

# An Improved Image Retrieval Method Based On Color Spatial Distribution and Shape Curve of Object

Wenbing Chen<sup>1, +</sup>, Qin Xu<sup>1</sup>

<sup>1</sup> College of Math and Physics, Nanjing University of Information Science & Technology,

Nanjing, 21004, China

(Received April 18, 2011, accepted August 29, 2011)

**Abstract.** Currently, the low-level image features such as color, texture, and shape are widely used for content-based image retrieval. However, it is rather difficult to realize an exact match between two images by using a single feature. To address this problem, a method combining color spatial distribution and shape of the object is presented. In this method, firstly, an improved mean shift algorithm is presented to segment an image into clusters. Secondly, based on these clusters a novel histogram is defined. Thirdly, based on each cluster, the shape curve of an object is extracted. Finally, integrating the information of color and shape of the objects in an image, a novel similarity measure is presented to realize an exact match between two images. Experimental results show that our method can efficiently increase match precision.

**Keywords:** image retrieval, principal cluster, shape curve, boundary point, transformation invariance.

## 1. Introduction

Currently, low-level image features such as color, texture, and shape are widely used for content-based image retrieval. Since color is intuitive, stable and simple, it is natural to use the color feature to express and identify an image. A color histogram is the most commonly used method to express the color feature. To use a color histogram to match two images, Swain and Ballard [1] proposed a histogram intersection method. Since then new methods using a histogram to match two images have been continuously proposed and improved [2-4]. However, the main drawback of histograms for classification is that the representation is dependent on the color of the object being studied, ignoring its shape and structure. Conversely, without spatial or shape information, similar objects of different color may be indistinguishable based solely on color histogram comparisons. There is no way to distinguish a red and white cup from a red and white plate. Usually, Euclidean distance, histogram intersection, or cosine or quadratic distances are used for the calculation of image similarity ratings. However, none of these values reflect the degree of similarity between two images in itself; it is useful only when used in comparison to other similar values. Shape is an important feature to identify a natural object in the real world. In some cases, an image can be regarded as a combination of natural objects. So, if we can express an object as a shape curve, an image can be approximately expressed as a sequence of shape curves of objects.

Currently, the main methods for extracting the shape feature of an object. are classified into two categories: region-based segmentation methods and edge-based detection methods. For the former category, mean shift segmentation and EM-GMM algorithms can be used to segment an image into regions of objects, e.g., clusters[5-10]. In a certain sense, a cluster can be regarded as an abstract expression of a natural object. For the latter category, traditional edge-based detection algorithms, such as Sobel, Laplacian of Gaussian (LoG) and Laplacian algorithm [13], can be used to detect the edge of an object. A novel snake algorithm [14, 15] can also segment an image and yield the edge of an interesting object. However, although the edge-based detection algorithms can detect the edge of an object, it is rather difficult to extract the edge feature of a single object from the detection result, which is a drawback for edge-based detection algorithms.

Based on the analysis above, since the detection result of the first category can be more easily used to extract a single object than the second one, in this paper an improved mean shift is presented to segment an

<sup>+</sup> Corresponding author. Tel.: +86-58731160.  
E-mail address: chenwb@nuist.edu.cn.

image into clusters. Furthermore, each cluster is regarded as a natural object. Hence, it is straightforward to use the cluster to extract the shape curve of a natural object.

In this paper, our framework intends to define a new similarity measure to match between two images, where such a similarity measure integrates both the information from color spatial distribution and shapes of objects. Intuitively it should have higher match precision than one using a single feature.

This paper is organized as follows. In section 2, the mean shift algorithm is briefly reviewed, and an improvement is presented with respect to the algorithm. In section 3, based on clusters of an image, methods for extracting a single object and generating the shape curve of an object are presented. In section 4, integrating information of color spatial distribution and shape of objects within an image, and a similarity measure are presented. In section 5, experiments are carried out and comparison made. In section 6, conclusion is given.

## 2. Segmenting an image into clusters

In this section, we briefly review the basic idea of the mean shift algorithm and give the general process of the two procedures of mean shift filtering and segmenting. Based on the above framework, a new expression of a cluster during the segmentation procedure is presented in order to satisfy the necessity of extracting a single object in successive sections.

### 2.1 A brief review of mean shift algorithm

Let  $A$  be a finite set in  $n$ -dimensional space. A formula computing  $n$ -dimension mean shift vector at location is defined by

$$ms(x) = \frac{\sum_a k(a-x)w(a)a}{\sum_a k(a-x)w(a)} - x \quad x \in a \quad (1)$$

where  $k(x)$  is a kernel function [15,16],  $w$  a weight factor,  $x$  the center of the kernel(window), and  $a$  in the kernel at the center of  $x$ . An image is typically represented as a two-dimensional lattice of  $p$ -dimensional vectors(pixels), where  $p = 1$  in the gray-level case, three for color images. The space of the lattice is known as the spatial domain, while the gray level or color is represented in the range domain. For the range domain, the RGB color space is not perceptually uniform, while the LUV space is approximately uniform. Hence, to keep the perceived image uniform to human vision, RGB space is transformed into LUV space. For both domains, the Euclidean metric is assumed. When the location and range vectors are concatenated in the joint spatial-range domain of dimension  $d = p + 2$ , their different nature has to be considered by proper normalization. Thus, the multivariate kernel is defined as the product of two radically symmetric kernels and the Euclidean metric allows a single bandwidth parameter for each domain.

$$G_{h_s, h_r}(x) = \frac{1}{h_s^2 h_r^2} g\left(\left\|\frac{x^s}{h_s}\right\|^2\right) g\left(\left\|\frac{x^r}{h_r}\right\|^2\right) \quad (2)$$

where  $x^s$  is the spatial part,  $x^r$  is the range part of a feature vector,  $g(x)$  the common profile used in both two domains antitheism paper a Gaussian profile,  $h_s$  and  $h_r$  are the used kernel bandwidths.

### 2.2 Mean shift filtering and segmenting

#### 2.2.1 Mean shift filtering

Smoothing through replacing the pixel in the center of window by the weighted average of the pixels in the window indiscriminately blurs the image, removing not only the noise but also salient information. Therefore, to enhance the robustness of the next segmentation and obtain a better segmented result, filtering an image is indispensable. Let  $x_i$  and  $z_i, i = 1, \dots, n$ , be the  $n$ -dimensional input and filtered image pixels in the joint spatial-range domain.

#### 2.2.2 Mean shift segmenting

Image segmentation partitions a gray level or color image into homogeneous regions. Homogeneity is usually defined as similarity in pixel values, i.e., a piecewise constant model is enforced over the image. A mean shift segmentation is given as follows.

