Feature Extraction Methods for Content Based Image Retrieval

Chesta Agarwal
chesta.agarwal@gmail.com
Delhi College of Engineering, Delhi University, Delhi, India

ABSTRACT

Pattern recognition is the act of taking in raw data and classifying it into predefined categories using statistical and empirical methods. Content Based Image Retrieval (CBIR) is one of the widely used applications of pattern recognition for finding images from vast and un-annotated image database. In CBIR images are indexed on the basis of low-level features, such as color, texture, and shape, which can automatically be derived from the visual content of the images. The paper discusses various techniques and algorithms that are used to extract these image features from the visual content of the images. The various similarity measures are used to identify the closely associated patterns. These methods compute the distance between the features generated for different patterns and identify the closely related patterns and these patterns are then generated as the result. The paper also discusses the variation in retrieval of images due to the different color spaces.

1. INTRODUCTION

Content-based image retrieval, a technique which uses visual contents to search images from large scale image databases according to users' interests, has been an active and fast advancing research area since the 1990s. During the past decade, remarkable progress has been made in both theoretical research and system development. However, there remain many challenging research problems that continue to attract researchers from multiple disciplines. Early techniques were not generally based on visual features but on the textual annotation of images. In other words, images were first annotated with text and then searched using a text-based approach from traditional database management systems. Text-based image retrieval uses traditional database techniques to manage images. Through text descriptions, images can be organized by topical or semantic hierarchies to facilitate easy navigation and browsing based on standard Boolean queries. However, due to rapid growth of technology the abundance of digital images a need for a system to organize un-annotated data has risen. Content Based Image Retrieval System addresses the problems of the early methods of handling huge critical database. It organises and analyses the data on the basis of the visual content of the images.

This paper discusses the various components of a CBIR system in Section 2. Section 3 discusses the general feature extraction approach. Section 4 describes the color content in an image retrieval system and analyses the various color spaces. It also discusses the various algorithms used to extract the color features of an image. Section 5 and Section 6 describe the techniques to extract the texture and shape components of an image respectively. Section 7 presents a general conclusion of the paper focussing on the potential of a CBIR system.

2. CONTENT BASED IMAGE RETREIVAL

The revolutionary internet and digital technologies have imposed a need to have a system to organize abundantly available digital images for easy categorization and retrieval. The need to have a versatile and general purpose content based image retrieval (CBIR) system for a very large image database has attracted focus of many researchers of information-technology-giants and leading academic institutions for development of CBIR techniques. Images can be retrieved thorough the indexes attached to them in the form of text or a hashing value generated by a hash function. However in case of un-annotated image database this form of *textual retrieval* becomes difficult, content based retrieval needs to be done in this case. Content based image retrieval is the retrieval of images from a large database depending on the visual content of the images rather than the textual content associated with the images.

This visual content may be the *general visual content*, represented by the low level feature descriptors like color, texture, shape, spatial relationship. The visual content may be *domain specific visual content*, specific to the application the CBIR system is developed for, for example the fingerprint matching system. The Domain specific content is application dependent and requires specific domain knowledge. A general Content Based Image Retrieval system is generally represented as in figure 1.

The various functional blocks of a general Content Based Image Retrieval System include an Image Handler, a Feature Extractor, a Similarity Comparator, a Database Handler and a GUI handler amongst its major components. The *Image Handler* module analyzes the image format of the input image. It reads the input image and converts the image into a data matrix which stores the color components and intensity values according to the different color spaces for each pixel. If RGB color space is considered for example, then the data matrix would be a 3 dimensional matrix of the dimension **Height X Width X 3**, storing the R, G and B component values for each pixel. The Image handler module is also responsible for converting the data matrix back to the image file while saving the results generated by the CBIR system, i.e. while saving the generated result in the form of an image rather than a matrix. The *GUI handler* module manages the entire GUI of the system required to ease the end user in specifying the query and to view the results. The *Feature Extractor* module is responsible for extracting the various low level and high level features of the image using the various techniques for extracting the color, texture and shape parameters of the image. The feature extractor would calculate the features of the image as a whole and of various sub images within the main image. The features are then saved in the database in the form of tuples of the different tables. The feature vector comprises of the pixel coordinates of the top left corner of the sub image along with the feature parameters corresponding to the method used for feature extraction. A typical feature vector looks like,

