一個自動化葉片辨識系統

研究生：黃薰瑩　　　　　　　　　　指導教授：陳玲慧 博士

國立交通大學多媒體工程研究所

摘 要

在郊外踏青總會接觸到許多植物，而讓遊客利用影像辨識了解它們勢必是個趨勢。葉片是植物重要元素之一，因此常用來當作植物辨識的標的物。在本篇論文中，提出了一個針對葉子區域本身為對象的兩階段葉片自動辨識系統。首先，我們萃取出五個特徵植，依據這五個特徵進行辨識動作。在第一階段的辨識中，會將形狀與輸入影像差異大者刪除。接著，便會依據第一階段過濾後的數種名單進行最後的辨識動作。該系統會依照辨識出來的結果提供參考影像，並允許使用者決定最正確的答案。在前五名的準確率為95.14%。此外，我們亦會在前處理時，將拍攝所造成的反光雜訊以及葉片旋轉問題去除。

索引詞：葉片、區域特徵、葉子影像辨識

An Automatic Recognition System of Leaves

Student: Hsun-Ying Huang　　　　　　　Advisor: Dr. Ling-Hwei Chen

Institute of Computer and Information Science
National Chiao Tung University

Abstract

When wandering around the field, we can touch many plants. It is useful knowing them through image recognition technique. Since leaf is one of the important features for characterizing various plants, it is often taken for plant recognition. The thesis proposes a hierarchical automatic region-based method for leaf recognition. First, delete impossible species to which the input leaf belongs according to the leaf shape represented by five extracted features. Next, based on these candidates, the system finds out the most similar images in our database and allows each user to choose the correct one. The precision rate is 95.14% for top 5. In addition, the proposed method is rotation invariant and solves the noises caused by light reflection in preprocessing.

Index term: leaves, region-based, leaf image recognition

致謝

感謝陳伶慧老師對我們的疼愛，就連平時不善表達的我，老師也都還是照顧有加，對於許多關於研究想法以及態度上，讓我收穫良多。感謝三位口試委員能讓我通過口試，順利畢業。感謝 AIP 實驗室所有學長姐、學弟以及同學們，不僅會為我們解答疑惑，平時的相處也更增添實驗室的輕鬆氣氛。感謝因為 TWISC 而認識的朋友 Stanly，如果沒有他好交朋友的個性，就不會和這麼多原先是交大的朋友接觸，讓我對於交大的體驗能更深入，最重要的是，也因為他的熱心促合，還讓我交到了俊賢這麼好的男朋友。感謝喬凱、佳昇、QMe、桶哥、豬頭聰、鴻哲和俊賢，在修課期間的相互討論以及一起抱佛腳的時光，讓我對於課業上的思考能夠有不同的想法。感謝高中好友刷琳，除了分享彼此平時的喜怒哀樂，更也由於她的樂觀與開朗，讓我除了在 AIP 實驗室之外，還可以遇到其他形形色色同居但多個不同實驗室的朋友，開闊了眼界，為平時規律的研究生活注入了不同的活力。重要的就是俊賢，不但在準備論文時期給予我許多的想法和鼓勵，願意包容我所有的缺點與任性，更因為他的堅強與溫柔，讓我學習到勇氣與成熟。最重要的是我的家人，感謝他們疼惜我的辛苦和健康，也很體諒我的忙碌。最後，僅以這篇論文獻給我的家人。

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1
Introduction

When wandering around the field, we can touch many plants. We may consult their name with the book about plants so that we can know them from tremendous plant species. However, it is inconvenient with such a book in hand all the time. Accompanying the computer aided, some systems, such as CalFlora [1] and TaiBNET [2], have been developed for plant recognition and plant data management. These systems help users recognize plant species by using textual inputs (such as plant name, scientific name, etc). Nevertheless, neither of them supports image search techniques. Leaf is one of the important features for characterizing various plants; it would be useful to develop an automatic system to recognize the leaf and retrieve the related information about the plant.

Content-based image retrieval (CBIR), also known as query by image content (QBIC), is an application of computer vision to the image retrieval problem, which is to search the correct images from large databases. The term, "content", may refer to colors, shapes, texture or other information that can be derived from the image itself.

Vein, one content for CBIR, is one of the characteristics of leaf, but how to extract the complete veins of the leaf automatically is a hard task. Park et al. [3] proposed a venation-based analysis for leaf categorization in 2006; nevertheless, the images in this research are hand-drawn. Additionally, a method presented by Fu and Chi [4], achieving an accuracy of 97.33% of venation extracting, was carried out by thresholding and neural network. Furthermore, these images are taken by the capture apparatus that includes a fluorescent light bank and a high-resolution digital camera. However, owning such equipment is not easy. Thus, it is difficult for amateur photographers or tourists taking pictures with clear venation of a leaf without the environment.

Shape, another content for CBIR, is commonly used in object recognition. In general, shape recognition is classified into two categories: boundary-based and region-based [5]. The former describes region of interest (ROI) by its external characteristics; the latter describes ROI by its internal characteristics. In boundary-based category, the majority of researches have focused on centroid-contour distance (CCD) and curvature scale space (CSS). Wang et al. [6] presented a leaf shape representation with a CCD curve and recognized the leaf with eccentricity and angle code histogram (ACH). The average recall rate is 38.1% for 20 return images. The CSS image was introduced by Mokhtarian and Mackworth [7] as another shape representation for planar curves. The precision rate in the retrieval method [8] using CSS is 95.25% for 5 returned images. Unfortunately, the common failing of boundary-based shape recognition is that the starting points of a leaf are hard to find.

It must be noticed that the representation invariant to translation, scaling and rotation for leaf is an important and essential criterion. Unlike Lee and Chen's system which is not rotation invariant [9], our region-based system satisfies the criterion.

In this thesis, we propose a two-stage automatic region-based method for recognizing the leaf. First, we use some techniques to remove the noises and solve the rotation problem in preprocessing phase. Then, we find possible species to which the input leaf belongs according to the leaf shape represented by five extracted features. Finally, based on these candidates, the system finds out the most similar images in our database and allows each user to choose the correct one.

The following is the organization of the remaining sections of this thesis. In Chapter 2, we describe the leaf images used in the study. Chapter 3 presents the proposed recognition method. The experimental results and discussions are given in Chapter 4, and concluding remarks in Chapter 5.