Let  $x_i$  and  $z_i, i = 1, \dots, n$ , be the  $n$ -dimensional input and filtered image pixels in the joint spatial-range domain and  $L_i$  the label-classified of the  $i$  pixel in the segmented image. Thus, the algorithm of the segmenting procedure can be described below.

1. Run the mean filtering procedure for an input image and store all the information about the  $d$ -dimensional convergence point into  $z_i$ .
2. Generate in the joint domain the clusters  $\{C_s(i)\}_{i=1, \dots, m}$  by grouping together all  $z_i$  which are closer than  $h_s$  in the spatial domain and  $h_r$  in the range domain.
3. For each  $i = 1, \dots, n$ , assign  $L_i = \{j | C_s(j)\}$ .
4. Abandon the spatial regions containing less than  $M$  pixels, i.e., only keep the principal cluster.

### 2.2.3 Redefining data structure of mean shift segmenting

To efficiently identify an object, the data structures of both cluster and pixel of the segmented image must be redefined. Suppose an input image  $I$  is segmented into  $m$  clusters by the above segmenting algorithm. Each cluster can be regarded as an approximate identification of a principal object. Furthermore, for a given object, if the object is regarded as a cluster, the cluster should be described by three main features, average color of the cluster, the number of pixels in the cluster and all pixels of which the cluster consisted. Here, the average color of the cluster is the color of a homogenous region, the number of pixels in the cluster represents the scale of an object, and all pixels in the cluster represent a specific object. Let  $I_s$  be a segmented image of input image  $I$ . We can use two arrays to describe  $I_s$ . Suppose  $I_s$  consisted of  $m$  clusters, i.e.,  $I_s$  is a set of  $m$  clusters and the sequence is denoted by  $CS$ . Moreover, let  $cs$  be a cluster.

$cs$  Consists of two components, that is,  $cs = (c, pn)$  where  $c$  is average color and  $pn$  the number of pixels of  $cs$ . Then  $CS = \{cs(i)\}_{i=1}^m$ . Hence, array  $CS$  approximately represents the principal objects of  $I_s$  or  $I$ . Furthermore, we need to represent the spatial distribution of pixels of each cluster. Define a new array  $P = \{p(j, i) | i = 1, \dots, m; j = 1, \dots, n_m\}$ , where element  $p(j, i)$  represents the  $j$ th pixel of the  $i$ th cluster, the element  $p(j, i)$  is  $2-d$  vector, i.e.,  $p(j, i) = (x, y)$  where  $(x, y)$  is the spatial location of the  $j$ th pixel. In summary, we obtain a new approximate form of image  $I_s$  or  $I$ . In the next sections, the novel form is described by

$$CS = \{cs(i)\}_{i=1}^m = \{(c(i), pn(i))\}_{i=1}^m \quad (3a)$$

$$P = \{p(j, i) | i = 1, \dots, m; j = 1, \dots, n_m\} \quad (3b)$$

(3a) can be used to approximately describe a segmented image, whereas (3b) represents a sequence of single objects in the image.  $m$  is the number of clusters of the image and  $n_i$  the number of pixels of the  $i$ th cluster. The relationship between two arrays is shown below.

$$\begin{aligned} cs(1) &= \begin{cases} p(1,1) = ((x(1), y(1)), 1) \\ p(2,1) = ((x(2), y(2)), 1) \\ p(3,1) = ((x(3), y(3)), 1) \\ \dots \\ p(n_1,1) = ((x(n_1), y(n_1)), 1) \end{cases} \\ cs(2) &= \begin{cases} p(1,2) = ((x(1), y(1)), 2) \\ p(2,2) = ((x(2), y(2)), 2) \\ p(3,2) = ((x(3), y(3)), 2) \\ \dots \\ p(n_2,2) = ((x(n_2), y(n_2)), 2) \end{cases} \end{aligned}$$

$$cs(m) = \begin{cases} p(1,m) = ((x(1), y(1)), m) \\ p(2,m) = ((x(2), y(2)), m) \\ p(3,m) = ((x(3), y(3)), m) \\ \dots \\ p(n_m, m) = ((x(n_m), y(n_m)), m) \end{cases}$$

The above segmented procedure is demonstrated by Fig.1 (a)-(b). In successive sections, we suppose that an input image  $I$  has been segmented into  $I_s$  and  $I_s$  is represented as two arrays  $CS$  and  $P$ .

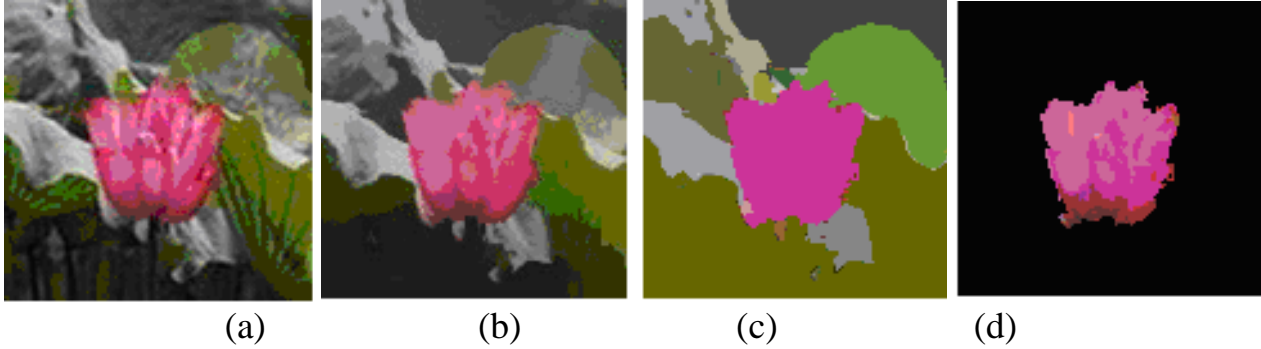


Figure 1: A process segmenting an image and extracting a single object from the image, (a) original image; (b) smoothed image; (c) segmented image; (d) extracted single object.

### 3. Construct the shape curve of the object

In this section, our purpose is to construct the shape curve of a single object. This process is organized as follows.

#### 3.1 Extract the boundary points of object

Suppose that input image  $I$  has been segmented by the above algorithm and its corresponding segmented image is  $I_s$ , and  $I_s$  has the form given by the above section, i.e.,  $I_s$  has been represented as the two arrays  $CS$  and  $P$ ,  $CS = \{cs(i)\}_{i=1}^m$  and  $p = \{p(j,i) | i=1, \dots, m; j=1, \dots, n_m\}$ . Based on the form of  $I_s$ , furthermore, we can give the method extracting the boundary of objects as follows.

