Plant Species Identification Using Leaf Image Retrieval

Carlos Caballero
Department of Languages and Computer Science
Engineerings School
C/ Doctor Ortiz Ramos s/n, 29071 Málaga, España (Spain)
carloscg@uma.es

M. Carmen Aranda
Department of Languages and Computer Science
Engineerings School
C/ Doctor Ortiz Ramos s/n, 29071 Málaga, España (Spain)
mcarmen@lcc.uma.es

ABSTRACT

In this paper, an effective shape-based leaf image retrieval system is presented. A new contour descriptor is defined which reduces the number of points for the shape representation considerably. This shape representation is based on the curvature of the leaf contour and it deals with the scale factor in a novel and compact way. A two-step algorithm for retrieval is used. In a first step, the database is reduced using some geometrical features. Then a similarity measure between the contour representations is used to rank conveniently leaf images on the database. We implemented a prototype system based on these features and performed several experiments to show its effectiveness for plant species identification.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

Content-based image retrieval, leaf image database, plant species identification, shape matching

1. INTRODUCTION

Many researches have proposed techniques for content-based image retrieval using image features such as color, shape, texture, and spatial relationship. Nevertheless, if images contain similar color or texture, shape-based image retrieval is more effective than other approaches. For instance, leaves of most plants are green or brown; but the leaf shapes are distinctive and thus can be used for identification. So, the shape of plant leaf is one of the most important features for characterizing various plant species.

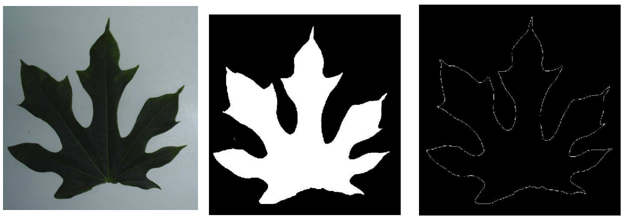
Shape-based image retrieval is composed, like typical content-based image retrieval, of two main steps. The first one is to represent shapes in such a way that it is invariant to translation, rotation, scale, and viewing angle changes. The other step is shape matching that determines how similar shapes are to a given query image.

In general, there are two basic approaches for shape representation: by contours and by regions. The former describes a region of interest using its external characteristics, while the latter represents a region of interest using its internal characteristics. Examples of region based shape representation are octrees, quadtrees, skeletons and morphology decomposition. Polygonal approximation, chain code, geometric primitives, parametric curves, Fourier descriptors and Hough transform are contour based shape representation methods.

Another important issue of shape-based image retrieval is the shape matching method on which the retrieval performance heavily depends. Usually the similarity measure developed for recovery depends on the representation method used. A shape-based image retrieval system can be developed to correctly discriminate and recognize leaf shapes of different species. Some research work has been done on this problem. Rui et al. [8] proposed a modified Fourier Descriptor (MDF) method to achieve translation, scaling and rotation invariance by considering the distance between the FD magnitude and the phase angle separately so as to decrease the discrimination noises. Yankov et al. [12] explored the metric properties of the rotation invariant distance measures and proposed an algorithm for fast similarity searching in the shape space. Xi et al. [11] introduced a new algorthm that is two to three orders of magnitude faster than brute force search to the problem of discovering shape motif, which are approximately repeated shapes within image collections.

Abbasi et al. [1] and Moktarian et al. [5] proposed a curvature scale space (CSS) image to represent leaf shapes for Chrysanthemum variety classification. Mokhtarian and Abbasi [4] improved the CSS method and applied it to leaf classification with self-intersection. Wang et al. [9, 10] described a method which combines different features based on a centroid-contour distance curve, and adopted a fuzzy integral for leaf image retrieval. In [2] the authors propose a new classification method based on hypersphere classifier based on digital morphological feature.

In this paper we develop a shaped based image retrieval system for plant species identification based on two important issues: first, a new shape representation is presented which gets a multiscale representation of the contour in a compact way; and second, it uses both types of descriptors, contour based and region based to build a two-step recovery system that allows to speed the matching process.



(a) (b) (c)

Figure 1: (a) original image (b) segmented image (c) contour of the image.

The rest of this paper is organized as follows. Section 2 describes the preprocessing techniques performed on the image. Section 3 describes shape representation methods. Section 4 presents how to perform image matching and retrieval. In section 5, some of the experimental results are presented and finally the last section concludes the paper and discusses some future work.