Chapter 2
Leaf Image Database

Leaves are usually clustered, so it is difficult to automatically extract a leaf from an impure background. As a result, a leaf plucked from a plant was put on a piece of white paper and then taken pictures with a digital camera. Thus, leaf images without the complex background can be obtained.

The database in this work collects 20 species of different plants. For robustness, we plucked 10~15 leaves from each species plant. Each species plant includes 60 leaf images, of which 20 images were randomly selected as database images and the remaining is taken as query data. Fig. 1 shows all species plant leaves for recognition.

Fig. 1. Twenty species of plant leaves used for recognition.

Chapter 3
Proposed Method

Fig. 2 shows the flow chart of our proposed method. The whole process contains three major phases: preprocessing, feature extraction and recognition. In preprocessing phase, we use some techniques to remove the noises and distortions of the leaf image; in feature extraction phase, we introduce how to extract features for recognition. A two-stage recognition based on the extracted features is applied in the last phase. These phases will be presented in Sections 3.1, 3.2, and 3.3 respectively.

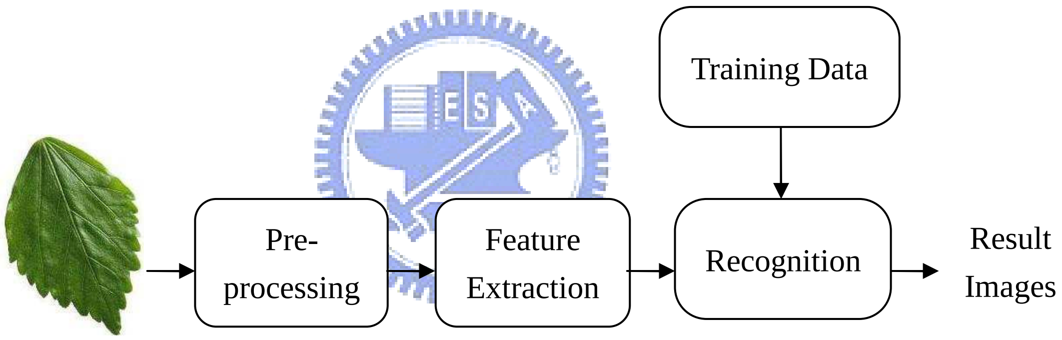


Fig. 2. The flow chart of the proposed method.

3.1 Preprocessing

When we take the leaf images, there are some distortions shown in Fig. 3(including rotation, translation and scaling) and few noises shown in Fig. 4 (caused by light reflection). Therefore, we have to eliminate these distortions and remove these noises to produce a uniform standard for future manipulation. Fig. 5 shows the flow chart of the preprocessing steps.

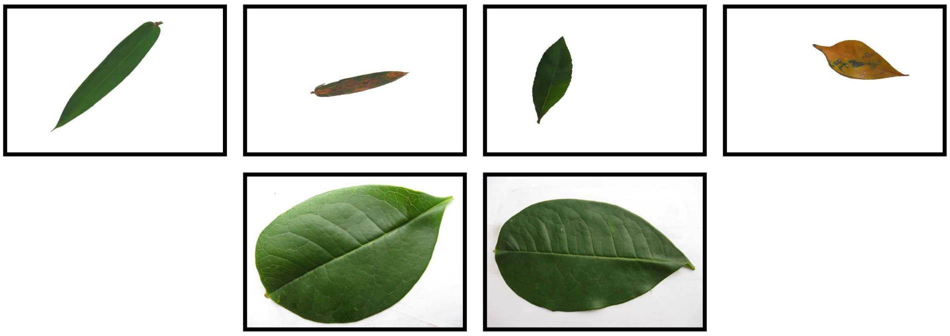
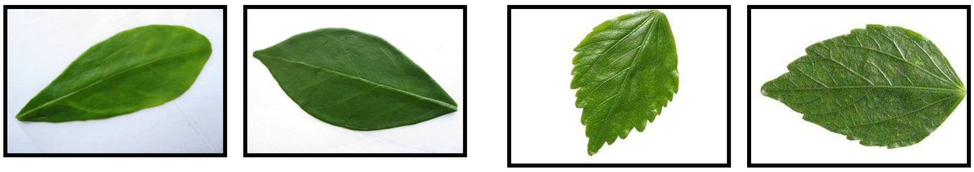


Fig. 3. Different rotation, translation and scaling of leaves.



(a) (b)

Fig. 4. The example of light reflection and blocky background.

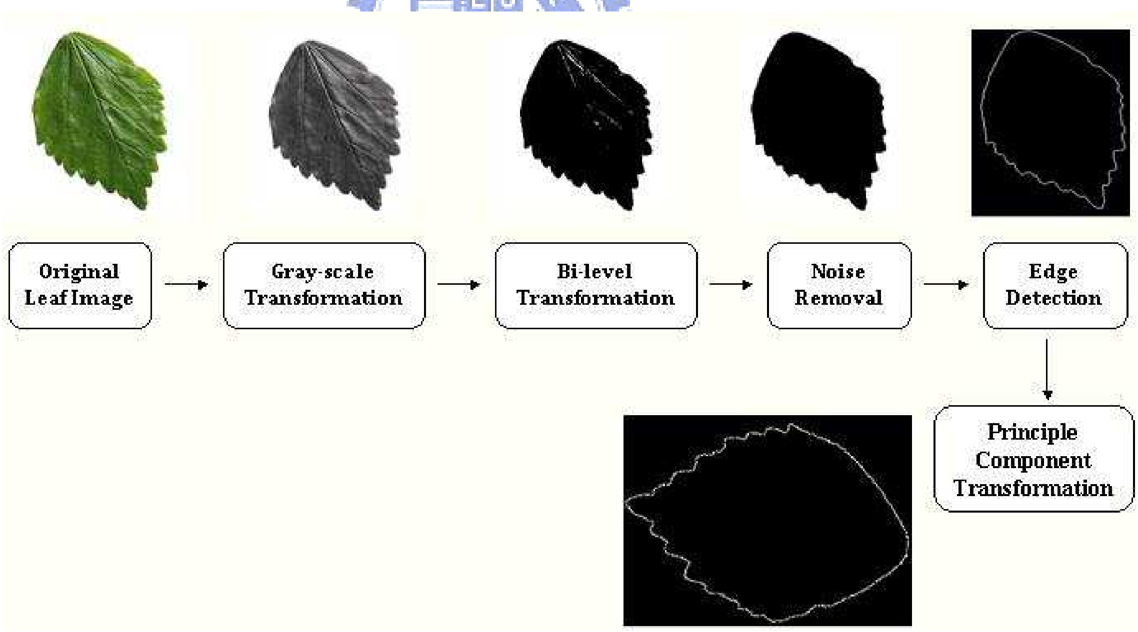


Fig. 5. The flow chart of the preprocessing steps.