For each cluster in  $I_s$ , i.e.,  $cs(i)$ ,  $i = 1, \dots, m$

- Make a dark copy of original image  $I$  and denote the copy as  $I_b$ ;
- Superimpose  $cs(i)$  onto  $I_b$ , i.e., assign the color of cluster  $cs(i)$  to pixel of  $I_b$  corresponding to location  $p(j, i)$ , i.e.,  $\{I_b(p(j, i)) = c(i)\}_{j=1}^{n_i}$ ;
- Use an edge-detection algorithm such as Sobel or region growing algorithm to detect the boundary of object  $cs(i)$ ;
- Extract the points of the boundary of object  $cs(i)$  and store them into array  $B(i)$ .

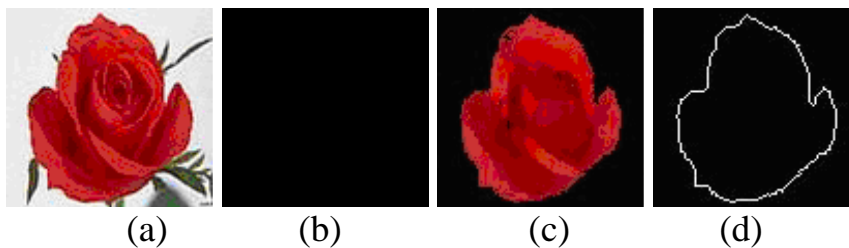


Figure 2: The process extracting the boundary points of object, in which the single object is a rose. (a) An input image  $I$ ; (b) The dark copy of  $I$ ,  $I_b$ ; (c) The synthesized image  $I_e$ ; (d) The edge image of single object,  $I_r$ .

This procedure is shown in Fig.2, in which the single object is a rose flower. Fig.2(a) is an input image  $I$ . Fig.2(b) is the dark copy of  $I$ ,  $I_b$ . Fig.2(c) is the image that the cluster of rose is superimposed on  $I_b$ ,  $I_e$ .

Fig.2(d) is the result that the edge of object is extracted from  $I_e$ ,  $I_r$ . Furthermore, we extract the boundary points of the rose from  $I_r$  and store these points into array  $B(i)$ .

### 3.2 Establish 2d coordinate system based on a single object

In section 3.1, the boundary points of the  $i$ th object of input image  $I$  has been extracted and stored into array  $B(i)$ , and let  $B_i = \{p_i(x(j), y(j)) | j = 1, \dots, n_i\}$ ,  $i$  is the index of a single object of segmented image  $I_s$ ,  $j$  the index of boundary points and  $n_i$  the number of pixels in the single object. Our purpose is to map boundary points into the 2d plane and construct the shape curve of a single object to match two single objects. To achieve the shape feature of a single object, the distance from the center to each boundary point is the most efficient representation of the shape feature of the single object. Furthermore, in order to calculate these distances, a 2d coordinate system based on the single object must be established. For such a coordinate system, it is key to decide its origin. It is intuitive to keep the  $x$  and  $y$  axis the same as the old system. The formula for calculating the origin is given by

$$x_o = \frac{1}{mn} \sum_{x=1}^m \sum_{y=1}^n f(x, y)x \quad y_o = \frac{1}{mn} \sum_{y=1}^m \sum_{x=1}^n f(x, y)y \quad (4)$$

where  $(x, y)$  is point of edge image  $I_r$  of the  $i$ th object  $cs(i)$ ,  $(x_o, y_o)$  is the origin coordinate to calculate, the function  $f(x, y)$  is defined by

$$\begin{aligned} f(x, y) &= 1 & (x, y) \in B(i) \\ f(x, y) &= 0 & (x, y) \notin B(i) \end{aligned} \quad (5)$$

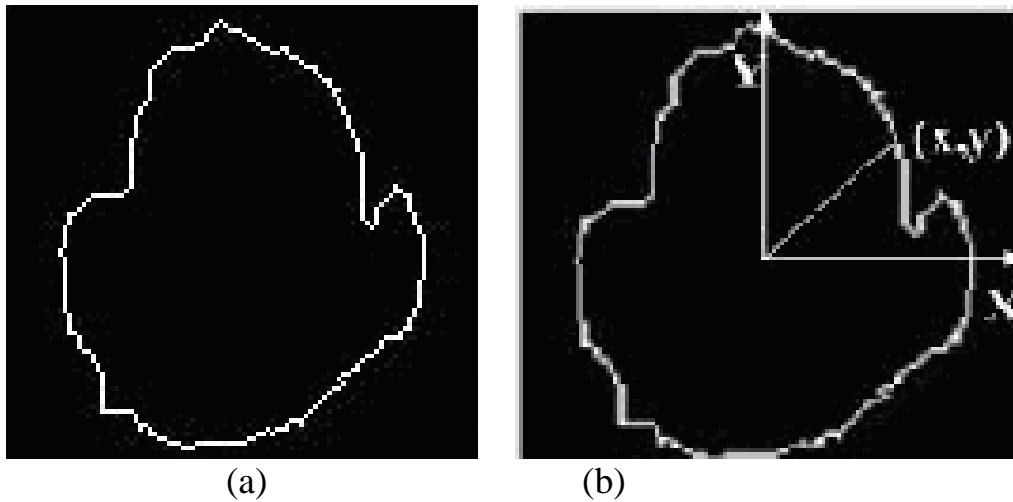


Figure 3: Establish 2d coordinate system for a single object, (a) edge image of the  $i$  object,  $I_r$ ; (b) edge image with 2d coordinate system for the  $i$  object.

### 3.3 Calculate index-distance pairs of boundary points of the object

In the new coordinate system, we traverse all boundary points of the object counterclockwise and calculate the Euclidean distance from the origin to the current boundary point. Suppose  $B_i = \{p_i(x(j), y(j))\}_{j=1}^{n_i}$  is the set of boundary points of the  $i$ th object,  $n_i$  the number of boundary points, and  $j$  is the index of the  $j$  point of  $B_i$  counting counterclockwise.  $j = 0$  only if the boundary of object intersects with the positive direction of the  $x$  axis. The distance from  $p_i(x(j), y(j))$  to the origin  $(x_o, y_o)$ , denoted by  $d_i(j)$ , is calculated by

$$d_i(j) = \sqrt{(x(j) - x_o)^2 + (y(j) - y_o)^2}, j = 1, \dots, n_i \quad (6)$$

Traversing through the whole boundary of  $B_i$ , a sequence of index-distance pairs is yielded,  $Bd_i = \{(j, d_i(j))\}_{j=1}^{n_i}$ . We map the sequence  $Bd_i$  into 2d real space  $R^2$  and let  $j$  be the horizontal axis and  $d_i(j)$  the vertical axis. The plot of the sequence is a few discrete points on the 2d plane, shown as Fig.4 (a).