2. LEAF IMAGE PREPROCESSING

The colors of plants are usually green. Moreover, the shades and the variety of changes of water, nutrient, atmosphere and season can cause change of the color, so the color feature has low reliability. Thus, we decided to recognize various plants by the gray-level image of plant leaf, while ignoring the color information. As a result, only gray component for each pixel is computed from the color image.

The region of interest (ROI) of a leaf image should include all the pixel their gray values are smaller than a specific threshold. In our study the threshold is automatically selected according to the histogram of the leaf gray image using the optimal threshold of Otsu [7]. Figure 1 shows the image of a leaf of the plant specie *Brachychiton acerifolius* (1(a)) and the binary image got after the segmentation (1(b)). Then the contour of the leaf can be extracted from the binary image (as shown in Figure 1 (c)) [3].

3. FEATURE EXTRACTION AND SHAPE REPRESENTATION

The goal of a good representation is to extract the minimal information capable of differentiating one object from another. As we mentioned before, shape representation is better suited to our problem. Shape descriptors are mainly based on contour or based on region. The last are classified into geometrical and topological. Geometrical descriptors measure aspects such as size, position, orientation, circularity, rectangularity, etc.

The leaf representation that we propose uses both geometrical and contour descriptors. They are described in the following paragraphs.

3.1 Geometrical descriptors

We have chosen some geometrical features that describe the inherent characteristics of the ROI to make a fast filtering database. The features used are:

- **Area:** This measure represents the ratio between the ROI area and the minimum bounding rectangle (MBR) area.

$$A = \frac{A_{ROI}}{A_{MBR}} \quad (1)$$

- **Eccentricity:** The eccentricity is defined as the ratio of the length of main inertia axis of the ROI (E_A) to the length of minor inertia axis of the ROI (E_B).

$$E = \frac{E_A}{E_B} \quad (2)$$

- **Diameter:** The diameter of a subset of a metric space is the least upper bound of the distances between pairs of points in the subset. So, if C is the contour of the ROI.

$$D = \max_{x,y \in C} \{d(x, y)\} \quad (3)$$

3.2 Contour representation

An extracted leaf contour often exhibits too many resolvable points, thus it is not suited to be directly applied to shape matching and the shape representations should be compressed. So, it is convenient to select a subset of points that represent the shape of the leaf and make efficient shape matching. These points are usually called dominant points, points of interest or characteristic points.

In our method the characteristic points are selected based on the graph of curvature values of the contour. Some attributes are associated to each point in order to achieve a final shape representation invariant to scale, rotation and translation.

3.2.1 Curvature of the Contour

There are psychological results showing that curvature plays a fundamental role in human shape perception. Although less methods perform curvature evaluation, this step is very important in contour description. The major difficulty is due to the discrete nature of the curve, making the majority of the methods noisy and scale dependent. We introduce the scale factor in the representation in a new and more compact way through a *lifetime* attribute associated to each point.

Let $C = \{(x_i, y_i)|i = 1, \cdots, n\}$ be the sequence of n points describing the contour of a leaf. We select a subset of shape characteristic points based on the graph of the curvature values $K = \{k_i|i = 1, \ldots, n\}$ computed from the digital curve C where k_i is the curvature value of (x_i, y_i). The continuous expression to calculate curvature values on each point of a parametric curve $\{x(t), y(t)|t \in R\}$ is

$$k = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \quad (4)$$

where ($\dot{}$), ($\ddot{}$) denote first and second derivatives respectively. To apply this expression to our discrete contour C we must control the spatial quantization noise. To minor the noise, we have estimated the curvature value at each point by using a Gaussian filter with scale σ, to smooth the quantization noise [6]. Let $K(\sigma)$ the graph of curvature calculated at the smoothing scale σ.