$$F_i = \{ x_i, y_i, f_1, f_2, ...f_n \} \quad (1)$$

Where, F_i is the feature vector corresponding to the i^{th} sub image. x_i and y_i are the pixel coordinates of the top left corner of the rectangular sub image. And f_1, f_2,....f_n are the parameters generated by the various techniques. The feature vectors corresponding to all the sub images or the image as a whole are stored in the database for future retrieval by the Similarity Comparator module. Further details of the Feature Extractor and the various techniques are discussed in section 3 onwards. The storing of these feature parameters and their retrieval is done from the database by the Database Handler module. The calculation of the image parameters is a one-time task thereby reducing the running time and the response time of the system required to generate the results. The system does the task of understanding the input query and calculating the similarity between the input query image and the images or the sub images in the database. This task is done by the *Similarity Comparator* module of the system. The similarity may be measured using the various distance measuring techniques. These are general modules of a CBIR system. Other module can be added to the system depending on the application the system is being developed for.

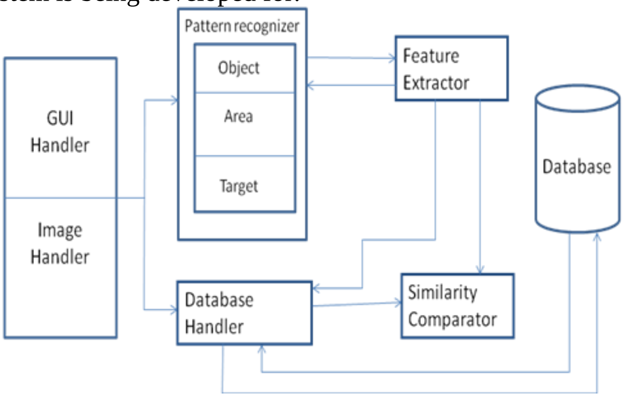

FIGURE 1. A GENERAL CONTENT BASED IMAGE RETRIEVAL SYSTEM

3. FEATURE EXTRACTION

Feature extraction algorithms and similarity measures used for image comparison underlie any CBIR system. A CBIR system uses the visual contents of an image such as *color*, *shape*, *texture*, and *spatial layout* to represent and index the image. In typical content-based image retrieval systems (Figure 1-1), the visual contents of the images in the database are extracted and described by multi-dimensional feature vectors. The feature vectors of the images in the database form a feature database. To retrieve images, users provide the retrieval system with example images or sketched figures. The system then changes these examples into its internal representation of feature vectors. The similarities /distances between the feature vectors of the query example or sketch and those of the images in the database are then calculated and retrieval is performed with the aid of an indexing scheme. All content-based retrieval methods can be classified into classes depending on the features they use: search by color, texture, or shape. Each class, in turn, is divided into subclasses by the type of the algorithm used for constructing the feature vector. Some researchers classify spatial features of images into a separate class. Spatial features are those reflecting spatial layout of homogeneous image regions in terms of one or another feature: for example, region of the same color or the same texture or a particular object. In other words, these are features of one of the classes (color, texture, or shape) with additional information on spatial layout. In what follows, the common algorithms for color, texture, and shape feature extraction are considered. For each of these classes, a more detailed classification is presented in a separate section.

4. COLOR

Color feature is the most significantly used visual content descriptor. It plays a vital role in searching collections of color images of arbitrary subject matter. Color has a strong significance in the human visual perception mechanism. Besides the simplicity involved in analyzing the color components of any image, color also offers special properties of being easy to extract and analyze and its property of invariance with respect to the size of the image and orientation of objects on it. These characteristics of the color component of an image make it the most sought after choice in the image retrieval systems. The quality of color feature vector greatly depends on the color space selection. The different color spaces behave differently with different applications and algorithms applied.

4.1. Color Spaces

A color space (also referred to as a *color model* or *color system*) is a specification of a coordinate system and a subspace within this system where each color is represented by a single point. Thus, each color in a color space has its *color coordinates. C*olor model is an abstract mathematical model describing the way colors can be represented as tuples of numbers, typically as three or four values or *color components.* The commonly used color spaces are as follows:

RGB, red, green and blue. The RGB color model is an additive color model in which red, green and blue light are added together in various ways to reproduce a broad array of colors. RGB is a *device-dependent* color model: different devices detect or reproduce a given RGB value differently. It is widely used for image display in monitors, cameras, etc. due to the close device dependence of the RGB color model.