3.1.1 Gray-scale Transformation

The colors of leaf are usually green, but the changes of shade, temperature and season cause the color various. For the reason, we transfer color images to gray-level ones. Each pixel is computed from the original image by

$$gray(i,\ j) = \frac{R(i,\ j) + G(i,\ j) + B(i,\ j)}{3}, \qquad (1)$$

where $R(i, j)$, $G(i, j)$ *and* $B(i, j)$ denotes the red value, green value and blue value at pixel (i, j). An example of gray-level transform is shown in Fig. 6.

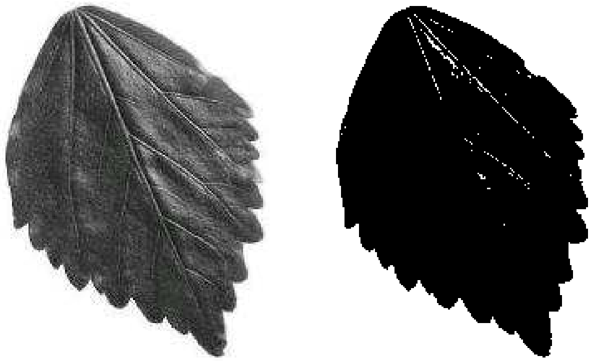
(a) (b)

Fig. 6. An example of gray-level transform. (a) Original image. (b) Gray-level image.

3.1.2 Bi-level Transformation

To simplify the cost of computation, we transform the 256-graylevel image to bi-level image. Otsu's method [10] is a classical method, which performs thresholding and the reduction of a gray-level image to a binary image. The algorithm assumes that an image contains two classes of pixels. It finds the optimum threshold separating the two classes so that their within-class variance is minimal.

Pixels with gray-level larger than the threshold t determined by Otsu's method are considered as background, and their gray values are set to 255; others are regarded as ROI and their gray values are set to 0. An example of bi-level transformation is shown in Fig. 7.



(a) (b)

Fig. 7. An example of bi-level transformation. (a) Gray level image. (b) Bi-level image of (a).

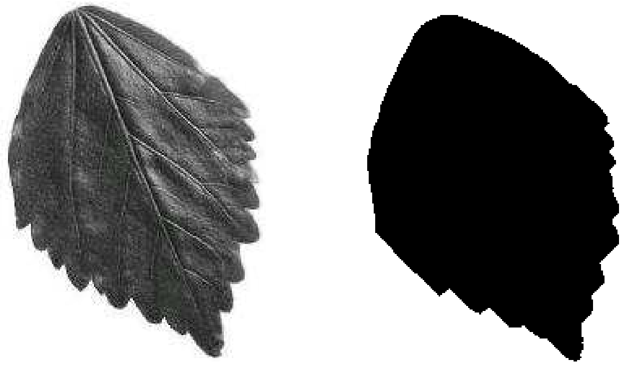
3.1.3 Noise Removal

That the surfaces of some leaves are glossy makes light reflection; thus we have to reduce these picture-taken errors.

Dilation and erosion are two basic operators in the area of mathematical morphology. The basic effect of dilation on a binary image is to gradually enlarge the boundaries of regions of foreground pixels. Thus, areas of foreground pixels grow in size while holes within those regions become smaller.

An example of a binary dilation for noise removal is shown in Fig. 8. The structure element is a 3×3 square, as shown in Fig. 8(a). Fig. 8(b) is a bi-level leaf image and the result of performing dilation in Fig. 8(b) is shown in Fig. 8(c).

0	1	0
1	1	1
0	1	0



(a) (b) (c)

Fig. 8. An example of dilation for noise removal. (a) The structure element of dilation. (b) A bi-level image. (c) The result of applying dilation on (a).

Even though dilation is a popular method, it still has a flaw in noise on the leaf images. Note that the wavy-toothed edge in Fig. 8(c) has been smoothed by dilation; erosion can not resume the tooth characteristic. To keep the shape of the original leaf edge, we propose a 2-step method to solve this problem.

The following is the steps for the proposed noise removal method:

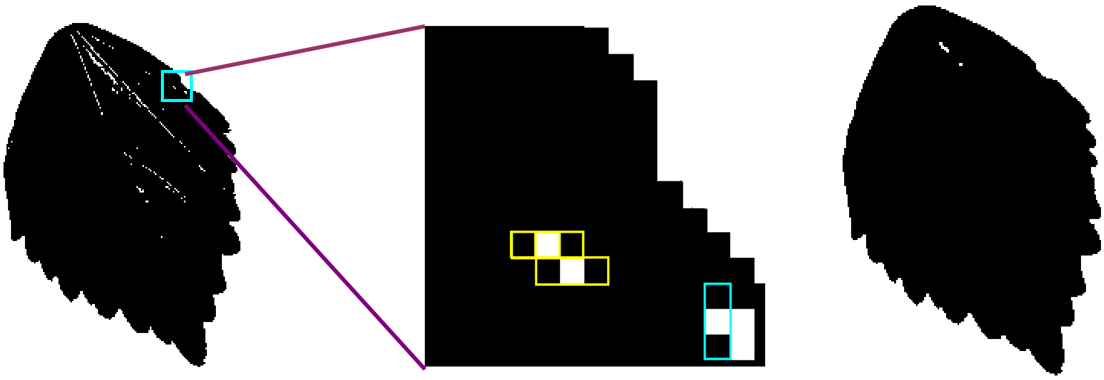
1. Fill the gaps with one-pixel

   1.1 For each white pixel, use 1×3 and 3×1 sliding windows to count the numbers of the black pixels respectively.

   1.2 We observe that the counts of most pixels in ROI are greater than or equal to 2. Based on the fact, if the count of one white pixel satisfies the condition, the pixel will be considered as a part of ROI. Then the pixel will be set to be black.

   Repeat steps 1.1 and 1.2 until there is no new assignment.

An illustration of step1 is in Fig. 9.



(a) (b) (c)

Fig. 9. The proposed method for noise removal. (a) The original bi-level image. (b) The illustration of using a 1×3 and 3×1 sliding window in noise removal step1. (c) The result image.