### 3.4 Generate shape curve of single object

Set the index  $i$  to be the independent variable and  $d_i$  the dependent variable. It is straightforward that a order  $n$  polynomial is the optimum fit to the sequence of index-distance pairs, i.e, a function  $f(x)$  is defined by

$$f(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n \quad (7)$$

where  $f(x)$  is a continuous function in 2d real space and its geometry sense is a expansion of distances from origin to boundary point, the horizontal is corresponding to the index of each boundary point. As Fig.4(b) shows, the curve of polynomial function  $f(x)$  is plotted with a red line.

With respect to the potential complexity of the boundary of the object, specially when the number of boundary points is large, it is possible to takesuch a mechanism that boundary points are divided into different subsets, and for each subset a fit is made piecewise to reduce the order of the polynomial and alleviate instability in the high order polynomial. In order to simplify the description of the method, in this paper we have supposed that the boundary of the object is simple. Such a hypothesis does not give rise to a change of the basic framework extracting and expressing the shape curve of an object.

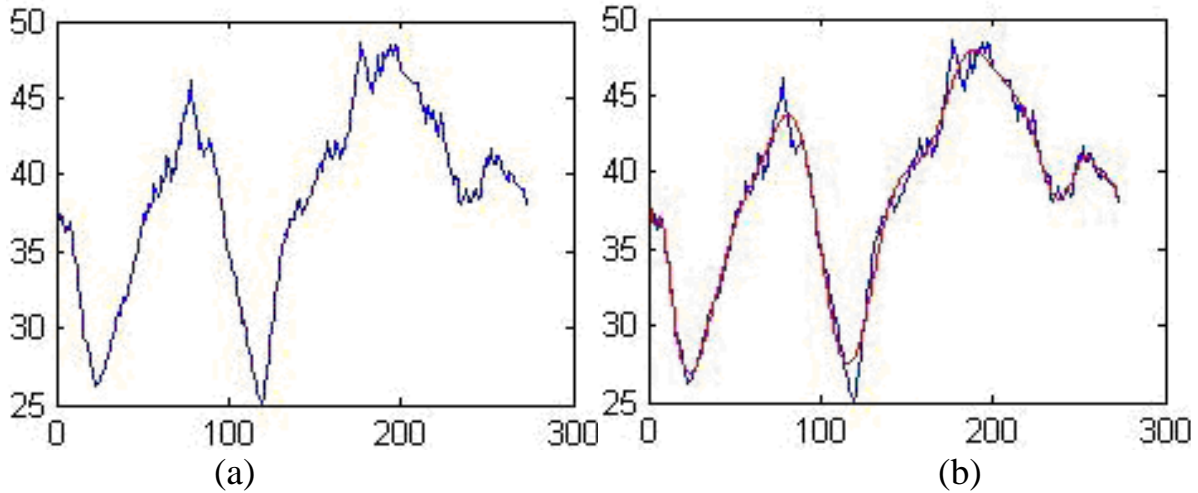


Figure 4: The process generating the shape curve of object, (a)plot of the boundary points of object rose; (b)the shape curve fitting to the boundary points of object rose, labelled with a red-line.

## 4. Preserve transformation invariance

Given an object, we have constructed its shape curve function. However, it is possible that the same natural object is translated, scaled or rotated when it is present in another image. In order to ensure the proposed method correctly matches the two objects, transformation invariance must be considered. In the next subsections, analysis and implementation are given with respect to the most common three transformations.

### 4.1 Transformation invariance of translation

In our method, the shape curve of an object is generated by the distances from the origin to boundary points. However, the transformation of translation preserves invariance of relative distance between two points in the same image. So, it is obvious that the proposed algorithm preserves transformation invariance of translation.

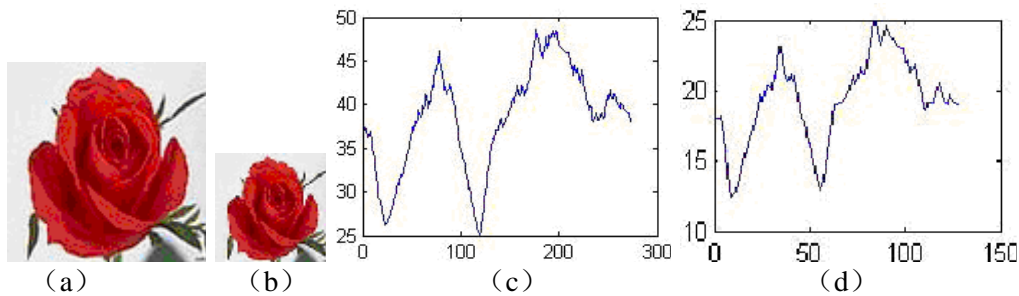


Figure 5: Scale transformation, (a) original object rose; (b)object rose scaled by 1/2 along both directions of  $x$  and  $y$ ; (c) plot of boundary points of the original rose (a); (d) plot of boundary points of the scaled rose (b);

## 4.2 Transformation invariance of scale

In comparison with translation, scale transformation of an object causes a change of distances from origin to boundary points. So, this transformation cannot exhibit invariance in the shape curve of an object. As Fig.5 shows, (a) is the original rose image, (b) the rose image scaled by along both directions of (a), (c) and (d) the corresponding plots of (a) and (b). Fig.5 shows that the two shape curves of (a) and (b) are very different. In order to ensure the transformation invariance of scale in (a) and (b), the shape curve with respect to Fig.(5)(d) must be transformed by

$$\begin{pmatrix} j' \\ d_i(j)' \end{pmatrix} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \begin{pmatrix} j \\ d_i(j) \end{pmatrix} \quad (8)$$

Where  $X = (j, d_i(j))^T$  and  $X' = (j', d_i(j)')^T$  denotes points of in the 2d plane mapped into by the original image and the scaled image respectively. As Fig.5 shows, let  $n_t$  and  $n_s$  be the number of boundary points of the original object and the scaled object respectively.

According to the property of geometry transformation, it is easy to obtain the formula computing the parameter  $\alpha$  and  $\beta$  as follows

$$\begin{aligned} \alpha &= \frac{n_t}{n_s} \\ \beta &= \frac{\overline{\mu_t}}{\overline{\mu_s}} \end{aligned} \quad (9)$$

where  $\overline{\mu_t}$  is average distance from boundary points to the origin of the original object and  $\overline{\mu_s}$  that of the scaled object. Equation (8) can be used to ensure the transformation invariance of scale between two objects.

