# 國立交通大學

## 多媒體工程研究所

## 碩 士 論 文

# 互 動 式 的 花 朵 辨 識 系 統

# An Interactive Method for Flower Recognition

研 究 生：徐子翔

指導教授：陳玲慧　教授

中 華 民 國 九 十 七 年 六 月

# 互 動 式 的 花 朵 辨 識 系 統
# An Interactive Method for Flower Recognition


研 究 生：徐子翔　　Student：Tzu-Hsiang Hsu

指導教授：陳玲慧　　Advisor：Ling-Hwei Chen

國 立 交 通 大 學
多 媒 體 工 程 研 究 所


碩 士 論 文

A Thesis
Submitted to Institute of Multimedia Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science

June 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年六月

# 互動式的花朵辨識系統

研究生：徐子翔　　　　　　　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

本篇論文中我們提出一個互動式花朵辨識系統，針對由數位相機直接從真實景色中拍攝的花朵影像進行辨識。花朵的形狀以及顏色是辨識上相當重要的特徵，在本篇論文中，總共使用了十四種特徵，針對花朵外型有三種，另外十一種為顏色特徵。我們提出一個與使用者互動的方法，加以限制花朵擷取的範圍，解決在花朵叢生所造成的重疊影像中，單一花朵輪廓擷取的困難。接著利用平均成本方法(NC method)進行邊界擷取。系統針對 348 張影像(共 24 種)做實驗，實驗結果顯示可以有效的進行輪廓擷取並且辨識花朵的種類。

# An Interactive Method for Flower Recognition

Student: Tzu-Hsiang Hsu Advisor: Dr. Ling-Hwei Chen

Institute of Multimedia and Engineering

National Chiao Tung University

## Abstract

In this thesis, we present an interactive method for recognizing flower on a flower image taken by a digital camera in the real scene. Shapes and colors of flower are very important features for recognizing, so fourteen features are used in our recognition system: three for shapes and eleven for colors. We provide an interactive method to limit the extraction area to solve the problem at extracting flower boundary in the overlapped flower image. Then we implement a normalized cost (NC) method for boundary extraction. Experiments are conducted on the database of 348 flower images to demonstrate the performance of the method, results show the effectiveness for recognition.

# 誌　　謝

這篇論文的完成，首先要感謝指導教授陳玲慧博士，在這兩年碩士生涯中，在課業上和生活上給予的指導和關心，讓我在交大學到的不僅是學業上的知識，還有更多做人做事的道理。此外，感謝口試委員李建興教授、張隆紋教授以及陳佑冠教授於口試中給予的指導與建議。尤其感謝李建興教授額外給予的指導，使整篇論文更趨完善。

接著，要謝謝實驗室一起生活一起奮鬥的夥伴們，博士班的學長們，民全、萓聖、文超、惠龍、俊旻以及占和，還有大我們一屆的學長姐們，芳如、佩瑩、立人以及維中，以及同屆的同學，偉全、信嘉和薰瑩，最後當然少不了貼心活潑的學弟們，志鴻、益成、明旭與志達，因為有大家的陪伴，讓我兩年的碩士生涯過得充實又豐富。

此外，還要感謝我親愛的姐妹淘，阿寶和詩敔，分享我所有快樂與不快樂的情緒，即使在新竹唸書，卻感覺妳們總是在我身邊，給我支持和打氣。

最後，感謝我最重要的家人，給我的信賴和依靠，因為有你們，我才有機會完成這些。謹以此篇論文獻給你們，以及所有關心我、鼓勵我的人。

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

## 1.1 Motivation

When people wander over the field, some blooming flowers around them can be found out easily. If there is no any placard for the plants, the names of them will not be known. In general, the name of the flower may be consoled with books or searched on the Internet. But there is a problem with consoling books and keying the keywords for searching. That is, finding a proper section or keywords to represent a specific flower is not an easy task. It would be useful if the flower can be recognized directly based on the photographs taken from the outdoor.

## 1.2 Previous work

In this thesis, an interactive method for recognizing flowers based on a photograph is proposed. Extracting a flower region of interest from the background is a necessary step for flower recognition. However, it is very difficult to achieve perfect segmentation from a complex background (e.g. nature scene). To avoid this difficulty, Saitoh and Kaneko [1] used a black sheet under the flower when they took photographs with it. They used k-means algorithm for clustering the background part and then separated it. For recognition, they proposed 17 features of flower and leaf images and received a high accuracy rate above 90%. But photographing flowers with a back sheet is inconvenient and laborious.