2. Fill the gaps with more pixels

   2.1 Let

   $n_1 = 3$,

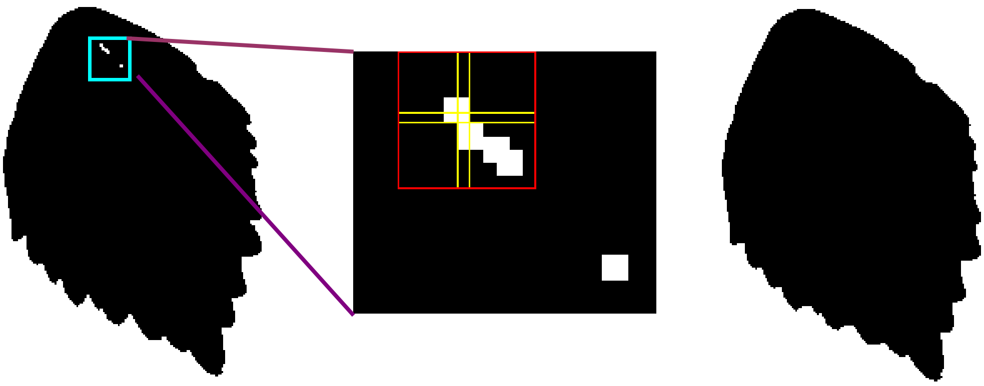
   $n_k = 2 * n_{k-1} + 1$, for $k = 2\ldots, 4$,

   $k = 2$.

   2.2 For each white pixel (i, j), use an $n_k \times n_k$ sliding window with the pixel as its center to check whether the colors of all the boundary pixels of sliding window are black. If the answer is true, then all white pixels in the sliding window are colored; otherwise, do nothing.

   2.3 Add one to k, and repeat step 2.2 to 2.4 until k equals 4.

An illustration of step 2 is shown in Fig. 10.



(a) (b) (c)

Fig. 10. The proposed method for noise removal. (a) The bi-level image after step 1. (b) The illustration of using an $n_k \times n_k$ sliding window in noise removal step 2. (c) The final result image.

3.1.4 Edge Detection

In order to reduce the amount of data and leave useful information, we apply an edge detector and thinning algorithm to get leaf boundary.

First, we use Sobel operators to get edge points. Next, we use Hilditch's Algorithm [11], which is the process of peeling off as many pixels as possible without affecting the general shape of the pattern, to thin the obtained edge points. An example of edge detection is shown in Fig. 11.

(a) (b) (c)

Fig. 11. An example of edge detection. (a) A bi-level image. (b) The edge points obtained by applying Sobel operators on (a). (c) The thinning result of (b).

3.1.5 Principle Component Transformation

That the directions of the leaf images are various is an issue for feature extraction. To solve this problem, we use the principle component to align the image with the direction of maximum variance.

Let

$\mathbf{X}_k = (x_k, y_k)$ be the position of the k-th boundary point of a leaf pixel,

n = the number of boundary pixels.

Steps of principle component transformation are presented as follows:

1. Compute the center point $\mathbf{m}_x$ of the boundary pixels by

$$\mathbf{m}_x = \frac{1}{n}\sum_{k=1}^{n} \mathbf{X}_k \tag{2}$$

2. Compute the covariance matrix

$$\mathbf{C}_X = \frac{1}{n}\sum_{k=1}^{n} \mathbf{X}_k \mathbf{X}_k^T - \mathbf{m}_k \mathbf{m}_k^T \tag{3}$$

3. Let $\mathbf{V} = (e_i, e_j)$ be the normalized associated eigenvector of the max eigenvalue of $\mathbf{C}_\mathrm{x}$, and the angle between it and x-axis is θ. Note that $\mathbf{V}$ is the principle component of the boundary points.

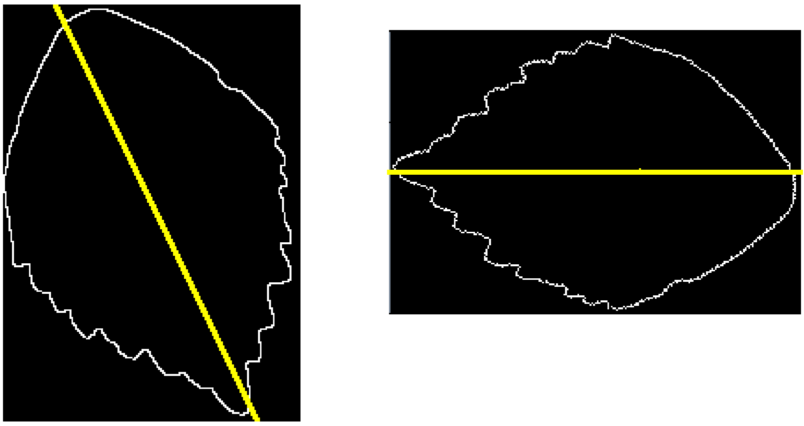
4. Let the transformation matrix $\mathbf{A}$ be $\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$.

5. Rotate the leaf θ degree according to

$$\mathbf{Y}_k = \mathbf{A}\left(\mathbf{X}_k - \mathbf{m}_x\right) \tag{4}$$

such that the principle component of the boundary points will coincide with the x-axis.

Fig. 12 shows an example of principle component transformation.



(a) (b)

Fig. 12. An example of principle component transformation. (a) The boundary of a leaf image and its principle component. (b) The transformed boundary of a leaf image.

3.2 Feature Extraction

According to the theory of plant taxonomy [12], external leaf characteristics are important for identifying plant species. These structures are a part of what makes leaves determinant; they grow and achieve a specific pattern and shape, then stop. However, there are morphological differences in different kinds of leaves; even in the same kind of leaves, there also exist some variance on scale.

Thus, we use ratios instead of those variable values. We find a rectangle enclosing leaf, and call it as a bounding box, *B*, shown in Fig. 13.

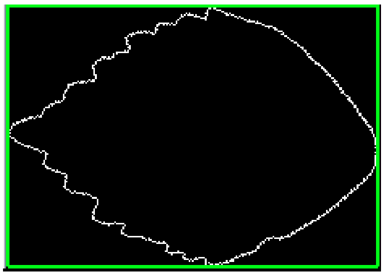


Fig. 13. The bounding box of a leaf.