## 4.3 Transformation invariance of rotation

As Fig.6 shows (a) is the original rose, (b) is it rotated by 90 degrees with respect to (a). (c) and (d) are the corresponding plots of (a) and (b). Fig.6 shows that the two shape curves of (a) and (b) have significant differences. Comparing (c) with (d), it is found that if this part of the curve over interval  $[0, y]$  of (c) is shifted to its tail, the shape of curve is approximately similar to that of (d). The value  $y$  is referred to as the phase-offset of (d) to (c). Therefore, in order to exhibit transformation invariance of rotation between two objects, such as (a) and (b), a method must be designed. Obviously, it is a key for designing the method to find an appropriate phase-offset  $y$ . An algorithm for finding the phase-offset  $y$  is presented as follows.

Let  $f(x)$  be the shape curve of the original object and  $g(x)$  be that of the rotated object, the number of boundary points of original object be  $n$ , and  $D$  be the distance between two shape curves of  $f(x)$  and  $g(x)$ , which is calculated by the 2-norm  $\|f(x) - g(x)\|_2$ .

$$D(t) = \int_0^{n-t} |f(x+t) - g(x)|^2 dx + \int_{n-t}^n |f(x+t-n) - g(x)|^2 dx \quad (10)$$

Therefore, the phase-offset  $y$  between  $f(x)$  and  $g(x)$  is computed by minimizing the distance  $D(t)$ , i.e.,  $y = \arg \min \{D(t)\}$ , in other word,  $y$  is the solution of a equation below.

$$\frac{\partial}{\partial t} D(t) = 0 \quad (11)$$

Since both  $f(x)$  and  $g(x)$  are polynomials, the integral  $D(t)$  is a high order polynomial with respect to  $t$ , and it is easy to compute the global minimum, the  $t$  minimizing  $D(t)$  is the phase-offset  $y$  of (d) with respect to (c).

With (10) and (11), transformation invariance of rotation can be ensured in a practical application.



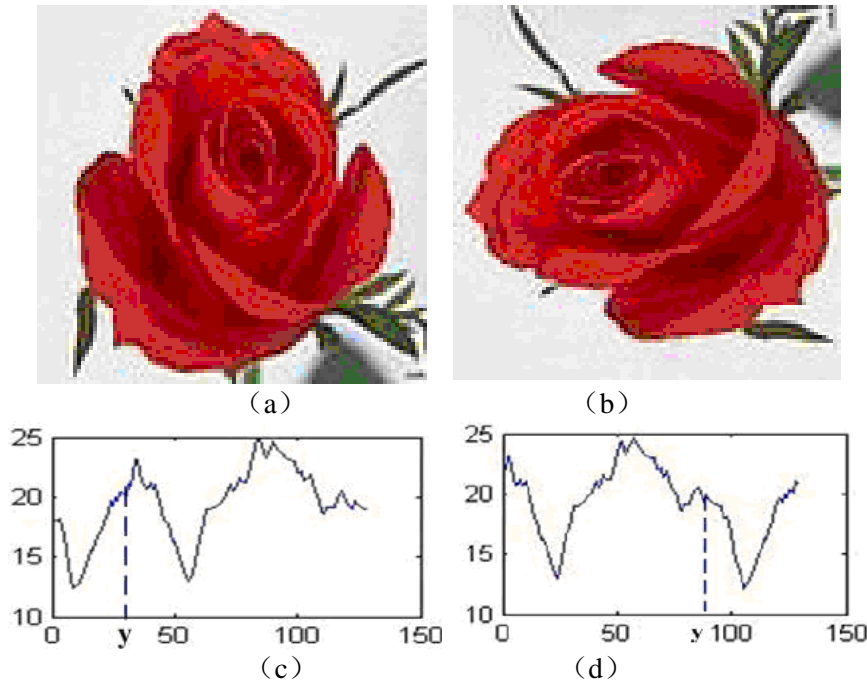


Figure 6: Rotation transformation, (a) original object rose; (b) rotated object rose; (c) plot of boundary points of the original rose (a); (d) plot of boundary points of the rotated rose (b)

## 5. A similarity measure between two images

In order to increase the precision of image retrieval, we incorporate both color and shape features for matching between two images. In other words, we seek such an image from database, which is similar to the input image with respect to both color and shape. Furthermore, based on user requirements, a user should be able to make a choice and make a decision to place the emphasis on either color or shape. Therefore, the definition of the new similarity measure should be able to ensure such flexibility. An optimum method to ensure flexibility is to use weighted factors. So, the basic idea of our similarity measure is given as follows.

The article is based on the nature of curvature OI extracted feature points on the contour curve. Curvature as the inner nature of the pixel, can well meet the shape of the displacement theory of invariance and rotation invariance requirements. We usually think that the main shape of the object changes in the information concentrated in the direction where the fastest, that is the extreme curvature points. This article first adjacent pixels based on the Euclidean distance between the contour curve to calculate the support of each point on the support area and radius, with linear interpolation of the endpoints to support regional. And then seek support for regional centre of mass to points on the curve and its corresponding support regional centred distance between the candidate selected as a characteristic feature point response. Experiments show that using this algorithm to extract feature points redundancy is small; the shape of the object to maintain high accuracy, the curve also has high rotation invariance.

### 5.1 Determining for improved support regions

Set the contours curve of object as  $C$ . After Digital processing, it is expressed as an integer point out in the plane:  $C = \{p_i = (x_i, y_i) | i = 1, 2, \dots, n\}$   $p_i$  and  $p_{i+1}$  are both with each neighbor, and it also satisfy the formula as  $|x_{i+1} - x_i| \leq 1$  and  $|y_{i+1} - y_i| \leq 1$  besides the same time satisfying zero. For an arbitrary point on the curve  $C$  that defines the support region as:  $\Omega(p_i) = \{p'_{i-m}, p_{i-m+1}, \dots, p_{i-1}, p_i, p_{i+1}, \dots, p_{i+n-1}, p'_{i+n}\}$  And  $p_{i-m+1}, \dots, p_{i-1}, p_i, p_{i+1}, \dots, p_{i+n-1}$  are all points on the curve. And  $p'_{i-m}$  and  $p'_{i+n}$  are two end points on the support radius, but they are not necessarily the point on the curve. Defining  $\|p_i p_{i+j}\|$  as the line length between two points on the curve:

$$\|p_i p_{i+j}\| = \sum_{k=1}^j \sqrt{(x_{i+k} - x_{i+k-1})^2 + (y_{i+k} - y_{i+k-1})^2} \quad (12)$$



By (12) can be obtained to support the region's two endpoints  $p'_{i-m}$  and  $p'_{i+n}$ . If the type can be accurately established then two end points  $p'_{i-m}$  and  $p'_{i+n}$  is the point on the curve  $C$  (with  $p'_{i-m}$  and  $p'_{i+n}$  overlap), or by linear interpolation can be used to strike the following formula:

$$p'_{i+n} = \mu p_{i+n-1} + (1 - \mu) p_{i+n}, 0 < \mu < 1 \quad (13)$$

Where  $\|p_i p_{i+n-1}\| < L$ ,  $\|p_i p_{i+n}\| > L$ ,  $\mu = \frac{(\|p_i p_{i+n}\| - L)}{\|p_i p_{i+n}\|}$  Can be obtained endpoint  $p'_{i-m}$  using similar methods. While this situation  $p'_{i-m}$  and  $p'_{i+n}$  not the points on the curve  $C$ , but in the subsequent calculations will them as pixels. Above the equation (12) can be met accurately.