Das et al. [2] presented an interactive extraction system for recognition flowers based on color. They had concluded domain knowledge on the precise color of flowers. For example, flowers are rarely green, black, grey or brown. Hong et al. [3] used the same domain knowledge as Das et al. [2] and defined a look-up table of 25 colors for flower and background. They used this table and color clustering method for flower segmentation. But it will cause wrong segmentation when the flower is overlapped with others. For recognition, they used two shape features for flower image retrieval, Centroid-Contour distance (CCD) and Angle Code Histogram (ACH). CCD is a set of distances from those points in the contour to the shape center. ACH is a distribution of angles from each contour point measured by two approximate lines coming to and leaving the point. However, the pattern matching of CCD and ACH in recognition is time consuming.

Zou and Nagy [4] used both color and shape information for segmentation and proposed new shape features for flowers. For Segmentation, they first obtained an initial flower segmentation using general foreground and background color distributions derived from a set of segmented training images. Each training image is segmented manually into a foreground and a background region. Then, they used a rose curve model fitting on the initial segmented flower region and allowed interactive adjusting by user to fit the real flower region. When adjusting the curve for the flower, the system would receive features at the same time and refreshed the candidate images. They only used the petal number and a ratio in rose curve for flower shape features. However, it needs a lot of user's adjustment to get high accuracy.

Nilsback and Zisserman [5] used the same method as [4] to do initial segmentation, and then they used a generic flower shape model for petal structure to modify the flower region. They used three types of features: color, shape and texture. However, both of shape and texture features will encounter scale variation problem which deteriorates the system's performance. Otherwise, the separability of texture feature is very small.

Saitoh et al. [6] proposed an automatic method for extracting flower region and recognition. Different from those methods mentioned above, it is based on "Intelligent Scissors" (IS) [7] to find the flower boundary. The advantage is that it does not need any prior information (e.g. color distributions derived from a set of segmented training images). The disadvantage is that it works under the assumption that the flower is well focused in the center of the photographs and the background is out of focus. Pictures were taken with macro mode setting F value from F2.8 to F3.5. They proposed 10 features for flower recognition and received high accuracy above 90%. In this thesis, an interactive method is proposed for flower recognition which utilizes user's interaction to eliminate the limitations of photographs and to get an efficient flower boundary. And by adopting fourteen various features including 3 for shapes and 11 for colors, the proposed method is concise and effective.

The rest of the thesis is structured as follows. Chapter 2 describes the flower images used in the study. The proposed recognition method is presented in Chapter 3. Some experimental results are reported in Chapter 4. The final chapter gives conclusions.

# CHAPTER 2

# FLOWER IMAGE DATABASE

Flower images used in the thesis were collected from 24 species of wild flowers in the fields. The flower images are obtained by using a digital camera with macro capability. The range of the F value for these images is F2.8 to F5.6. Cameras we used include SONY T9, Canon IS 860 and NIKON S1. For robustness, images are taken from different flowers for each species. There are 4 to 33 images for each species. Several images contain multiple, tiny, overlapping flowers shown in Fig. 1. Flower images are resized to 400 x 300 pixels. Fig. 2 shows 24 species of flowers in our database.

Fig. 1. The variation of flower photographs.

Fig. 2. Images of 24 distinct flowers.

# CHAPTER 3

# THE PROPOSED METHOD

The flow diagram of the proposed method is shown in Fig. 3. The whole process consists of three major phases: segmentation, feature extraction, and recognition. In the segmentation phase, a user first draws an area containing the flower being recognized, and then the flower boundary is extracted. In the feature extraction phase, three shape features and eleven color features are extracted for recognition. In the recognition phase, based on the extracted features, a similarity measure is provided. Based on the similarity measure, the flower image in the database that is most similar to the input image is determined.