CMY, cyan, magenta, yellow or the CMYK, cyan, magenta, yellow and black are two other similar color models. The CMYK color model is a subtractive color model, used in color printing since the color printing process also involves 3 inks and uses subtractive process to generate the different colors. The conversion between RGB color space and CMY/CMYK color space is rather a little difficult since both the color spaces are device dependent. Hence separate color management systems need to be employed to interchange between the two color spaces.

HSV, hue, saturation, value (also referred to as HSB, B for brightness) and HSI (I for Intensity) are two other most commonly used color space models used in image processing and analysis applications. It is a cylindrical coordinate representation of the RGB points in the Cartesian coordinate representation of the RGB color space. The hue is invariant to the changes in illumination and camera direction and hence more suited to object retrieval. HSV color model is a more intuitive way of describing the color parameters in comparison to the other color models. It represents the way how human eye and brain observes and analyzes the color in terms of its chromaticity and brightness instead of realising a color image as a sum of 3 primary sub images. The following expressions convert a RGB color space system in the range [0,1] to a HSV color space in the range [0,1]:

$$V = \max(R,G,B) \tag{2}$$

$$S = \frac{V - \min(R,G,B)}{V} \tag{3}$$

$$H = \frac{G-B}{6S}, \text{ if } V = R \tag{4}$$

$$H = \frac{1}{3} + \frac{B-R}{6S}, \text{ if } V = G \tag{5}$$

$$H = \frac{2}{3} + \frac{R-G}{6S}, \text{ if } V = B \tag{6}$$

The traditional color space for representing digital images is RGB color space. However image processing algorithms typically use HSV color space.

The following sections describe the various color feature extraction techniques.

4.2. Color Based Feature Extraction

4.2.1 Color Moments

Color moments have been successfully used in many retrieval systems (like *QBIC*), especially when the image contains just the object. The *first order* (*mean*), the *second* (*variance*) and the *third order* (*skewness*) color moments have been proved to be efficient and effective in representing color distributions of images.

Mathematically, the first three moments are defined as:

$$\text{Energy} = E_i = \frac{1}{N}\sum_{j=1}^{N} p_{ij} \tag{7}$$

$$\text{Standard Deviation} = \sigma_i = \left(\frac{1}{N}\sum_{j=1}^{N}(p_{ij}-E_i)^2\right)^{\frac{1}{2}} \tag{8}$$

$$\text{Skewness} = s_i = \left(\frac{1}{N}\sum_{j=1}^{N}(p_{ij}-E_i)^3\right)^{\frac{1}{3}} \tag{9}$$

The CBIR systems usually use 3 numbers (parameters) for each band of the image. Hence a total of 9 parameters (in case of 3 bands: RGB, HSV) are used to represent the entire image. The Color moments are easy to compute and compact set of parameters facilitating easy similarity distance measurement.

4.2.2. Color Moments with Fuzzy Regions

Color moments though a very simple and efficient technique has a few shortcomings. It does not take into account spatial layout of colors. To overcome this, a modification to the above technique is practised. The image is partitioned into layout regions of fixed size (or more complicated image segmentation) and calculating features of color distribution for each of them. The image is divided into “fuzzy regions.” The following five regions are introduced: central ellipsoidal region and four surrounding regions as in figure2. All regions are defined by membership value described by a membership matrix. An example of a membership matrix is shown in figure3.

According to this membership matrix, pixels located strictly in the centre of the image completely belong to the central region and thus affect the feature vector of the central region only. The closer the pixel to the region border, the lesser is its influence to the region’s feature vector. Pixels located on a border separating two regions affect the feature vectors of both regions. Experiments show that such an approach makes it possible to improve retrieval results in the case of more complicated queries, when it is required to take into account spatial layout of objects on the image.