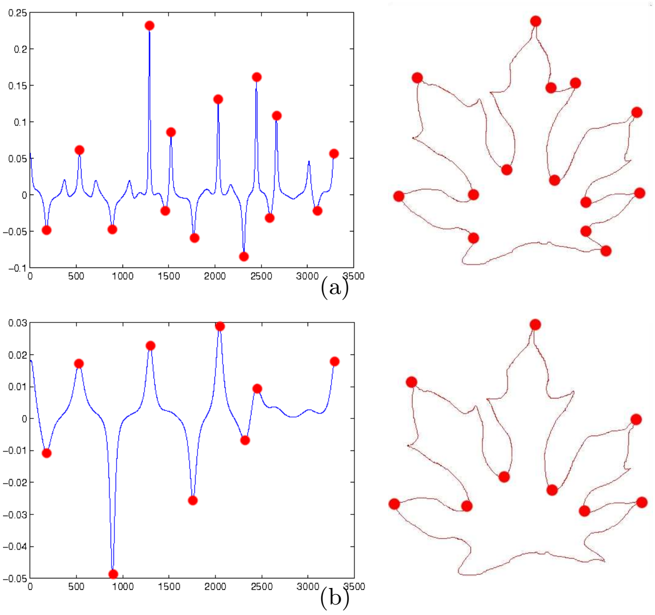


Figure 2: (a) Characteristic points for the scale 11 (b) for the scale 14.

3.2.2 *Shape representation: the characteristic points' vector (CPV)*

The set of characteristic points is selected as the local extremes of $K(\sigma)$. The smoothing scale σ is chosen so that the degree of smoothing eliminates the noise while retaining the details of leaf shape. The final shape representation consists on a vector which stores the following attributes for each characteristic point:

- **Value of curvature:** This value indicates the degree of curvature of the characteristic point.
- **Sign:** It indicates whether the point belongs to a convex or concave part of the curve.
- **Absolute distance:** Number of points of the contour between the characteristic point and the next one. This attribute is stored to make a shape matching depending on the scale of the object.
- **Relative distance:** Distance to the next point divided by the total length of the contour.
- **Lifetime:** This attribute reflects the existence of this point at different scales. Each characteristic point is recalculated to a value greater smoothing, this attribute indicates the smoothed levels where the characteristic point still appears. Therefore only the more representative points survive to larger scales.

  To estimate the lifetime it is necessary to calculate the curvature values of the digital curve C at several scales. Let $K(\sigma^1, \ldots, \sigma^L) = \{k_i(\sigma^j) | i = 1, \ldots, n\}$ the set of graphs of curvature values at the smoothing scale σ^j with L being the total number of scales. Local extremes are also calculated for each graph $K(\sigma^j)$.

  We consider that a characteristic point survives from one scale to the next if there is a local extreme near the position the former was detected. Then the lifetime of the point increases.

  Figure 2 (a) shows the histogram for the curvature function on the scale 11, in which there are a set of peaks that represent the most significant characteristic points, also shows a representation of these points on the contour of the image. Figure 2 (b) shows again the histogram and the image but with the scale 14 where it can be seen as some points are missing. This is due to these points have not exceeded more levels of smoothing and therefore these points are less relevant than those points that do remain.

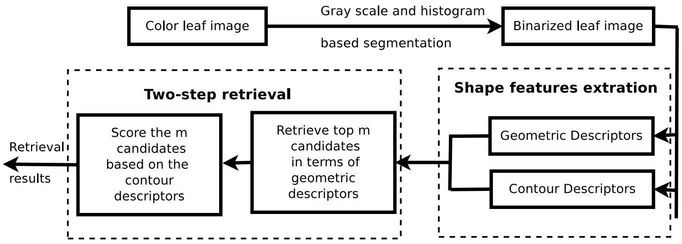


Figure 3: Block diagram of our two-step leaf image retrieval approach.

4. IMAGE MATCHING AND RETRIEVAL

The final step of image retrieval is image matching and browsing. In this section, we present an efficient matching method for obtaining ranks of all database images in an approximate order of similarity to the query image. It consists of a two-step retrieval. Firstly, geometrical descriptors are used for fast filtering database. Contour descriptor is then used for the assessment of similarity between two images. A general overview of the system is described in Figure 3.

4.1 Similarity between geometrical descriptors

Geometrical descriptors are used to reduce the database to a smaller one without losing relevant information. In this filtering stage, an image of the database is selected if the Euclidean distance between the value of each geometrical descriptor and the value of the corresponding descriptor for the query image is less than a threshold.