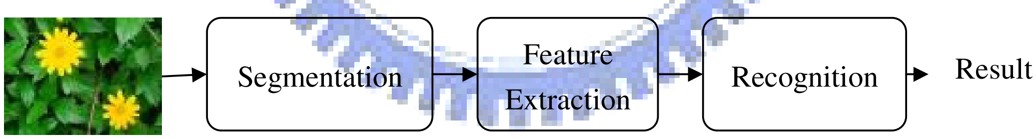
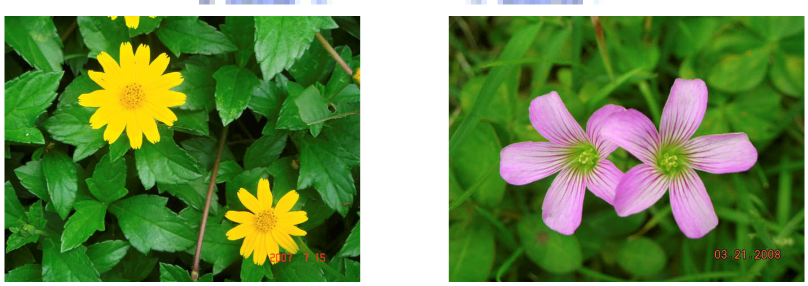



Fig. 3. Flow diagram of the proposed method.

## 3.1 Segmentation

In this section, an interactive segmentation method will be described for flower boundary extraction on the images which contain tiny, multiple or overlapping flowers. Extracting a flower of interest from the background is a necessary step for flower recognition. Saitoh et al. [6] provided the normalized cost (NC) method to do flower segmentation. This method will extract the boundary of a flower located in the center of the input image based on Intelligent Scissors (IS) [7] which searches an object boundary by minimizing the normalized cost, which is the sum of local costs divided by the route length. Local cost is calculated based on image gradient. Pixels that exhibit strong edge property have low local cost. Details will be described in Section 3.1.2.

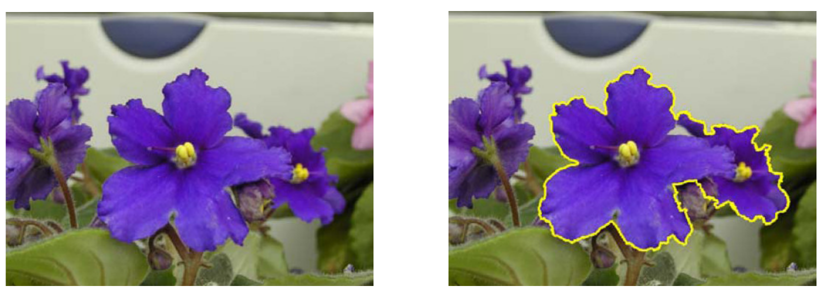
Fig. 4 shows two examples of input photographs. A flower may neither be placed on the center (Fig. 4(a)) nor well-focused from background and may overlap with other flowers (Fig. 4(b)). It is hard to extract the flower region from this kind of pictures automatically.



(a) (b)

Fig. 4. Input photographs.(a) With multiple flowers. (b) With overlapping flowers.

If the method proposed by Saitoh et al. [6] is applied to this kind of pictures, it may cause wrong extraction boundary shown in Fig. 5. The reason is that their method works on two assumptions. One is that the query flower is well-focused and the background is well-defocused, this means that the query flower is well separated from the overlapping flowers or background. The other is that the flower is placed in the center of the input image. Here, an interactive flower region extracting method is provided to deal with the photographs with tiny, multiple and overlapping flowers. Steps of segmentation are shown in Fig. 6 and described below.



(a) (b)

Fig. 5. An example of wrong extraction in overlapping flowers. (a)Original image. (b) The extracted flower boundary.

### 3.1.1 Flower and stamen bounding rectangles location

In order to reduce the limitations of input images and to extract the correct flower boundary, an interactive method is provided for user to bind the query flower. First, a rectangle is drawn by user to bind an interested flower using mouse click and drag controls. After this operation, the system will detect four edge points of the