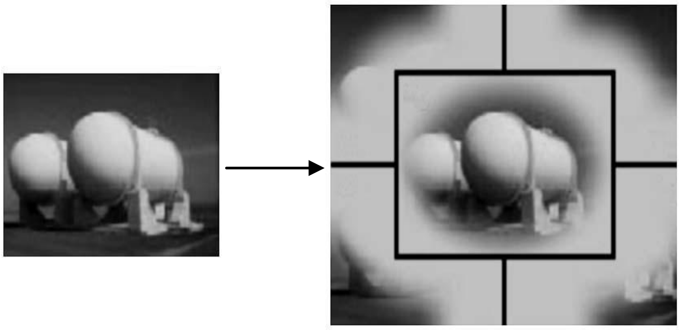

FIGURE 2. PARTITIONING AN IMAGE INTO FUZZY REGIONS

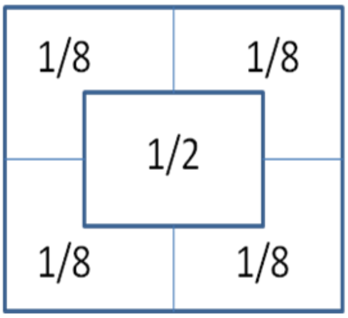

FIGURE 3. MEMBERSHIP MATRIX

The equations described in above section (eq. 7 to 9) are applied to each fuzzy region to generate 9 parameters (for 3 bands) for each fuzzy region.

The overall Parameters of the image are then computed as

$$F(v) = \sum_{i=1}^{5} \mu_i * f_i(v) \qquad (10)$$

where, F(v) is the value of the overall parameter for parameter v (v = mean, standard deviation, skewness) μ_i is the value membership value of a fuzzy region in an image and $f_i(v)$ is the value of the parameter v in the i^{th} region. Here though each of the five regions has 9 parameters but the overall parameters of the image remain 9 only. Color moments with fuzzy regions though computationally expensive as compared to the Classical Color Moments approach, generate more efficient results due to larger contribution by the central pixels than those at the boundary.

4.2.3. Color Histogram

The color histogram serves as an effective representation of the color content of an image if the color pattern is unique compared with the rest of the data set. The color histogram is easy to compute and effective in characterizing both the distribution of colors in an image. In addition, it is robust to translation and rotation about the view axis and changes only slowly with the scale, occlusion and viewing angle. Since any pixel in the image can be described by three components in a certain color space (for instance, red, green, and blue components in RGB space, or hue, saturation, and value in HSV space), a *histogram*, i.e., the distribution of the number of pixels for each quantized bin, can be defined for each component. Clearly, the more bins a color histogram contains, the more discrimination power it has. However, a histogram with a large number of bins will not only increase the computational cost, but will also be inappropriate for building efficient indexes for image databases. Furthermore, a very fine bin quantization does not necessarily improve the retrieval performance in many applications. In addition, color histogram does not take the spatial information of pixels into consideration, thus very different images can have similar color distributions. This problem becomes especially acute for large scale databases. To increase discrimination power, several improvements have been proposed to incorporate spatial information. A simple approach is to divide an image into sub-areas and calculate a histogram for each of those sub-areas. As introduced above, the division can be as simple as a rectangular partition, or as complex as a region or even object segmentation. Increasing the number of sub-areas increases the information about location, but also increases the memory and computational time.

5. TEXTURE

Like color, image texture is also very important for visual human perception. It gives us information on structural arrangement of surfaces and objects on the image. Texture is not defined for a separate pixel; it depends on the distribution of intensity over the image. Texture possesses periodicity and scalability properties; it can be described by main directions, contrast, and sharpness. Texture analysis plays an important role in comparison of images supplementing the color feature. There exist many methods for representation of information on image texture and comparison of images on the basis of texture. Texture features are classified into statistical, geometrical, model, and spectral ones. Statistical features describe distribution of intensity over the image by means of various statistical parameters. The most frequently used statistical features include:

– general statistical parameters calculated from pixels' intensity values,

– parameters calculated based on the co-occurrence matrices,

– texture histograms built upon the Tamura features.

It is common for geometrical features to describe texture with *texture elements*, or *primitives*. They specify texture by defining texture primitives and rules of their placement relative to one another. Structural methods are not appropriate for describing irregular textures. Model methods of texture analysis rely on construction of a model that can be used not only for description but also for texture generation. Spectral features describe texture in the frequency domain. They are based on expanding a signal in terms of basis functions and using the expansion coefficients as elements of the feature vector. Few of the frequently used techniques for extracting texture features are as described in the following section.