Let U and V be the query and database image, respectively. Let U_A be the value of the area descriptor for the query image and V_A the same value for an image of the database (the same for U_E and U_D). Then an image is selected if it meets that:

$$|V_i - U_i| < T_i \; \forall i \in \{A, E, D\} \tag{5}$$

where T_A, T_B and T_D are the thresholds for the area, eccentricity and diameter descriptors respectively. They are calculated as the difference between the maximum value of the descriptor in the database and the value of the descriptor for the query image:

$$T_i = (\max_{j \in DB} \{V_i^j\} - U_i) \cdot \theta_i \quad i \in \{A, E, D\} \tag{6}$$

where the parameter θ_i allows us to choose the final threshold and is a value in the range $[0, 1]$. The value 0 implies

Table 1: Example of eccentricity error intervals

θ_e	T_e	Error intervals
0	0	[4.7, 4.7]
0.2	1	[3.7, 5.7]
0.5	2.5	[2.2, 7.2]
0.8	3	[1.7, 7.7]
1	5	[-1.7, 9.7]

that the filtered database is limited to those images with values identical to those of the query image of the corresponding descriptor to the query image. On the contrary, the value 1 provides the complete database without deleting any records, because all items fulfil the criterion.

Table 1 shows the error intervals obtained supposing the query image has a value of eccentricity of 4.7 and the maximum value of eccentricity in the database is 9.7.

4.2 Similarity between contour descriptors

In the second phase of the retrieval, the characteristic points' vectors of the query and database images are used to measure the similarity between images. Each vector contains the values of curvature (c), sign (s), absolute distance (a), relative distance (r) and life (l) of each characteristic point of the shape as it was described in Section 3.2.

Let U and V be the query and database image, respectively, and u_i and v_i their i_{th} characteristic point. The similarity between the two images is calculated as the normalized sum of distances between points as it is described in Equation 7.

$$D(U,V) = \frac{PP}{N} \cdot \frac{\sum_{i=1}^{N} d(u_i, v_i)}{\sum_{i=1}^{N} \min(l_{u_i}, l_{v_i})} \quad (7)$$

Where PP is the penalty for deleting points and takes a value in the range $[0, 1]$ and N is the number of points compared. If the number of points of interest extracted from both images (the query image and the database image) is different, some of them have to be reduced. This point will be addressed in Section 4.2.1.

The distance between points $d(u_i, v_i)$ is calculated taking into account all attributes extracted for each point: value of curvature, distance between points, sign and lifetime. It will be explained in detail in Section 4.2.2.

4.2.1 Standardization Characteristic Points

Let $|u|$ and $|v|$ be the number of points of interest extracted from the query image and the database image, respectively. If $|u|$ is less than $|v|$, the number of points of interest of the database image has to be reduced. If $|v|$ is less than $|u|$, the points will have to be eliminated from the query image. The points are removed only for the purpose of calculating the distance and they are not eliminated from the database.

In both cases the algorithm for removing characteristic points depends on the values of lifetime and curvature of each point and it has the following steps:

1. Find the points with lower lifetime value.
2. If there is only one point with minimum life.
   (a) Delete the point with the lowest lifetime value.
   (b) Recalculate the relative distance between neighboring points of the deleted point.
3. If there are more than one point with lowest lifetime value, then find the point with the lowest value of curvature between the points of lower lifetime value.
   (a) If there is one point with less life and less value of curvature, it is removed.
   (b) Recalculate the relative distance between neighboring points of the deleted point.
4. In case of multiple points of lower lifetime value and lower value of curvature, it is removed the first of them.

The similarity function should give more importance to images that initially had the same number of points. So, the penalty factor PP for deletion operations is introduced into the measure. It is defined as follows,

$$PP = \begin{cases} \frac{(n-x)}{n} & if\, n-x \leq \theta_P \\ 0 & if\, n-x > \theta_P \end{cases} \quad (8)$$

where n is the number of points in the image with the highest number of points and x is the number of points in the image with less number of points. In this way, the greater the number of points removed, the greater the penalty. The value θ_P defines the maximum number of points that can be deleted to evaluate the similarity between the images.