3.2.1Aspect Ratio (AR)

The aspect ratio represents the ratio of the horizontal length and the vertical length of *B*. It is computed by

$$AR = \frac{width\ of\ \ B}{height\ of\ \ B}. \tag{5}$$

3.2.2 Rectangularity Ratio (RR)

The rectangularity ratio represents the ratio of the area of leaf and the area of *B*. It is computed by

$$RR = \frac{the\ area\ of\ \ leaf}{the\ area\ of\ \ B}. \tag{6}$$

3.2.3 Circularity Ratio (CR)

The circularity ratio represents the ratio of the average of the top one-tenth of the largest centroid-contour distance (CCD) and the average of the top one-tenth of the shortest CCD. It is computed by

$$CR = \frac{CCD_{short}}{CCD_{long}}. \tag{7}$$

The longest CCD and shortest CCD is shown in Fig. 14.

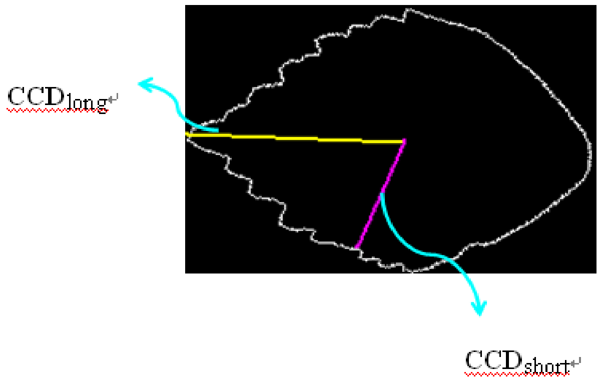


Fig. 14. An example to illustrate CCD_{short} and CCD_{long}.

3.2.4 UpAndDown Ratio (UDR)

The UDR represents the ratio of the area of the upper part and the area of the lower part of the leaf. It is computed by

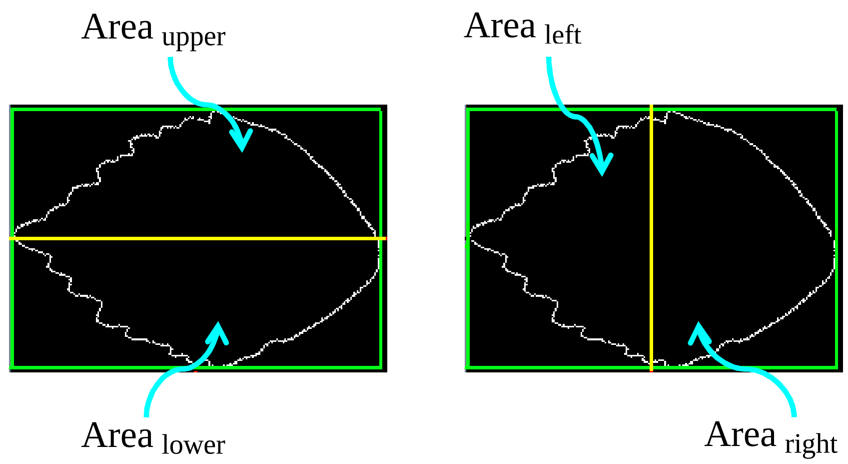
$$UDR = \frac{Area_{\ upper}}{Area_{\ lower}}. \tag{8}$$

3.2.5 LeftAndRight Ratio (LRR)

The LRR represents the ratio of the area of the left part and the area of the right part of the leaf. It is computed by

$$LRR = \frac{Area_{\ left}}{Area_{\ right}}. \tag{9}$$

Fig. 15 shows an example of UDR and LRR. Fig. 15(a) illustrates the upper part and the lower part of the leaf with the horizontal auxiliary line. Fig. 15(b) illustrates the left part and the right part of the leaf.



(a) (b)

Fig. 15. An example of UDR and LRR. (a) UDR. (b) LRR.

3.2.6 Training Data

3.2.6.1 Normalization

For robustness, the values of the features should be normalized. The following is the steps of normalization.

1. Let

   Avg_{min} be the average of the top one-tenth of the minimum features,

   Avg_{max} be the average of the top one-tenth of the maximum features.

2. The rules as following:

   2.1 If the original values of the features are grater then the corresponding Avg $_{max}$, then set the normalized values of the features to be 1.

   2.2 If the original values of the features are less then the corresponding Avg $_{min}$, then set the normalized values of the features to be 0.

   2.3 Or, compute the normalized values by

$$\text{the normalized value} = \frac{\text{the orignal value} - Avg_{\min}}{Avg_{\max} - Avg_{\min}}$$

3.2.6.2 Modified Basic Sequential Algorithm Scheme (MBSAS) Algorithm [13]

The MBSAS algorithm is an algorithm to cluster n objects based on attributes into k partitions, $k < n$. Unlike k-means, MBSAS algorithm does not know the numbers of the classes in advance. In order to reduce the size of the database and to speed up the search process, here, for each plant species, we use MBSAS algorithm to select representative images as the key images.

The feature vector (*AR*, *RR*, *CR*, *UDR*, *LRR*) is taken as attribute $f = \{f_1, f_2, f_3, f_4, f_5\}$, and it is used in the MBSAS algorithm. The followings are the steps of the algorithm:

1. For each plant species, randomly take m several images as seed images of the corresponding classes, $1 < m < 20$.
2. For each images of the plant specie, compute the distances between it and all seed images by

$$dist_k = \sum_{i=1}^{|f|} \sqrt{(f_i - s_{k_i})^2}, \quad k = 1,2,\ldots,m$$

where f_i denotes the i-th element of f , and s_{ki} denotes the i-th element of the attribute of the k-th seed image.

3. If $dist_j$ is the minimum, and

   3.1 $dist_j$ is less than or equal to a threshold; classify the image to the j-th class.

   3.2 $dist_j$ is larger than a threshold; set the image as the seed image of a new class.

4. Compute the centroid attribute of each class.

   Repeat step 2 to step 4 until there is no new assignment.

5. Set the closest image to the centroid of each class to be the key image.

After applying MBSAS algorithm, we will have random number classes for each plant species.

3.3 Recognition

Fig. 16 is the flow chart of recognition method. The proposed two-stage recognition method contains two phases. The preliminary phase is to omit some impossible species the query image belongs to using three features; the essential phase is to recognize the query image using five features in local view.