## 5.2 Contour feature point detection

There are two steps in detect feature: First Connecting  $\Omega(p_i)$  two points on all the adjacent form a closed polygon  $G$ , Let  $G$  be the center of mass  $g$ , and then, each point on the curve of the contour using the above method to determine its support of regional, The length parameter is usually set  $n/40 \sim n/10$ ,  $n$  is the total number of curve points columns. Between the  $p_i$  and  $g_i$ , then the characteristics of strength by the following formula

$$d(p_i, g_i) = \sqrt{(x_i - \frac{1}{m+n+1} \sum_{k=-m}^n x_{i+k})^2 + (y_i - \frac{1}{m+n+1} \sum_{k=-m}^n y_{i+k})^2} \quad (14)$$

$x_{i+k}$  and  $y_{i+k}$  is Support the coordinates of points in the region.  $x_{i+n}$ ,  $y_{i+n}$  and  $x_{i-m}$ ,  $y_{i-m}$  indicate both ends of the point coordinates in support of regional. By (12) obtained Feature pixels greater than the threshold strength as a candidate feature points, threshold can be set to  $L/5 \sim L/2$  (14) a point on the curve reflects the degree of bending, The greater the value the greater curvature of the point. So the basic idea is to use the algorithm on the image contour is calculated as all points on the curve with their respective support areas as the distance from the centroid point of the characteristics of strength, greater than a certain threshold of points as the candidate of the feature points.

Finally, the application of non-maximum suppression Points along the curve of the order of columns, Detected with a maximum intensity value of pixels as the final feature points, Set point to be detected  $p_i$ , Detection region is  $\{p_{i-k}, \dots, p_{i-1}, p_i, p_{i+1}, \dots, p_{i+k}\}$ ,  $k$  can take 10.

## 5.3 Image similarity

A given target image and retrieve image, as defined similarity algorithm

Step-1. The outline of the image curve into discrete ordered sequence of points;

Step-2. Sequence of ordered point scale transformation, and retrieval of the target sequence at the point on the interval and the mean value from one to one

Step-3. Extract the target image feature points on the contour, the  $i$ -th feature point as  $p_i$ ,  $1 \leq i \leq n$

Step-4. To find and retrieve the image outline on the outline of the target image pixels corresponding to position  $p'_i$ ,  $1 \leq i \leq n$

Step-5. Calculate the shape differences between the two images degrees, with Step-3 Step-4 with the corresponding pixel in the Euclidean distance Indicated  $s_i = \sum_{i=1}^n \|p_i - p'_i\|_2$

Step6. Will point to retrieve an ordered sequence of contour images rotate right, repeat Step-4 and Step-5 up through all the pixels image retrieval;

Step-7.  $S$  were normalized for each treatment, defined as the similarity of two images  $s_i = 1 - \min_{1 \leq i \leq n} s_i$

## 6. Experiments

To verify the validity and accuracy of the proposed method, we evaluate the method from two aspects, transformation invariance and precision of image retrieval. Moreover, we also compare the method with two of the most common algorithms.

### 6.1 Verify the transformation invariance of an object

In this section, our purpose is that for invariance of scale and rotation transformations, we test the validity of our method. Experimental target is to observe the consistency of the shape curves of objects before and after a transformation. As Fig.7 shows, two different objects, a rose and a butterfly, are selected and tested under scale and rotation transformations. Fig.7(a) is an image containing the original object rose and Fig.7(d) an image containing the original object butterfly. Fig.7(b) and (e) are two corresponding images obtained by rotating the original objects of (a) and (d) respectively. Fig.7(c) and (f) are two corresponding images obtained by scaling the original objects of (a) and (d) respectively.

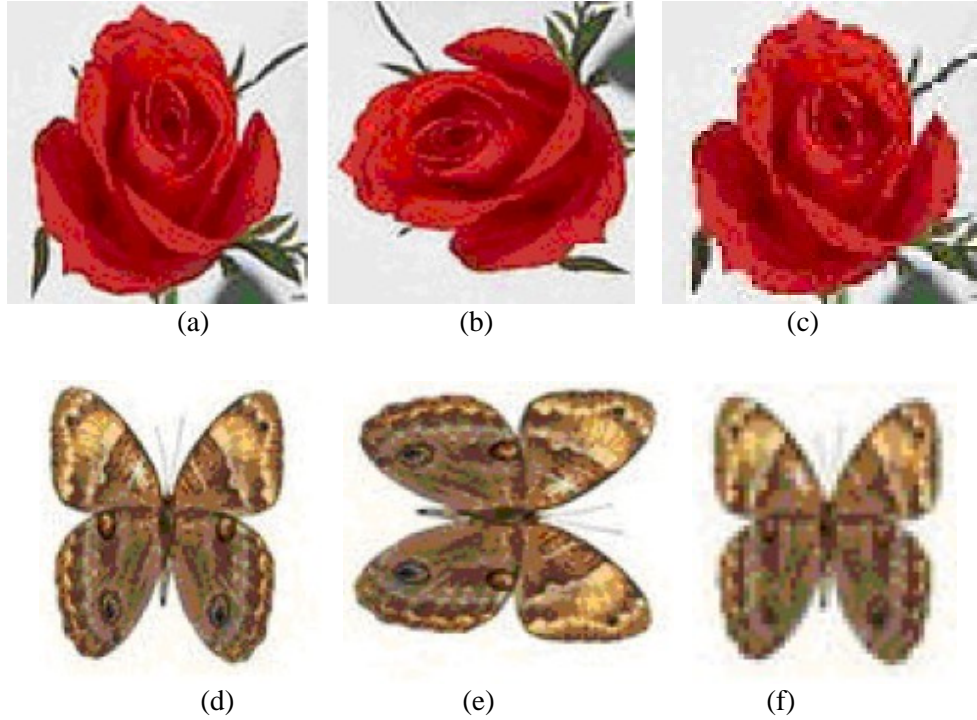


Figure 7: Verify the transformation invariance, (a)Original object rose; (b) Rotated rose; (c)Scaled rose; (d)Original object butterfly; (e)Rotated butterfly; (f)Scaled butterfly.