5.1. Texture Based Feature Extraction

5.1.1. Gray Level CoOccurrence Matrix

The grey level co-occurrence matrix is defined by Haralick et al. as a frequency matrix of pairs of pixels of certain intensity levels located with respect to one another in a certain way. Mathematically, the co-occurrence matrix C of an image I of size $N \times M$ is defined as follows:

$$C(i,j) = \sum_{p=1}^{N}\sum_{q=1}^{M}\begin{cases}1, \text{ if } I(p,q) = i, \\ I(p+\Delta x, q+\Delta y) = j, \\ 0, \text{ otherwise,}\end{cases} \quad (11)$$

Where, $(\Delta x, \Delta y)$ is an offset (or displacement) vector determining mutual location of pixels and $I(p, q)$ is intensity of a pixel located in position (p, q). An example of a co-occurrence matrix for an image region for $(\Delta x, \Delta y) = (1, 0)$ is shown in Fig 4. It is not difficult to see that the offset $(\Delta x, \Delta y)$ makes such a representation of the texture feature sensitive to image rotation, which is usually undesirable for a CBIR system. Therefore, it was suggested to construct several matrices with different offset vectors corresponding to image rotation through0°, 45°, 90°, and 135°. Further, various statistical descriptors can be extracted from the co-occurrence matrices as the texture representation of images. Haralick et al. Suggested 14 descriptors, including the angular second moment, contrast (variance, difference moment), correlation, and others. Each descriptor represents one texture property. For example, the angular second moment shows proximity of the co-occurrence matrix distribution to the diagonal matrix distribution, which corresponds to the texture homogeneity. The correlation measures the number of local variations of the intensity level, i.e., "diversity of colors" of the image. However, the use of all 14 descriptors, together with the necessity of constructing several co-occurrence matrices for one image (for different offset vectors), would make any image retrieval (or classification) system extremely slow in view of the great amount of calculation needed to process each image. The following descriptors are considered to be the most discriminative ones and are used more frequently than others: (eq. 12-20)

$$\text{Mean} = \text{mu\_i} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} i * P[i][j] \quad (12)$$

$$\text{Variance} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} (i - \text{mu\_i})\text{\^{}}2 * P[i][j] \quad (13)$$

$$\text{Skewness} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} ((P[i][j]-\text{mu\_i})/(\text{sd\_i}))\text{\^{}}3 \quad (14)$$

$$\text{Entropy} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} P[i][j] * \log(P[i][j]) \quad (15)$$

$$\text{Angular Second Moment} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} (P[i][j])\text{\^{}}2 \quad (16)$$

$$\text{Contrast} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} (i-j)\text{\^{}}2 * P[i][j] \quad (17)$$

$$\text{Homogeneity} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} 1/(1+(i-j)\text{\^{}}2) * P[i][j] \quad (18)$$

$$\text{Standard Deviation} \quad (\text{variance})\text{\^{}}1/2 \quad (19)$$

$$\text{Correlation} = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} ((i-\text{mu\_i})*(j-\text{mu\_j})* P[i][j])/(sd*sd) \quad (20)$$

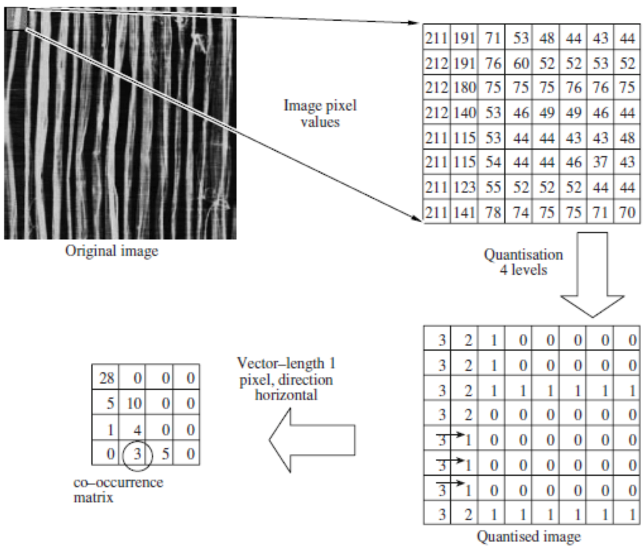

FIGURE 4. AN EXAMPLE CALCULATION OF THE CO-OCCURRENCE MATRIX FOR A REGION OF THE IMAGE WITH $(\Delta X, \Delta Y) = (1, 0)$

5.1.2. Gabor Filter

The *Gabor filter* has been widely used to extract image features, especially texture features. It is optimal in terms of minimizing the joint uncertainty in space and frequency, and is often used as an orientation and scale tuneable edge and line (bar) detector. There have been many approaches proposed to characterize textures of images based on Gabor filters. The basic idea of using Gabor filters to extract texture features is as follows.

