixed     │◀───────────────────│ using euclidean │
│ threshold value │                    │ metric          │
└─────────────────┘                    └─────────────────┘
```

Figure 6: Image retrieval by using histogram

Histogram Based Image Retrieval

In diagram 6 the process of using histogram to retrieve image has
been shown. First the global/local color histograms for each of
the database images are calculated and stored. Then the query
image is processed and global/local histogram for the query image
is computed. Then using Euclidean distance metrics like Equation
7, the similarities of between the query image and the database
images are calculated. A fixed threshold value is taken into account
to identify the relevant images.

Color Moment Based Image Retrieval

Color moments based image retrival has been briefed in diagram
7. The visual features of an image is extracted using the equations
1,2 and 3. Color moments are also computed using the histogram
computed in the previous process (using equations 5 and 6). These
are calculated for databases images and stored. Then query images
mean, standard deviation are calculated. After that these are compared
with the same types in the database image. A threshold value is
fixed which depends on the difference between mean and standard

```
┌─────────────────────────┐              ┌─────────────────────────┐
│ Extract the visual      │              │ Calculate related values│
│ features                │─────────────▶│ for                     │
│ using the equations     │              │ DB's images             │
└─────────────────────────┘              └─────────────────────────┘
                                                      │
                                                      ▼
┌─────────────────────────┐              ┌─────────────────────────┐
│ Compare the values of   │              │ Calculate related values│
│ query                   │◀─────────────│ for                     │
│ images with those of DB's│             │ query images            │
│ images                  │              │                         │
└─────────────────────────┘              └─────────────────────────┘
         │
         ▼
      ┌─────────────────────────────────┐
      │ Rank and retrive the relevant   │
      │ images                          │
      └─────────────────────────────────┘
```
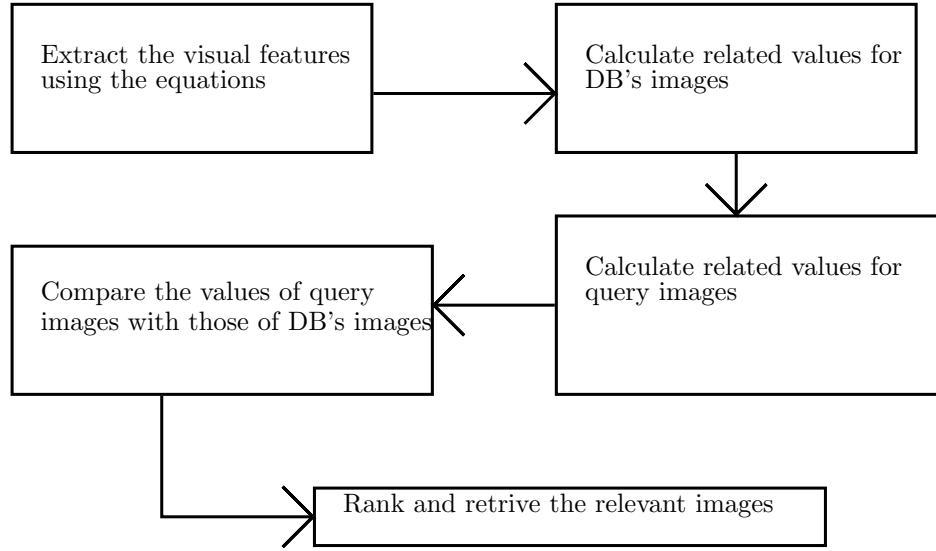
Figure 7: Image retrieval using color moments

deviation of query image and database images. The relevant images
are ranked using it.

Color Coherent Vector Based Image Retrieval

The image retrieval process using CCV is shown in the diagram 8.
First for all the images in the database, the pixels in the images
are categorized as coherent or incoherent by computing connected
component. Then we use a color coherent vector to represent this
classification for each of the colors. We replace the pixel values
with the average value in a small neighborhood. As a result the
image is slightly blurred. This can be done using gaussian filter.
We descretize the color space so that only R,G and B colors are
present in the image. Then color coherent vector is computed for the
database images and query images. We retrieve the relevant images
by comparing the similarity between query vector and databases
image vectors.

We used combination of different color features and retrieving images
based on it.