Fig.8 shows that our method is used to extract the shape curves of objects under scale and rotation transformations. For object rose and object butterfly, Fig.8(a) and (b) are the plots of their shape curves under scale and rotation transformations. For the curves of Fig.8 (a) and (b), the blue line represents the shape curve of original object, the green line one of the rotated object and the red line one of the scaled object. As Fig.8 shows, experimental results display that for the same object the shape curves of transformed and original objects are approximately identical.

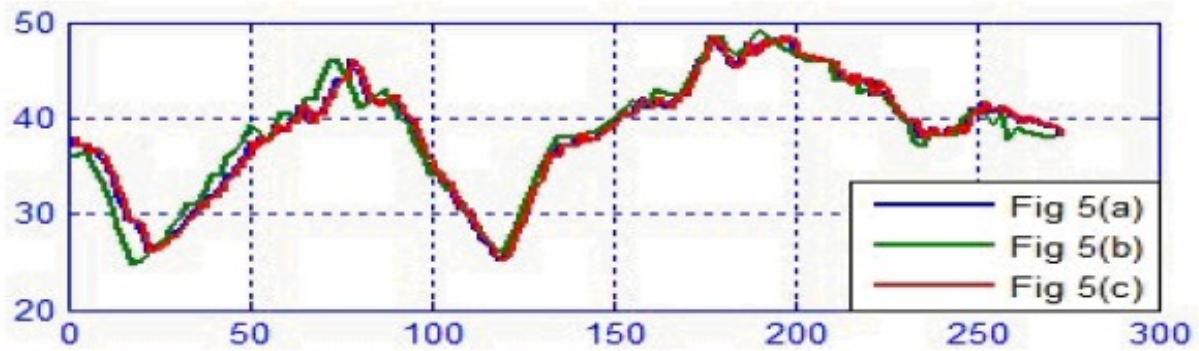
### 6.2 Experimental Analysis

In this paper, an object-based image retrieval system is designed. It includes a database of about 1000 images. In order to more effectively organize and analyze the experiments, these images are divided into 50 categories so that each category contains about 20 images. Moreover, the two assessment indicators, recall and precision, are defined and used for benchmark comparison.

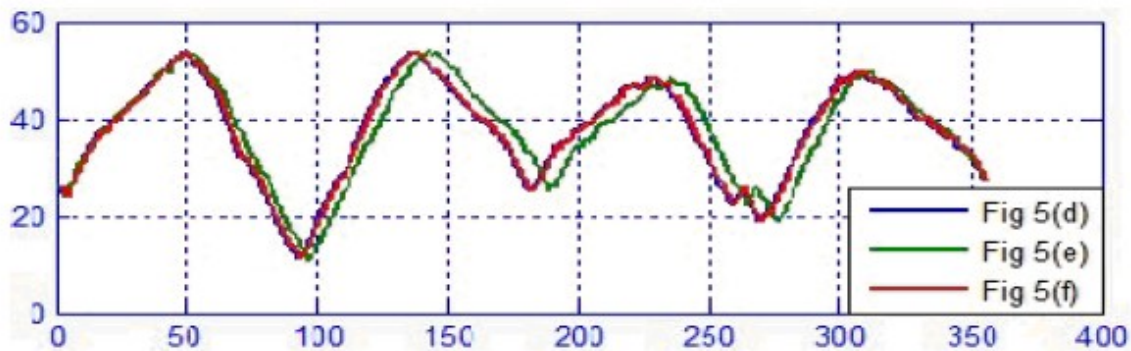
#### 6.2.1 Define assessment indicators

All images in the database are prior-labeled with respect to category by hand and let those images in the same category be  $C_i (i = 1, \dots, m)$  where  $i$  representing the  $i$ th category. Suppose that an input image  $I$  belongs to the  $i$ th category, i.e.,  $I \in C_i$ , the result set returned by retrieving  $I$  is  $C_{ir}$ , and set  $S_i = C_i \cap C_{ir}$ . Then, the two assessment indicators, recall and precision, can be calculated by

$$R = \frac{|S_i|}{|C_{ir}|} \quad P = \frac{|S_i|}{|C_i|} \quad (15)$$



(a)



(b)

Figure 8: (a)the shape curve of all transformations for original object rose; (b)the shape curve of all transformations for original object butterfly.

### 6.2.2 Experimental comparison

To qualitatively evaluate the advantage of the proposed method, two typical relevant algorithms[16,17] are selected to compare with the method. The method in [17], in which an image is considered as a set consisting of a number of color blocks and a similarity is computed with respect to the radius histogram and the angle histogram for each color block, is referred to as RAH. The method in [17], which applies the feature point to represent the color information of an image, is referred to as HCH. The purpose of the experiment is to evaluate the two indicators, recall and precision, among the three methods. For each method, we randomly select the same image categories and take eight images from each category. For each result set of images returned by a retrieval, the number of the result set is set to 16. Using the system, 240 experiments were carried out. Fig.9 shows some experimental results.

As Fig.9 shows, the experiments display that the proposed method has a better effect than the other two algorithms for the two assessment indicators. Table 1 shows the result of comparison for recall and precision with respect to the three algorithms.

Furthermore, analyzing these datum in Table 1, the recall is 10% more than that of the HCH algorithm and 8% more than that of the RAH algorithm. To evaluate the relation between the number of images in result set to the two indicators, the number is changed from 16 to 50. Average recall and precision are calculated and plotted in Fig.10.



Figure 9: The three image retrieval methods are used to retrieve an image by our system, (a), (b) and (c) are the result set for Original object butterfly; (d), (e) and (f) one for Original object five-stars; (g), (h) and (i) one for Original object rose.