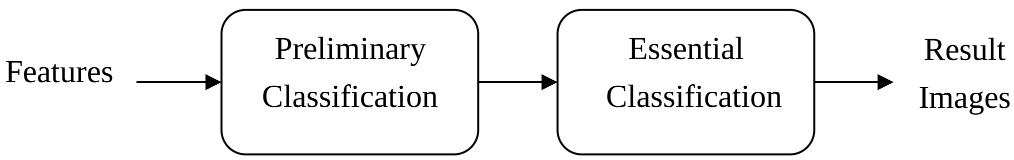


Fig. 16. The flow chart of recognition method.

3.3.1 Preliminary Classification

Since each plant species has specific shape patterns, we can use the shape characteristics to omit most impossible species. Here, we take (*AR*, *RR*, and *CR*) as features. The followings are the steps of preliminary classification:

1. Let $range_i(AR)$, $range_i(RR)$ and $range_i(CR)$ be the ranges of *AR*, *RR* and *CR* of the *i*-th plant specie in database, respectively.
2. Let $range_i = \{range_i(AR), range_i(RR), range_i(CR)\}$.
3. Consider each plant species in database as a candidate plant species of the query image if the feature (*AR*, *RR*, and *CR*) of the query image are in the range of the plant species.

Table 1 shows the ranges of AR, RR and CR of all plant species in our database and the illustration of the distributions of these ranges are shown in Fig. 17(a), Fig. 17(b) and Fig. 17(c) respectively.

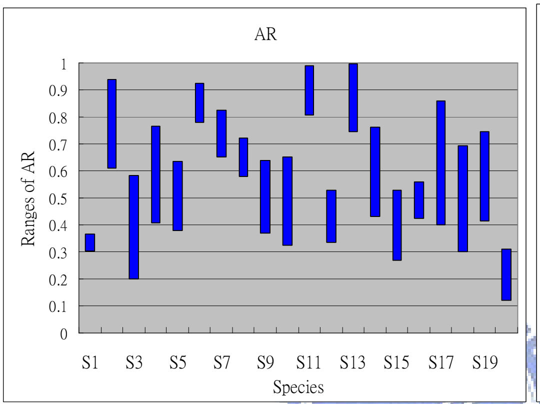
Table 1. The ranges of *AR*, *RR* and *CR* of every plant species in our database. (continued)

Plant Species		AR		RR		CR	
		min	max	min	max	min	max
S1		0.3021978	0.3658088	0.6917577	0.7429926	0.31192043	0.36635575
S2		0.61165047	0.936255	0.5543392	0.6898887	0.52551013	0.76205605
S3		0.20111732	0.5825243	0.5946818	0.7114526	0.20618778	0.56196773
S4		0.4068441	0.76724136	0.5412388	0.68510383	0.41154012	0.6782402

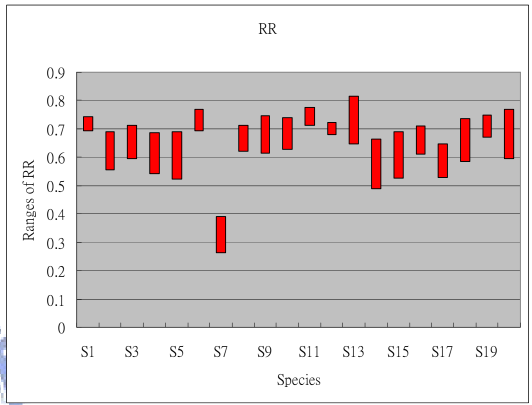
S5		0.3800813	0.63380283	0.52117145	0.68991154	0.38765764	0.5626684
S6		0.7803738	0.9232737	0.6941148	0.76962507	0.35281536	0.45798305
S7		0.65337425	0.8229665	0.26182264	0.39001942	0.25512147	0.34907347
S8		0.5780886	0.7204969	0.62200063	0.713519	0.55668813	0.6687718
S9		0.36818182	0.63799286	0.613013	0.74635494	0.35771707	0.6110392
S10		0.32509506	0.6511156	0.6280546	0.7379492	0.32315022	0.6298897
S11		0.808	0.990991	0.71349865	0.77655447	0.73534054	0.86972296
S12		0.3347732	0.5275	0.6815387	0.7228905	0.31594595	0.50810903
S13		0.74496645	0.99590164	0.6471868	0.81376123	0.23674217	0.6788229
S14		0.43260187	0.76344085	0.4896226	0.66393507	0.38459423	0.6690141
S15		0.26923078	0.5268817	0.52427256	0.68889624	0.26636413	0.4937982
S16		0.4235537	0.5601578	0.6100821	0.708394	0.3923918	0.5372656
S17		0.40167364	0.85897434	0.5294677	0.64786536	0.39352012	0.592582

Table 1. The ranges of *AR*, *RR* and *CR* of every plant species in our database. (continued)

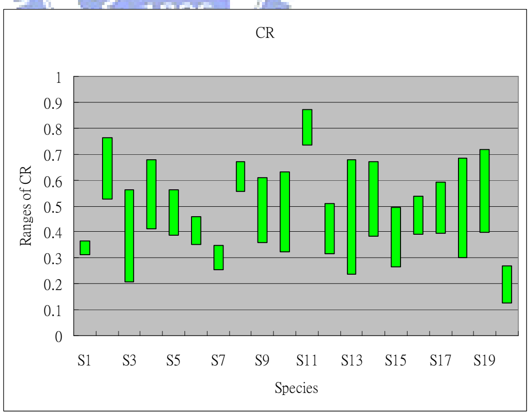
18		0.30110496	0.6939891	0.5862853	0.7344906	0.3008486	0.6832951
S19		0.41276595	0.74475527	0.6697521	0.7476528	0.39881355	0.71584564
S20		0.12008733	0.30921054	0.59406495	0.76971114	0.1268859	0.26997045



(a)