```
┌─────────────────────────┐        ┌─────────────────────────────┐
│ Categorize images' pixels│───────▶│ Use CCV to represent the    │
│ as coherent or incoherent│        │ coherent and incoherent pixels│
│                         │        │ for each color              │
└─────────────────────────┘        └─────────────────────────────┘
                                                 │
                                                 ▼
┌─────────────────────────┐        ┌─────────────────────────────┐
│ Compare the CCVs of     │◀───────│ Compute DB's images CCV     │
│ DB images and query images│       │ and query image's CCV       │
└─────────────────────────┘        └─────────────────────────────┘
            │
            ▼
      ┌─────────────────────────┐
      │ Retrive relevant images │
      │ based on a fixed threshold│
      └─────────────────────────┘
```
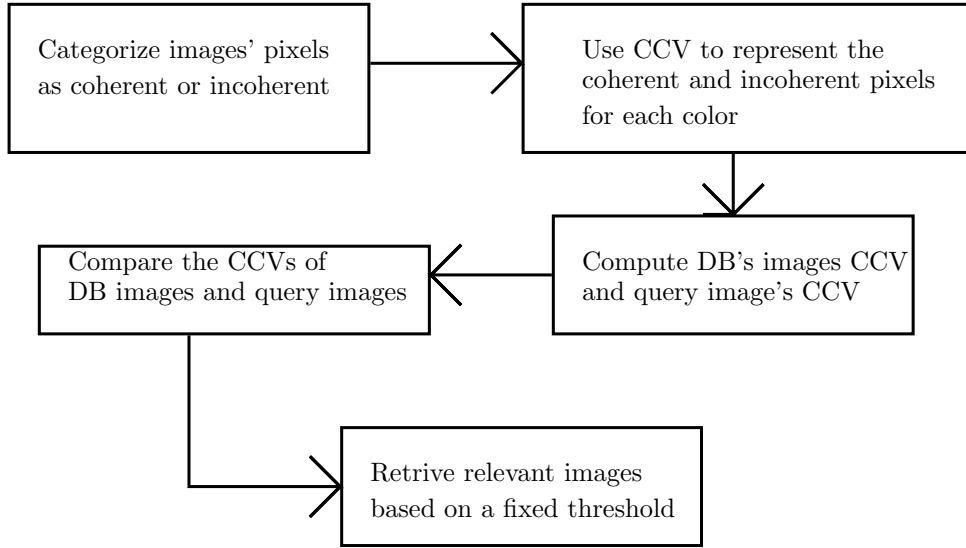
Figure 8: Image retrieval by using CCV

Results

We used the previously mentioned database. The database has categories like Africans and villages, Beaches, Buildings, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountains and glaciers, Food, Faces, Objects, Drawings, Textures and Natural scenes. Five images of each of the types are used as query images for the retrieval purpose. All these images quality are different. Some have uniform color distribution, some have non-uniform color distribution and others have widely scattered colors. The tables below shows parts of our result.

We can see the precision and recall of the color descriptors in the tables (Table 1 and 2). The color moments gave better result when the color distribution is of average level. Color histogram gives better result in case of uniform color distribution images. And for widely scattered colors Color Coherent Vector gives the best result. But the combination of different color descriptors produce remarkable results among all.

Table 1: **Category - Dinosaurs (Single Color Descriptor)**

Descriptor	Recall	Precision
CM	0.100137931	0.698748797
	0.09862069	0.754219409
	0.097586207	0.815092166
GCH	0.099586207	0.529325513
	0.09862069	0.580121704
	0.096551724	0.615114236
LCH	0.097586207	0.5
	0.095586207	0.508064516
	0.093586207	0.534672971
CCV	0.10062069	0.555597867
	0.099586207	0.585801217
	0.096551724	0.631483987

Table 2: **Category - Africans (Multiple Color Descriptors)**

Descriptor	Recall	Precision
GCH & CCV	0.088531187	0.871287129
	0.08249497	0.872340426
	0.079476861	0.877777778
LCH & CCV	0.100603622	0.487804878
	0.099597586	0.5
	0.097585513	0.510526316
CCV, CM, LCH & GCH	0.095573441	0.95959596
	0.09054326	0.97826087
	0.088531187	0.988764045
GCH, LCH & CCV	0.098591549	0.439461883
	0.096579477	0.507936508
	0.095573441	0.50802139

Conclusion

In the paper, comparison of three color feature for image retrieval is presented. The analysis and the data collection process that are presented in the paper are quite agreeable. These data can be used to further test and analyze the new systems built on the similar principles. As with time new methods of image retrieval are being discovered, these results as well as the process of achieving these results can come to help to further facilitate research in related areas.

Bibliography

[1] http://wang1.ist.psu.edu/.

[2] http://sipi.usc.edu/database/.

[3] http://vasc.ri.cmu.edu/idb/.

[4] http://pics.psych.stir.ac.uk/.

[5] http://www.chemtrailcentral.com/forum/album.php.

[6] S-K Chang and Arding Hsu. Image information systems: where do we go from here? *IEEE transactions on Knowledge and Data Engineering*, 4(5):431–442, 1992.

[7] Christos Faloutsos, Ron Barber, Myron Flickner, Jim Hafner, Wayne Niblack, Dragutin Petkovic, and William Equitz. Efficient and effective querying by image content. *Journal of intelligent information systems*, 3(3-4):231–262, 1994.

[8] L Fuhui, Z Hongjiang, and DF David. Multimedia information retrieval and management. 2003.

[9] Abby A Goodrum. Image information retrieval: An overview of current research. *Informing Science*, 3(2):63–66, 2000.

[10] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE transactions on information theory*, 8(2):179–187, 1962.

[11] Jing Huang, Shanmugasundaram Ravi Kumar, Mandar Mitra, and Wei-Jing Zhu. Image indexing using color correlograms, June 12 2001. US Patent 6,246,790.

[12] T Huang, Yong Rui, and Shinh-Fu Chang. Image retrieval: Past, present, and future. In *International Symposium on Multimedia Information Processing*, volume 108, 1997.

[13] Arnold WM Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1349–1380, 2000.

[14] Hideyuki Tamura and Naokazu Yokoya. Image database systems: A survey. *Pattern recognition*, 17(1):29–43, 1984.

[15] JM Zachary and Sitharama S Iyengar. Content based image retrieval systems. In *Application-Specific Systems and Software Engineering and Technology, 1999. ASSET'99. Proceedings. 1999 IEEE Symposium on*, pages 136–143. IEEE, 1999.