A two dimensional Gabor function $g(x, y)$ is defined as: (eq. 21)

$$G_{\theta,f}(x,y) = \exp\left(-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right) \cdot \cos(2\pi f x_\theta)$$

with,

$$\begin{cases} x_\theta = x\sin\theta + y\cos\theta \\ y_\theta = x\cos\theta - y\sin\theta \end{cases}, \qquad (22, 23)$$

where f gives the frequency of the sinusoidal plane wave at an angle θ with the x - axis and σx and σy are the standard deviations of the Gaussian envelope along the x and y axes, respectively.

Basically, Gabor filters are a group of wavelets, with each wavelet capturing energy at a specific frequency and a specific direction. Filter $G_{\theta, f}(x, y)$ computed in eq. 21, optimally captures both local orientation and frequency information from a digital image. Each image is filtered using the Gabor function, at various orientations, frequencies and standard deviations.

6. SHAPE

Shape features of objects or regions have been used in many content-based image retrieval systems.

Construction of the feature vector representing shape of an object on the image can be divided into several stages. The first stage is image segmentation; the second stage is shape representation in terms of abstractions; then, description of the result obtained in terms of a finite set of number parameters composing the feature vector (shape parameterization). On each stage, different algorithms are used. Note that the method of shape parameterization depends obviously on the method of shape representation, whereas the segmentation and extraction of objects is a separate task. The method of image segmentation consists in determining homogeneous regions on the image (or heterogeneities, as boundaries between the regions). The regions obtained are described by means of the shape feature vectors. Further in this section. The simplest shape features are centre of gravity, area, direction of the principal axis, perimeter, and eccentricity. These parameters are easy to determine, but they have limited capabilities. For more accurate search by shape, more complicated methods are used, which make it possible to describe a figure in more detail. Methods for representing and describing shapes can be divided into two groups: external methods, which represent the region in term of its external characteristics (its boundary), and internal ones, which represent the region in terms of its internal characteristics (the pixels comprising the region). The former are also referred to as boundary-based, since they use only the outer boundary of the shape, and the latter are called region-based, since they use information on the entire shape region.

6.1. Shape Based Feature Extraction

6.1.1. Moment Invariants

Classical shape representation uses a set of *moment invariants*. If the object R is represented as a binary image, then the central moments of order $p+q$ for the shape of object R are defined as:

$$\mu_{p,q} = \sum_{(x,y)\in R} (x-x_c)^p (y-y_c)^q \qquad (24)$$

where (xc, yc) is the center of object. This central moment can be normalized to be scale invariant.

$$\eta_{p,q} = \frac{\mu_{p,q}}{\mu_{0,0}^{\gamma}}, \quad \gamma = \frac{p+q+2}{2} \qquad (25)$$

Based on these moments, a set of moment invariants to translation, rotation, and scale can be derived through the following eq. 26 – 32)

$$\phi_1 = \mu_{2,0} + \mu_{0,2}$$

$$\phi_2 = (\mu_{2,0} - \mu_{0,2})^2 + 4\mu_{1,1}^2$$

$$\phi_3 = (\mu_{3,0} - 3\mu_{1,2})^2 + (\mu_{0,3} - 3\mu_{2,1})^2$$

$$\phi_4 = (\mu_{3,0} + \mu_{1,2})^2 + (\mu_{0,3} + \mu_{2,1})^2$$

$$\begin{aligned}\phi_5 = {} & (\mu_{3,0} - 3\mu_{1,2})(\mu_{3,0} + \mu_{1,2})\left[(\mu_{3,0} + \mu_{1,2})^2 - 3(\mu_{0,3} + \mu_{2,1})^2\right] \\ & + (\mu_{0,3} - 3\mu_{2,1})(\mu_{0,3} + \mu_{2,1})\left[(\mu_{0,3} + \mu_{2,1})^2 - 3(\mu_{3,0} + \mu_{1,2})^2\right]\end{aligned}$$