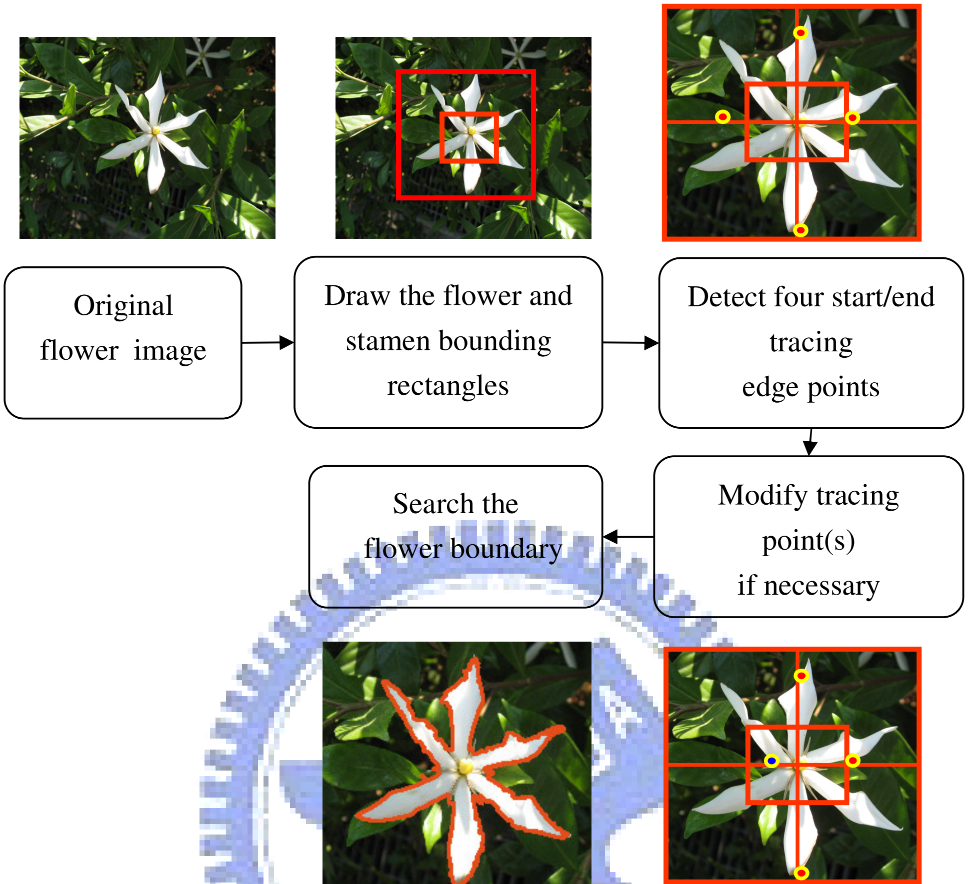



Fig. 6. Flow diagram of the segmentation steps.

flower automatically. Since, some flowers contain obvious stamen in the flower center and the color of stamen differs from the petal. This will cause a strong edge response in the stamen which may confuse us when detecting edge points. To treat this problem, a stamen bounding rectangle is located automatically, and the size is 1/9 of flower bounding rectangle. Then, the edge points within the stamen bounding rectangle are ignored when locating four flower edge points. Fig. 7 shows flowers with stamen parts and the located stamen bounding rectangle. Note that the stamen bounding rectangle actually contains the stamen part.

Fig. 7. Flowers with stamen parts and located bounding rectangles.

### 3.1.2 Start/end tracing edge points locating

In this subsection, we will describe the edge point detection method used to locate four edge points. These four edge points will be used as start/end tracing points for boundary searching. These four edge points will divide the flower boundary into four parts; we trace each part separately to speed up segmentation processing time. Before describing the method, the local cost of each pixel is defined as

$$\text{Local cost} = 1 + MG - G\,, \tag{1}$$

where $G$ denotes the gradient magnitude for the pixel and $MG$ denotes the maximum gradient magnitude of all pixels in the image. Here we add 1 to the local cost value let the cost have 1 at least in order to deal with the boundary searching algorithm (described in subsection 3.1.3) which finds the flower boundary based on minimum cost.

According to the definition of cost, we can see that a pixel with strong edge response has low cost. To compute the gradient magnitude of a pixel, we use Sobel operators (see Fig. 8). The approximation of the gradient magnitude $G$ for the pixel $P_{x,y}$ is defined as

$$G = \sqrt{G_x^{\ 2} + G_y^{\ 2}},$$

Where $G_x = (P_{x+1,y-1} + 2P_{x+1,y} + P_{x+1,y+1}) - (P_{x-1,y-1} + 2P_{x-1,y} + P_{x-1,y+1})$

and $G_y = (P_{x-1,y+1} + 2P_{x,y+1} + P_{x+1,y+1}) - (P_{x-1,y-1} + 2P_{x,y-1} + P_{x+1,y-1})$. (2)

$G_x$ is the magnitude of horizontal gradient and $G_y$ is the magnitude of horizontal gradient.

| | | |
|---|---|---|
| -1 | 0 | 1 |
| -2 | 0 | 2 |
| -1 | 0 | 1 |

| | | |
|---|---|---|
| -1 | -2 | -1 |
| 0 | 0 | 0 |
| 1 | 2 | 1 |

Fig. 8. Two Sobel operators $G_x$ and $G_y$.