4.2.2 Similarity Between Characteristic Points

Once both images have the same number of points, the distance between two is calculated as follows

$$d(u_i, v_i) = \frac{d_c(u_i, v_i) + d_r(u_i, v_i)}{2} \cdot PS \cdot min(l_{u_i}, l_{v_i}) \quad (9)$$

The function d_c represents the distance between the values of curvature of two points and it is calculated as a variant of the lambda membership function. Its definition is shown in Equation 10.

$$d_c(c_{u_i}, c_{v_i}) = \begin{cases} 1 - \frac{|c_{u_i} - c_{v_i}|}{m_{curv}} & if\ (c_{u_i} - m_{curv}) < c_{v_i} \\ & and\, c_{v_i} \geq c_{u_i}\ and \\ & c_{v_i} < (c_{u_i} + m_{curv}) \\ 0 & in\ other\ case \end{cases} \quad (10)$$

where c_{u_i} is the value of curvature of the i_{th} point of the query image U and similarly for c_{v_i} but of the database image V. m_{curv} is a parameter which define the grade of similarity between curvature's value of the image U and V. m_{curv} is a very important parameter, if set to a value larger m_{curv} different images will get a high degree of similarity, whereas if you define a small value of m_{curv} similar images will get a low degree of similarity. For this reason, a consensus value for m_{curv} must be defined.

The value of m_{curv} *enlarged* or *flattens* the lambda membership function where the vertex of the membership function is c_{u_i}. To clarify the concept, consider the following example: Suppose $c_{u_i} = 50$, $m_{curv} = 30$, the range where

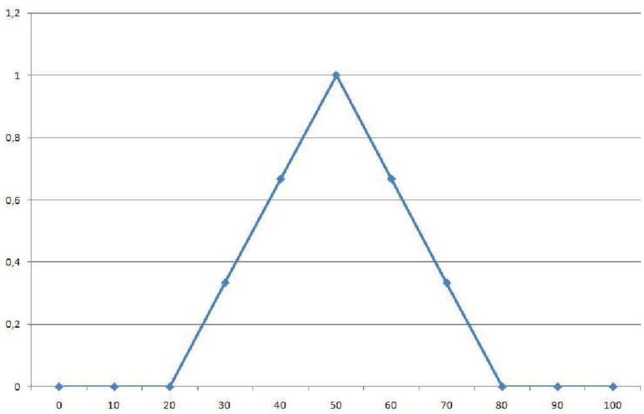


Figure 4: Lambda membership function with $m_{curv} = 30$ and $c_{u_i} = 50$.

the distance function is: $[50 - 30, 50 + 30] = [20, 80]$ as can be seen in Figure 4.

Any point outside the range will get the value 0, while the point with the same value as c_{u_i} will get the value 1. As the curvature values are being obtained by differentiating an lower outcome. As the curvature values are different the results are lower.

The calculation of the distance between the values of relative distance d_r is analogous to the calculation shown in Equation 10 for the values of curvature. In this case, m_{dist} is defined as a value equivalent to m_{curv} but to be used in d_r.

Two points are compared if they have the same sign value, i.e., both belong to a concave or convex portion of the curve. If it does not, a penalty factor called penalty of sign PS is taken into account,

$$PS = \begin{cases} 0 & if\, S_{u_i} = S_{v_i} \\ penaltySign & in\, other\, case \end{cases} \quad (11)$$

where $penaltySign$ is a value in the range $[0, 1]$.

To end the description of Equation 7, it is noted that the comparison of points is weighted with the lifetime attribute. The minimum of the lifetime value of both points is selected and it defines the weight of importance of that point.

Note that the measure is not invariant to rotations, could be invariant to rotations by changing the initial point in the matching, but with increased computational cost.

5. EXPERIMENTAL RESULTS

To verify the botanical CBIR we have taken near 300 leaf samples corresponding to 13 classes of plants collected by ourselves (as shown in Figure 5). Each class includes between 5 and 30 leaf samples. It have been selected images very diferent for each plant species, exist diferent images within the same plant species. In our implementation we used MatLab 7.0 on Intel PC 2.4 GHz 2 GB RAM.

5.1 Geometrical Descriptors' Results

As presented in Section 3.1 three geometrical descriptors have been selected for this study: area, eccentricity and diameter. Each of the geometrical descriptors has a threshold that is in the range $[0, 1]$. In this paper the geometric de-

Figure 5: Thirteen classes of plant leaves used for recognition.