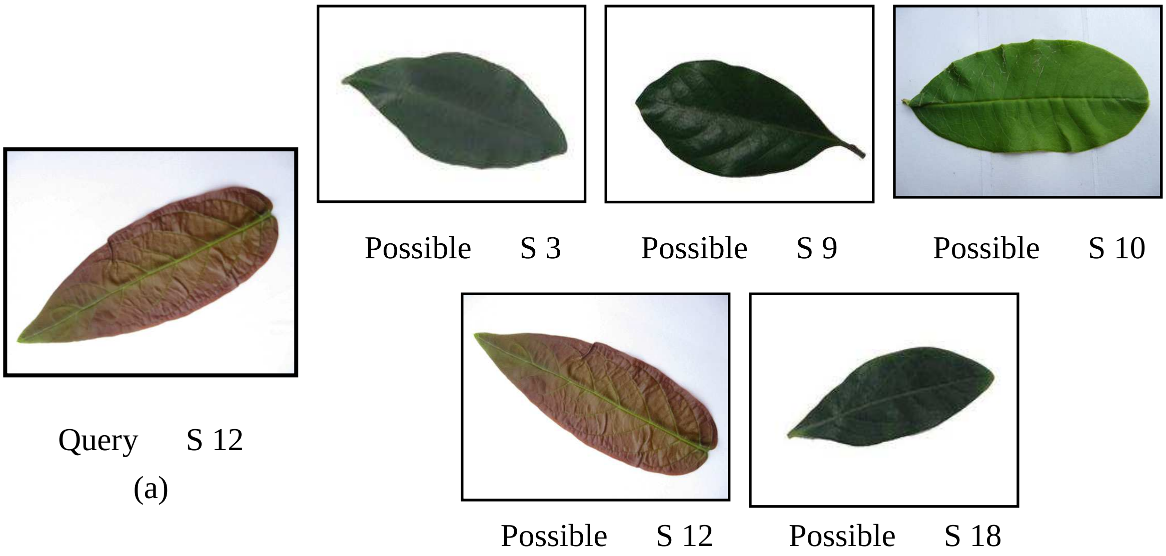
(b)



(c)

Fig. 17. The illustration of the distributions of these ranges. (a) The distribution of *AR*. (b) The distribution of *RR*. (c) The distribution of *CR*.

Fig. 18 shows an example of the result of the classification. Fig. 18(a) is a query image. There are 20 plant species in our database; from Fig. 18(b), we know that there are only five species left as the candidate species after the preliminary classification.



(b)

Fig. 18. An example of the result of the preliminary classification. (a) The query image. (b) The possible species of (a).

3.3.2 Essential Classification

After preliminary classification, some candidate species have been extracted. Based on these candidates, we take (*AR*, *RR*, *CR*, *UDR*, and *LRR*) as a feature vector to find the exact species of the query image. The following is the steps of essential classification:

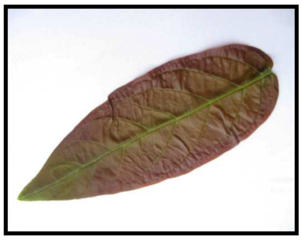
1. Calculate the distance between the feature vector of the query image and the feature vector of each key image in the possible species extracted from preliminary classification by

   $$dist = \sum_{i=1}^{5} \sqrt{(f_i - K_i)^2}$$

   where f_i denotes the i-th elements of the feature vector in the query image, and K_i denotes the i-th elements of the feature vector of a key image in a possible species.

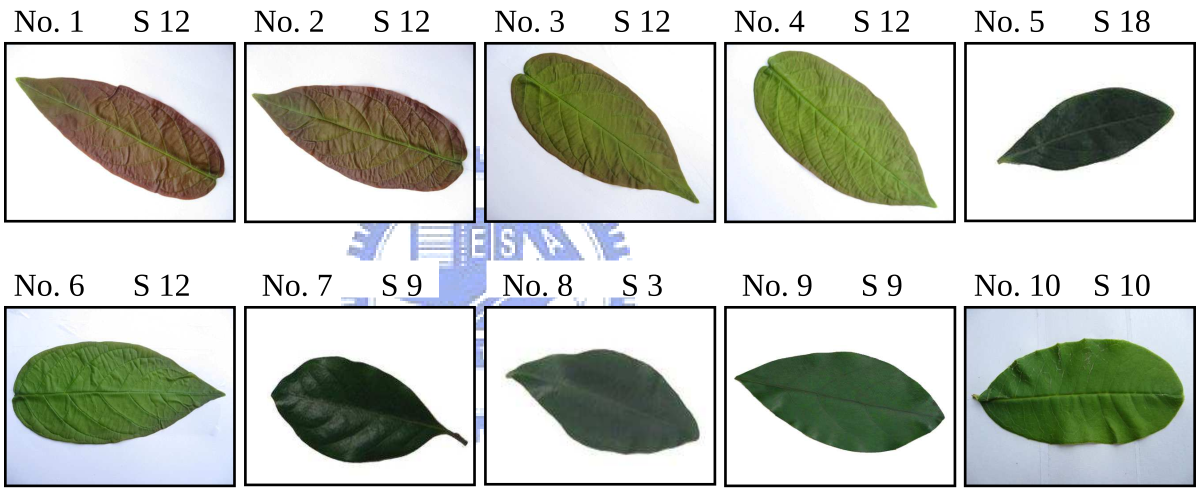
2. We can get the higher ranking of the corresponding plant species with the smaller *dist*.

Fig. 19 shows an example of result. Fig. 19(a) is one query image of the species S 12, and the final result for top 10 is show in Fig. 19(b). The number of total training images of the species S 12 in our database is 5, and these training images rank No. 1, 2, 3, 4 and 6 respectively.



Query S 12

(a)



(b)

Fig. 19. An example of the essential classification result. (a) The query image. (b) The final result for top 10.

Chapter 4
Experimental Results

A database containing 1200 color leaf images from 20 plants taken by us was used to conduct the experiment. Each species includes 60 images, of which 20 images are selected randomly as database images and the remaining is used for testing data. Moreover, some key images were selected from these database images according to MBSAS algorithm described above.

Two measures to estimate the performance are recall rate and precision rate. The recall rate is defined as $(N_{re} / N_t) \times 100\%$, where N_{re} is the number of the returned image which has the same species as that of the query image and N_t is the number of the key images which have the same species; the precision rate is defined as the numbers of the successful query images over the total numbers of the query images. Note that the successful query image means there is one image correct in top k returned images.

Two experiments, only essential phase_and essential phase_with preliminary phase, were conducted by our approach. The result is schematized in Fig. 20. The finger indicates that the overall precision rate for top 5 is 87.63158% and 92.97297% in only essential phase_and two-phase, respectively. On the other hand, the average execution time for each query image in two-stage method is about 5 seconds; it is less than that in only essential phase method. It proves that a two-stage hierarchical_method has higher precision and efficiency.