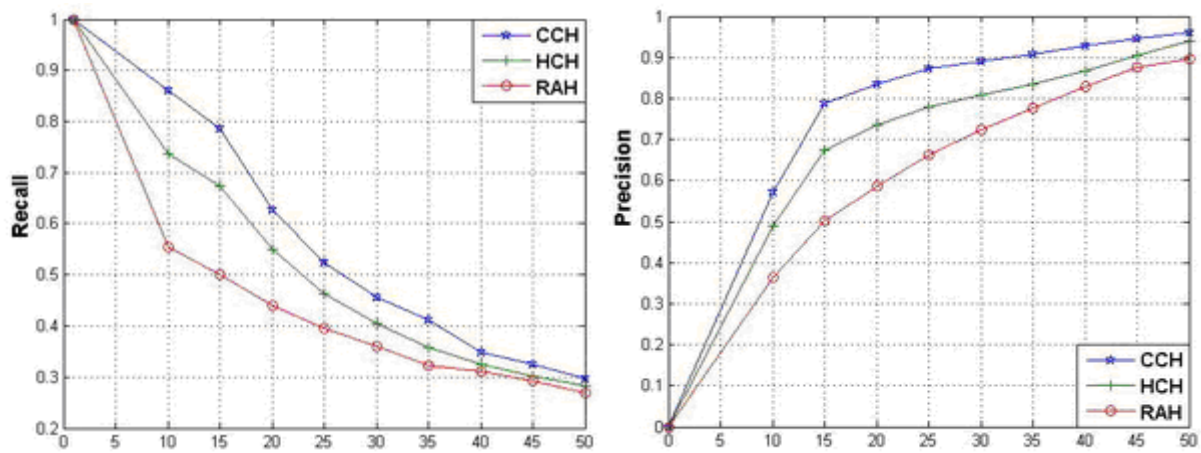


Figure 10: The trend that the two indicators vary with the number of images in result set.

Table 1: Assessment of recall and precision with respect to three algorithms CCH HCH RAH

Retrieval object category	CCH		HCH		RAH	
	R	P	R	P	R	P
Obj <sub>1</sub>	0.5500	0.5867	0.4375	0.4670	0.5250	0.5600
Obj <sub>2</sub>	0.8375	0.8930	0.2875	0.3070	0.5500	0.5867
Obj <sub>3</sub>	0.5125	0.5467	0.3050	0.3200	0.4750	0.5067
Obj <sub>4</sub>	0.8500	0.9070	0.4125	0.4400	0.7000	0.7467
Obj <sub>5</sub>	0.9370	1.0000	0.9370	1.0000	0.9370	1.0000
Obj <sub>6</sub>	0.8150	0.7562	0.8150	0.7562	0.7500	0.7264
Obj <sub>7</sub>	0.6875	0.5000	0.4000	0.5000	0.5625	0.5000
Obj <sub>8</sub>	0.9375	0.6522	0.6250	0.4167	0.8125	0.5417
Obj <sub>9</sub>	0.8125	0.5910	0.8125	0.5910	0.6875	0.5000
Obj <sub>10</sub>	0.7500	0.5715	0.6250	0.4762	0.6875	0.5239

## 7. Conclusion

In this paper, a novel image retrieval method is proposed, which applies not only the color feature but also the shape feature of objects. For the color feature expression, we present a new histogram, which uses the distance between the average color of principal object and each pixel color in an image. This idea has two advantages: it reflects the general idea of the traditional histogram, and in the meantime, it has the property of the color spatial distribution of objects. On the other hand, the method implements the extraction of the shape feature of an object so that direct match between objects is available and effective. Moreover, since in practice the match between two images considers not only the color feature but also the shape feature, intuitively the similarity integrating both color and shape should have a higher flexibility. In fact, experiments also show that our method has a significant advantage over the other two methods.

In future work, we intend to investigate the affine invariance and other invariances to solve the match between deformable objects.

## 8. Acknowledgements

This work was supported in part by the National Science Foundation of China under Grant 60973157.

## 9. References

- [1] M. Swain and D. Ballard. "Color indexing". *Int.Journal of Computer Vision*, 7(1), pp. 11-32, 1991.
- [2] X. Y. Wang; J. F. Wu and H. Y. Yang, "Robust image retrieval based on color histogram of local feature regions ", Springer Netherlands, pp.1573-7721, 2009.
- [3] C. L. Novak and S. A. Shafer. "Anatomy of a color histogram. *Computer Vision and Pattern Recognition*". Proceedings CVPR '92. IEEE Computer Society Conference on 15-18, 1992. pp. 599 - 605, June 1992.
- [4] E.A.Bashkov and N.S.Kostyukova. "To the Estimation of Image Retrieval Effectiveness Using 2D-color Histograms". *Journal of Automation and Information Sciences*, 38(11), pp.84-89, 2006.
- [5] Y. Cheng. "Mean shift, mode seeking and clustering". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 17(8), pp.790-799,1995.
- [6] D.Comanicui and P. Meer. "Mean shift: a robust approach toward feature space analysis". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 24(5), pp. 603 - 619, 2002.
- [7] A. Miguel and C. Perpinan. "Gaussian Mean-Shift Is an EM Algorithm. *IEEE Trans. Pattern Anal. Mach. Intell*, 29(5), pp. 767-776, 2007.
- [8] Y. M. Liang; H. C. Zhai and P. Chavel. "Fuzzy color-image retrieval".*Optics Columniations*, 212(4/6), pp. 247 250, 2002.



- [9] H. Thomas; Y. Gousseau and F. Schmitt. "Adaptive image retrieval based on the spatial organization of colors". Computer Vision and Image Understanding, pp. 101-113, 2008.18
- [10] W. B. Chen; Q. Z. Li and J. W. Yang. "Image Retrieval Based on Composite of Mean Shift and Assignment Model". Fifth International Conference on Information Assurance and Security, pp. 568-571, 2009.
- [11] R. O. Plotze; M. Falvo and J. G. Pdua. "Leaf shape analysis using the multiscale minkowski fractal dimension, a new morphometric method: A study with passiflora (Passifloraceae)". Botany 83(3), pp. 287-301, 2005.
- [12] M. A. Pazoti; R. E. Garcia and J. D. C. Pessoa. "Comparison of shape analysis methods for guinardia citricarpa ascospore characterization". Electron. J. Biotechnol, 8(3), pp. 267-275, 2005.
- [13] A. Huertas and G. G. Medioni. "Detection of intensity changes with subpixel accuracy using Laplacian-Gaussian masks". IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(5), pp. 651-664, 1986.
- [14] M. Kass; A. Witkin and D. Terzopoulos. "Snakes: Active contour models". Int. J. Comput. Vision, 1(4), pp. 321-331, 1988.
- [15] J. Batista and R. Freitas. "An adaptive gradient-based boundary detector for MRI images of brain". Seventh International Conference on Image Processing and its Applications, pp. 440-444, 1999.
- [16] W.T. Wong; Y. S. Frank, J. Liu. "Shape-based image retrieval using support vector machines, Fourier descriptors and self-organizing maps". Information Sciences, pp. 1878-1891, 2007.
- [17] J. G. Zhang and T. N. Tan. "Affine invariant classification and retrieval of texture images". Pattern Recognition, pp. 657-664, 2007.19