$$\phi_6 = (\mu_{2,0} - \mu_{0,2})[(\mu_{3,0} + \mu_{1,2})^2 - (\mu_{0,3} + \mu_{2,1})^2] + 4\mu_{1,1}(\mu_{3,0} + \mu_{1,2})(\mu_{0,3} + \mu_{2,1})$$

$$\phi_7 = (3\mu_{2,1} - \mu_{0,3})(\mu_{3,0} + \mu_{1,2})\left[(\mu_{3,0} + \mu_{1,2})^2 - 3(\mu_{0,3} + \mu_{2,1})^2\right]$$

6.1. 2. Grid Based

Grid Based method is an easy to compute method used mainly for the description of object shape. The basic idea of the method is as follows: (1) a grid with cells of certain size is superimposed on the object, and (2) the cells of the grid are traversed from the right to the left and from top to bottom; if the cell covers at least a part of the object, it is associated with number 1; otherwise, with 0. The sequence of zeros and ones obtained in this way describes the shape of the object. Let us illustrate this by an example. The objects depicted in Fig. 5 (a, b) are associated with the following vectors:

(a)001111000011111111111111111111111111 111110111 0111000011

(b)001100000011100000111100000111101111 111111110 001111000.

The authors of the method proposed to determine similarity of two figures in terms of the number of cells covered by only one figure. In other words, the measure of similarity in this feature space is defined as a sum of elements of the vector that is the result of the XOR operations on the two original vectors. It should be noted that this shape representation is not invariant with respect to rotation and scaling. Invariance of the representation is achieved through preliminary normalization of the figure relative to its major axis. Placing the figure in such a way that its major axis is parallel to the *X* axis (Fig. 6), we get invariance with respect to rotation. Before comparing two figures, we change their scales such that their major axes have equal length. This will make the representation scale invariant.

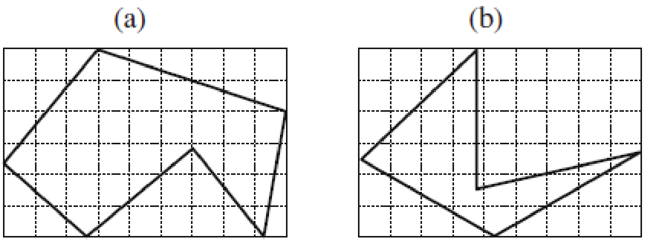

FIGURE 5. IMPOSING FIGURES ON THE GRID TO GET INDEX

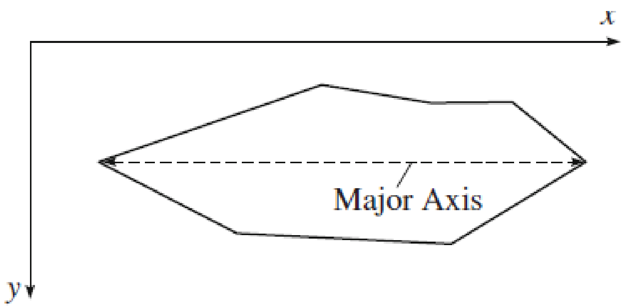

FIGURE 6. NORMALISATION OF A FIGURE WITH RESPECT TO THE MAJOR AXIS

7. Conclusion

The content-based image retrieval is an interesting and complex problem studied by many researchers all over the world. The complexity of this problem is due to many factors. The most important among them are a “semantic gap” between semantic content of the image and its visual features, and necessity to search large amount of multidimensional data in real time. In this paper, few popular algorithms for constructing feature vectors that reflect various image features, such as color, texture, and shape of the objects have been discussed. The quality of the retrieval is determined, first of all, by these algorithms: the feature vector should reflect specific features of an image that make it possible to determine the degree of similarity of the query with other images from the collection. The metrics defined on the feature space should reflect the human visual perception. Each of the considered algorithms analyzes a certain feature, thus, making it possible to perform retrieval only on the basis of this feature. To get acceptable retrieval quality, it is required to combine various algorithms. This is especially true in retrieval of heterogeneous collections, for which it is impossible to extract the most important feature for all images of the collection. In addition to this, development of a valuable image retrieval system requires solving multidimensional indexing problem, visualizing system responses, and providing the user with a convenient and intuitively clear way of query formulation.