Table 2: Top 5 results for the F_{DB} value and recall value exceeding 80%

θ_a	θ_e	θ_d	$\%F_{DB}$	%Recall	%Precision
0,2	0,6	0,6	56,96	81,58	19,17
0,2	0,6	0,7	55,15	83,35	18,87
0,3	0,5	0,6	54,46	81,85	16,53
0,2	0,6	0,8	54,04	83,74	18,60
0,2	0,6	0,9	53,88	83,90	18,38

scriptors thresholds takes the values in the set $A = \{0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$, in total there is $9^3 = 729$ settings for the thresholds. Each image of the database takes the role of input image, so 300 iterations are performed for each configuration The measures used to compare the settings are:

- **Recall** is the fraction of the images that are relevant to the query that are successfully retrieved.

$$Recall = \frac{\#(relevant\, images\, retrieved)}{\#(relevant\, images)} \cdot 100 \quad (12)$$

- **Precision** is the fraction of retrieved images that are relevant to the search.

$$Precision = \frac{\#(relevant\, images\, retrieved)}{\#(retrieved\, images)} \cdot 100 \quad (13)$$

- F_{DB} is the percentage reduction in the database.

$$F_{DB} = \frac{\#(retrieved\, images)}{\#(images\, in\, the\, database\,)} \cdot 100 \quad (14)$$

These measures are applied to each image in the database for each configuration to obtain a representative value.

The results are good. Note in Table 2 where is obtained a *recall* value of 81.58% and F_{DB} value of 56.96%, ie, the database is reduced to more than half and just deleted 2 relevant images of 10 images relevant. Futhermore, note that if the database is large there is not detrimental effect on the value of F_{DB} because this value is calculated from the selected images after applying geometric descriptors.

The recall value in practice is much better than shown in Table 2, because it counts as a loss of relevant information when the system doesn't selects a leaf of a plant species belonging to the same class as the class of the input image. Sometimes this is not loss of relevant information. For example, Figure 6 shows two sets of images of different kinds of plant species. The larger image is the input image and blocked images are the images that have been selected after using geometric descriptors. Figure 6 is easy to see how

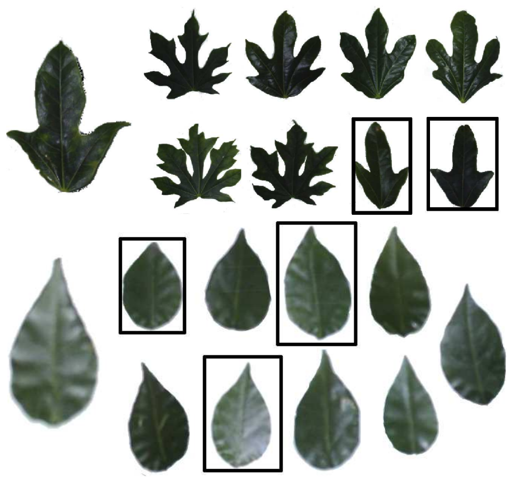


Figure 6: Selecting images. The blocked images are those that have exceeded the threshold values.

images of the same plant species are not alike and therefore these images are not selected.

5.2 Contour Descriptor's Results

To make the study of contour descriptors is used the following configuration of geometric descriptors: $area : 0.2$, $eccentricity : 0.6$ and $diameter : 0.6$.

These tests have used the following sets of parameters: $Scale$ takes values of the set $S_S = \{5, 7, 11\}$, m_{curv} takes values of the set $S_M = \{10, 20, 30, 40, 50, 60, 70, 80\}$, m_{dist} takes values of the set $S_D = \{10, 20, 30, 40, 50, 60, 70, 80\}$, $Penalty$ takes values of the set $S_F = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ and θ_P takes values of the set $S_P = \{3, 5, 7\}$. 2112 configurations are calculated for each scale. The calculations were made with 3 scales and therefore there are $3 \cdot 2112 = 6336$ configurations. For each configuration are calculated the following measures:

- **P_1:** Percentage of times that appears with greater value of similarity an image of the same class that the class of the input image.
- **P_3:** Percentage of times that appears between the top three an image of the same class that the class of the input image.
- **P_5:** Percentage of times that appears between the top five an image of the same class that the class of the input image.
- **C_A:** Percentage of times the class (plant species) is the same as the class of the input image. This measure is defined as:

$$C_A(e, P) = \frac{1}{\#DB} \cdot \sum_{i=1}^{\#DB} f(e_i, P_i) \qquad (15)$$

Where e is the set of elements that have exceeded gemetric descriptor thresholds and have been evaluated with the descriptors of contour and P is the set of similarity values of the elements.