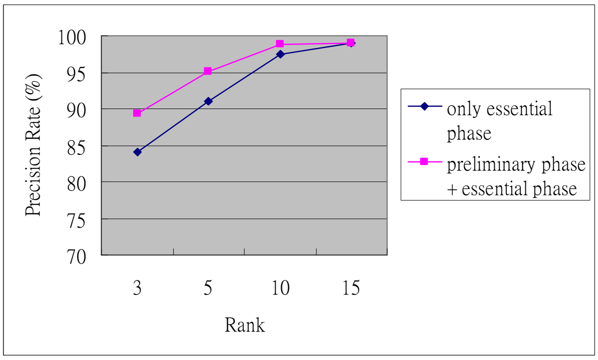


Fig. 20. The ranking-precision curve.

We compare the proposed method with the retrieval method using DFH descriptor provided in [14]. Swedish trees leaves database, collected by the Swedish Museum of Natural History for a project with Linköping University, was used in this research. It contains 15 different Swedish trees species. Each species is represented by 75 images, and all of them demonstrate isolated leaves. The result of the comparison depicts with the rank-precision curve shown in Fig. 21. Note that some methods (such as CSS descriptor, Hough transform [15], CCH descriptor and EOH [16]) have already been implemented to measure the contributions of DFH descriptor proposed in [14]. From Fig. 21, we can see that our method is superior to those in [14].

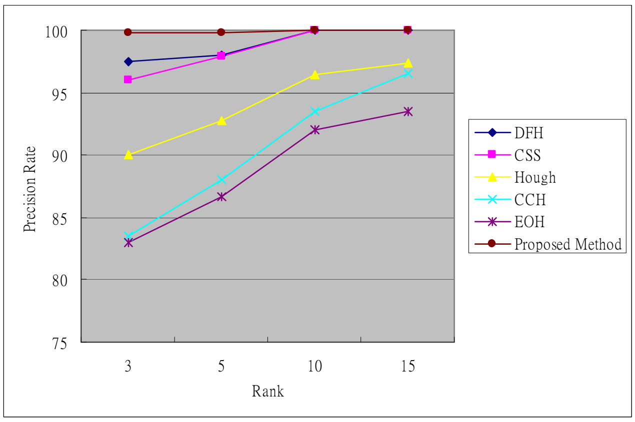


Fig. 21. The result of comparisons between our proposed method and [14].

Chapter 5
Conclusion

A region-based hierarchical classification method for leaf images has been proposed in this study. The goal of the research was to satisfy translation, scaling and rotation criterion and to recognize an unknown leaf. In the preprocessing stage, we solved the rotation problem and proposed a noise removal method, which can solve the problem of light reflection. Next, the five features according to the characteristics of the leaf shapes had been extracted. Then we preserve possible species according to the leaf shapes. Finally, the system found out the similar images to the query one and allows the user to make the final decision and the illustration is shown in Fig. 22.

Fig. 22. The illustration of final decision.

On average, the classification accuracy for top five is 92.97297% and the recall rate for 15 returned images is 72.36487%. In addition, the hierarchical strategy is adapted to raise precision and efficiency.

Future work can be listed as follows.

1. More leaf images should be collected to generalize the proposed classification

method.

2. More features of leaf should be included to improve the classification performance of the proposed method. For example, the texture of venation or contour-based features can provide useful information.

3. It would be systematic in botany to integrate flower image system, stem image system as well as the whole plant image system into the proposed one.

Reference

[1] Calflora, UC Berkeley Digital Library Project, URL: http://www.calflora.org/.

[2] TaiBNET, Taiwan Biodiversity National Information Network, URL: http://taibnet.sinica.edu.tw/.

[3] J. Park, E. Hwang, and Y. Nam, "A Venation-Based Leaf Image Classification Scheme," Lecture Notes in Computer Science, Springer-Verlag, Vol. 4182, pp. 416-428, Oct. 2006.

[4] H. Fu and Z. Chi, "Combined Thresholding and Neural Network Approach for Vein Pattern Extraction from Leaf Images," IEE Proceedings - Vision, Image & Signal Processing, Vol. 153, Issue 6, pp. 881-892, Dec. 2006, U.K.

[5] Y. Nam, E. Hwang, and K. Byeon, "ELIS: An Efficient Leaf Image Retrieval System," Lecture Notes in Computer Science, Springer-Verlag, Vol. 3687, pp. 589-597, Aug. 2005.

[6] Z. Wang, Z. Chi and D. Feng, "Shape Based Leaf Image Retrieval," IEE Proc.-Vis. Image Signal Proc., Vol.150, No. 1, pp. 34-43, Feb. 2003, U.K.

[7] F. Mokhtarian and A. Mackworth, "Scale-Based Description and Recognition of Planar Curves and Two-Dimensional Objects," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.8, No.1, pp. 34-43, Jan. 1986.

[8] F. Mokhtarian, S. Abbasi and J. Kittler, "Reliable Classification of Chrysanthemum Leaves Through Curvature Scale Space," Lecture Notes in Computer Science, Vol. 1252, pp. 284-295, 1997.

[9] C.L. Lee and S.Y. Chen, "Classification of Leaf Images," International Journal of Imaging Systems and Technology, Vol. 16, No. 1, pp. 15-23, Jul. 5, 2006.

[10] N. Otsu, "A Threshold Selection Method from Gray-level Histogram," IEEE Transactions on Systems, Man, Cybernetics 8, 1978.

[11] C. J. Hilditch, "Comparison of Thinning Algorithms on a Parallel Processor,"

Image Vision Computing, Vol. 1, No. 3, pp. 115-132, 1983.

[12] Plant taxonomy, Wikipedia Google, URL: http://en.wikipedia.org/wiki/Category:Plant_taxonomy.

[13] S. Theodoridis and K. Koutroumbas, Pattern Recognition, Academic Press 391–392, 1999.

[14] Itheri Yahiaoui, Nicolas Hervé and Nozha Boujemaa, "Shape-Based Image Retrieval in Botanical Collections." Pacific-Rim Conference on Multimedia (PCM), pp.357-364, Nov., 2006.

[15] M. Ferecatu, Image Retrieval with Active Relevance Feedback Using Both Visual and Keyword-based Descriptors, PhD thesis, Université de Versailles Saint-Quentin en Yvelines, 2005.

[16] A. K. Jain and A. Vailaya, "Image retrieval using color and shape", Pattern Recognition, Vol. 29, No. 8, pp. 1233-1244, Aug. 1996.