References

[1]. SAJJANHAR, A. AND LU, G., A GRID BASED SHAPE INDEXING AND RETRIEVAL METHOD, *AUSTRALIAN COMPUT. J.*, 1997, VOL. 29, ISSUE 4, PP. 131–140.

[2]. HU, M.-K., VISUAL PATTERN RECOGNITION BY MOMENT INVARIANTS, *IEEE TRANS. INFORMATION THEORY*, 1962, VOL. 8, ISSUE 2, PP. 179–187.

[3]. A. K. JAIN, AND F. FARROKNIA, "UNSUPERVISED TEXTURE SEGMENTATION USING GABOR FILTERS," *PATTERNRECOGNITION*, VO.24, NO.12, PP. 1167-1186, 1991.

[4]. M. K. HU, "VISUAL PATTERN RECOGNITION BY MOMENT INVARIANTS," IN J. K. AGGARWAL, R. O. DUDA, AND A.ROSENFELD, *COMPUTER METHODS IN IMAGE ANALYSIS*, IEEE COMPUTER SOCIETY, LOS ANGELES, CA, 1977.

[5]. L. YANG, AND F. ALGREGTSEN, "FAST COMPUTATION OF INVARIANT GEOMETRIC MOMENTS: A NEW METHOD GIVING CORRECT RESULTS," *PROC. IEEE INT. CONF. ON IMAGE PROCESSING*, 1994.

[6]. TUDOR BARBU, “CONTENT–BASED IMAGE RETRIEVAL USING GABOR FILTERING” IN 20TH INTERNATIONAL WORKSHOP ON DATABASE AND EXPERT SYSTEMS APPLICATION, 2009

[7]. VIDYA MANIAN AND MARCEL J. CASTRO, “ COLOR IMAGE CLASSIFICATION USING GABOR FILTERS”, COMPUTING RESEARCH CONFERENCE PROCEEDINGS, MAY, 2000

[8]. A. JAIN, N. RATHA, S. LAKSHMANAN, “OBJECT DETECTION USING GABOR FILTERS”, IN PATTERN RECOGNITION, VOL. 30, PP. 295-309, 1997.

[9]. T. BARBU, “NOVEL AUTOMATIC VIDEO CUT DETECTION TECHNIQUE USING GABOR FILTERING”, COMPUTER AND ELECTRICAL ENGINEERING

[10]. M. STRICKER, AND M. ORENGO, "SIMILARITY OF COLOR IMAGES," *SPIE STORAGE AND RETRIEVAL FOR IMAGE AND VIDEO DATABASES III*, VOL. 2185, PP.381-392, FEB. 1995

[11]. J. D. FOLEY, A. VAN DAM, S. K. FEINER, AND J. F. HUGHES, *COMPUTER GRAPHICS: PRINCIPLES AND PRACTICE*, 2ND ED., READING, MASS, ADDISON-WESLEY, 1990.

[12]. STRICKER, M. AND DIMAI, A., SPECTRAL COVARIANCE AND FUZZY REGIONS FOR IMAGE INDEXING, *MACHINE VISION APPLICATIONS*, 1997, VOL. 10, PP. 66–73.

[13]. P. BRODATZ, "TEXTURES: A PHOTOGRAPHIC ALBUM FOR ARTISTS & DESIGNERS," DOVER, NY, 1966

[14]. N.S.VASELIER, “CONTENT-BASED IMAGE RETRIEVAL METHODS”, PROGRAMMING AND COMPUTER SOFTWARE,2009, VOL.35, NO.3, PP. 158-180

[15]. P.S.HIREMATH, S. SHIVASHANKAR, AND JAGADEESH PUJARI, “WAVELET BASED FEATURES FOR COLOR TEXTURE CLASSIFICATION WITH APPLICATION TO CBIR”. INTERNATIONAL JOURNAL OF COMPUTER SCIENCE AND NETWORK SECURITY, VOL. 6 NO. 9A, SEPT, 2006

[16]. B.S. MANJUNATH AND W. Y. MA, “ TECTURE FEATURES FOR BROWSING AND RETRIEVAL OF IMAGE DATA”, IEEE TRANSACTION ON PATTERN ANALYSIS AND MACHINE LEARNING, VOL. 18, NO. 8, AUG, 1996