The function f is applied to each image in the database to determine if the recovered class is correct or not. Finally, the value obtained is divided by the total number of images of the database.

Where f is defined as:

$$f(e, P) = \begin{cases} max_C(\forall class\, c \sum_{j=1}^{\#e} P(c, j)) & 1 \\ In\, other\, case & 0 \end{cases} \qquad (16)$$

Where the function $\max_C$ is responsible for selecting the class that contains the highest value after performing the sum of the similarity values of each of the classes and weighed in the number of items that have the class in the database.

5.2.1 Individual Results

This section shows the individual results using descriptors of contour. Configurations will be analyzed for values of P_1, P_3 and P_5.

For the parameter P_5 the best result is obtained using the scale 5, where the best result is 50.8%. This value indicates that more than half the time, individually, the CBIR system retrieves an image belonging to the correct class. In other words, every two requests appear an image of the correct class among the top five retrieved images by the CBIR system

For the parameter P_3 the best result is obtained using the scale 5 too, where the best result is 46,9%. As you can see this value is lower than that obtained using the parameter P_5, obviously this is what should happen because the parameter P_3 is a subset of the parameter P_5. In this case, a small percentage below half the time, individually, the system retrieves an image belonging to the correct class among the top three.

For the parameter P_1, following the trend of previous results obtained the best result for the scale again 5. Where the best result is 28.1%.

Figure 7 (a) shows that the system has recovered 3 images, of which the first two belong to the correct class. An individual level study shows that the first right image gets a value of 12.1% similarity. The following image gives a value of 8.80. Finally, the last image is not of the correct class but it seems very slightly as seen in the value of 1.35% obtained.

Figure 7 (b) shows that the system has recovered 6 images, of which the first, fifth and sixth belong to the correct class (*Bougainvillea glabra*). The best results were obtained in the first image with a similarity value of 58.27% which belong to the correct class. The third image corresponds to the plant species *Prunus cerasifera* obtaining a similarity value 34.05%. Finally, the second and fourth image belong to another plant species obtaining similarity values of 37.49% and 29.27% respectively.

The second, third and fourth image get a similarity value better than the fifth and sixth image because the number of characteristic points between the input image and the retrieved image is not very different and a set of these characteristic points are located in a similar way. However this result does not affect the discretion of the CBIR system, as individual and class CBIR system responds efficiently.

5.2.2 Results Using Classes

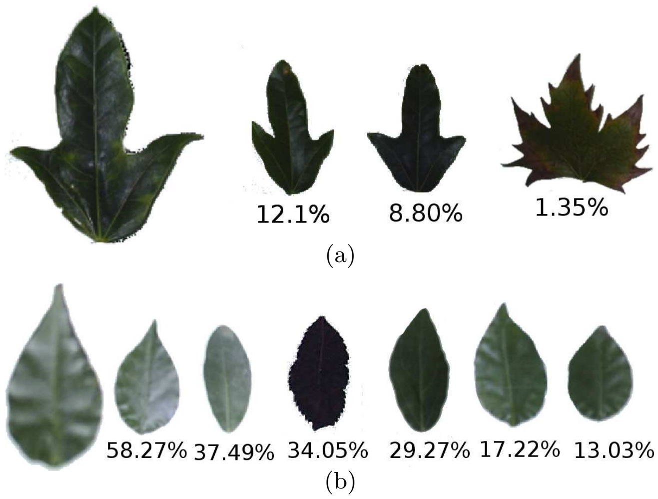


Figure 7: Retrieval results. The largest image is the input image, small images are the retrieved images with their similarity value. (a) Brachychiton acerifolius (b) Bougainvillea glabra.

Table 3: Top 5 results for parameter C_A using the scale 5.

m_{curv}	m_{dist}	PS	θ_P	C_A
10	10	0	3	92,96
10	10	0	5	92,96
10	10	0	7	92,96
10	20	0	3	91,40
10	10	0,1	3	91,40

In this section will be analyzed the retrieval results for classes. C_A is the most important metric in this study. Within seconds, the CBIR system determines precisely the class which is the input image.

In Table 3 shows the top 5 results obtained from the parameter C_A using the scale 5. This result is quite positive results are achieved close to 100%, which today is an utopia. The best result is 92.969%. This means that the CBIR system retrieves the correct class more than 9 times out of 10 input images.