Since the flower image is given in the RGB color coordinate, we compute gradient magnitude for each color band and calculate the maximum over the three respective outputs to produce a single valued local cost image.

According to the above definitions, the edge point detection method is described next. We scan vertical and horizontal lines in the middle of the flower bounding rectangle. One edge point for each scan line will be detected. On the flower bounding rectangle, five points are taken: the flower's center $P_0$ and the middle points $P_1, P_2, P_3, P_4$ of the four rectangle sides as shown in Fig. 9.

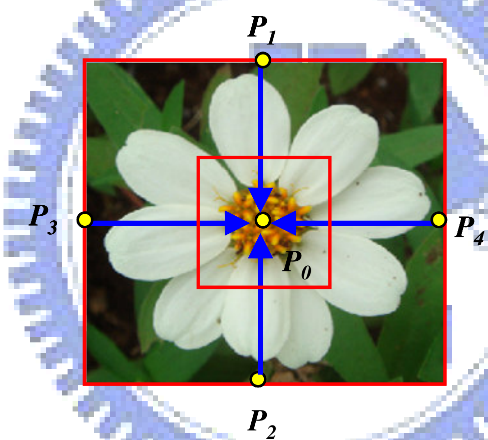



Fig. 9. Four scan lines in the flower bounding rectangle.

Then four profiles of local costs along the straight lines from $P_0$ to all of the four middle points are computed. Next, on each profile excluding the stamen area (as $P_3 \rightarrow P_5$, see Fig. 10), 5% minimum cost points are identified. A threshold is calculated by averaging the 5% minimum cost points' local cost values. If the local cost value of a point smaller than the threshold, it will be considered as a candidate edge point. The one closest to the border of the flower bounding rectangle is taken as the edge point (see $e_1$, $e_2$, $e_3$ and $e_4$ in Fig. 10). Note that for a small percentage of pictures, the method aboved may be failed to get correct start/end tracing edge points on each scan line. Fig. 11 is an example. The reason is that some flower image a stamen part is smaller than the defined one (1/9 of the flower rectangle region) which causing the real edge be excluded outside. In such a case, point is modified by user using a mouse to drag the point to the correct position.

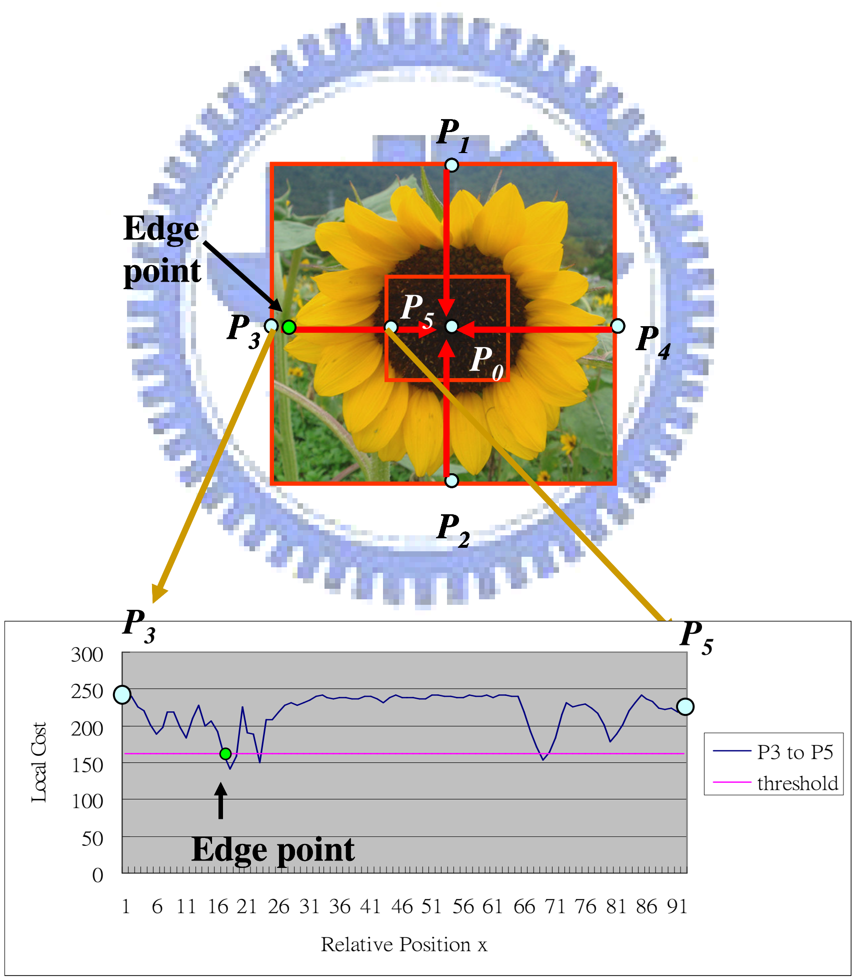
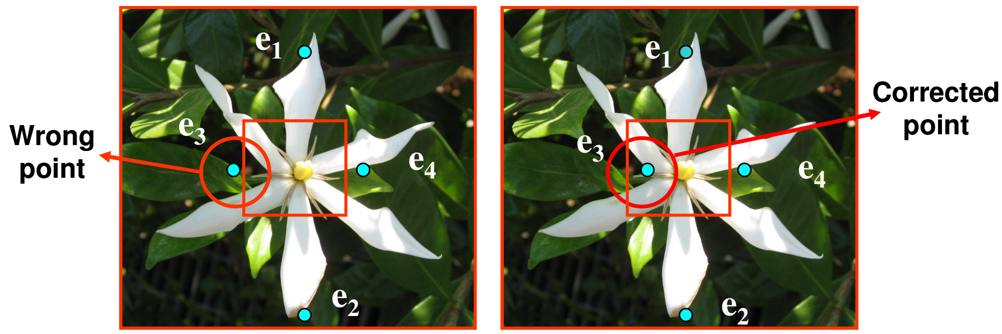