As scale increases CBIR system efficiency decreases. This is so because a smaller scale the curvature will be subject to a lower number softened. Performing a lower number of smoothed imply there will be more information in the CBIR system to be a greater number of possible points for an image characteristic.

The average number of characteristic points for the scale 11 are 8-15 points per image against the 15-20 characteristic points per image on the scale 8. Finally using the scale 5 is getting better results because it has a average of characteristic points greater, around 40-45 points per image.

Then, perform analysis of results on class level of plant species to Figures 7 (a) and 7 (b). In Figure 7 (a) is trivial class which is associated to the input image. The plant species to which the input image belongs to the Figure 7 (a) is the plant species *Brachychiton acerifolius* because the sum of the values of the similarity of the images of the plant species recovered and weighted by the number of images of the class is greater than the value obtained for the plant species *Platanus hispanica*.

In Figure 7 (b), the CBIR system retrieves the class to which belongs the input image correctly. This is because the similarity values of the recovered images of the plant species *Bougainvillea glabra* accumulated a total of *88.52* units, weighted to 10 images of this plant species in the database gets a value of *8.852* units of similarity. Furthermore, the similarity values of the recovered images of the plant species *Prunus cerasifera* accumulated a total of *34.05* units, weighted to 6 images of this plant species on the database gets a value of *5.675* units of similarity for the plant species. Finally, the plant species of the second and fourth image accumulates a value of *66.76* units, weighted to 9 images of this plant species on the database gets a value of *7.417* units of similarity. Therefore, the input image in Figure 7 (b) will be associated with the plant species *Bougainvillea glabra*.

6. CONCLUSIONS

It has been developed a CBIR system applied to botany. Results have been very positive, because the correct class of plant specie has been recovered in more than 90% of cases. This has been achieved by using two types of descriptors: geometrical and contour. The use of geometrical descriptors allow us to perform operations with greater computational cost at a later stage on a small database. They are able to reduce the database to more than half with little loss of relevant information. Once the database has been reduced, the next step is the use of contour descriptors. In this case an important attribute was the new factor added to the characteristic points: *lifetime*. The lifetime attribute allowes us to decide which are the points most relevant in the image.

7. REFERENCES

[1] Abbasi, S., Mokhtarian, F., and Kittler, J. Reliable classification of chrysanthemum leaves through curvature scale space. *ICSSRC97* (1997), 284–295.

[2] Du, J., Wang, X., and Zhang, G. Leaf shape based plant species recognition. *Applied Mathematics and Computation 185* (2007), 883–893.

[3] González, R., and Woods, E. *Digital Image Processing (3rd Edition)*. Prentice Hall, 2008.

[4] Mokhtarian, F., and Abbasi, S. Matching shapes with self-intersection: application to leaf classification. *IEEE Trans. on Image Processing 13* (2004), 653–661.

[5] Mokhtarian, F., Abbasi, S., and Kittler, J. Efficient and robust retrieval by shape content through curvature scale space. *Proceedings of the International Workshop Image DataBases and MultiMedia Search* (1996), 35–42.

[6] Mokhtarian, F., and Mackworth, A. Scaled-based description and recognition of planar curves and two dimensional shapes. *IEEE Trans. Pattern Anal. Machine Intell. 8*, 1 (1996), 34–43.

[7] Otsu, N. A threshold selection method from gray level histograms. *IEEE-T SMC(9), n. 1, pp. 62-79* (1979).

[8] Rui, Y., She, A., and Huang, T. Modified fourier descriptors for shape representation - a practical approach. *First International Workshop on Image*

Databases and Multi Media Search, Amsterdam (1996).

[9] WANG, Z., CHI, Z., AND FENG, D. Fuzzy integral for leaf image retrieval. *Proceedings of Fuzzy System 1* (2002), 372–377.

[10] WANG, Z., CHI, Z., AND FENG, D. Shape based leaf image retrieval. *Image Signal Process 150* (2003), 34–43.

[11] XI, X., KEOGH, E., WEI, L., AND MAFRA-NETO, A. Finding motifs in database of shapes. In *In Proc. of SIAM International Conference on Data Mining* (2007).

[12] YANKOV, D., KEOGH, E., WEI, L., AND XI, X. Fast best-match shape searching in rotation-invariant metric spaces. *IEEE Transactions on Multimedia 10* (2008), 230–239.