Fig. 10. An example of profile on sun flower.




(a) (b)

Fig. 11. An example of wrong edge detected. (a) Wrong detection result. (b) The wrong point corrected by user.

### 3.1.3 Boundary search algorithm

In this subsection, a boundary search algorithm will be described which divides the boundary searching into four parts to speed up segmentation process. After obtaining the four start/end tracing edge points, the flower boundary will be found next. The pixel within the flower bounding rectangle will be traced as a graph by considering each pixel as a vertex in the graph. Edges in the graph connect adjacent pixels. Each pixel is adjacent to 8 neighbors. the cost of an edge is defined from one pixel to its neighbor as the neighbor pixel's local cost.

The proposed algorithm modified the 2-D dynamic programming graph search algorithm proposed by Mortensen et al. [7]. The concept of average path cost is used in the proposed algorithm. The average cost is only considered on a partial average cost from the previous pixel and the next pixel to decide which direction to move. The partial average cost is updated by adding the average of the previous pixel cost and the next pixel cost. The boundary searching is divided into four parts and use 1/4 sub-region (see $R_1$, $R_2$, $R_3$ and $R_4$ in Fig. 12) of the flower bounding rectangle as the search region for each search to reduce the computation loading. The 4 pairs of edge points ($e_1$, $e_2$), ($e_2$, $e_3$), ($e_3$, $e_4$) and ($e_4$, $e_1$) are considered as the start and end tracing points for each search respectively. And the proposed algorithm will start from the start point and stop when the end point is reached. After 4 times boundary searching, a flower boundary is received (see the yellow curve in Fig. 12).The modified 2-D graph search algorithm is as follows:

Algorithm: 2-D graph search (Continued)

`Input:`

`s` Start pixel.
`f` End pixel.
`c(q,r)` Cost function for link between pixels q and r.

`Data Structures:`

`L` List of active pixels (i.e. pixels are not determined with the minimum average cost yet and will be chosen as candidates to expand at next step) sorted by average cost (initially empty).
`N(q)` Neighborhood set of q (contains 8 neighbors of pixel).
`e(q)` Boolean function indicating if q has been expanded/processed.
`g(q)` Cost function from start pixel to q.

`Output:`

`p` Pointers from each pixel indicating the minimum cost path.

```
Algorithm:

g(s)=0; L=L+{s};       { Add active list with zero cost start pixel.}
while (L≠ϕ and not e(f))do begin{ While still points to expend.}
    q←min(L);          { Remove minimum cost pixel q from active list.}
    e(q)=TRUE;         { Mark q as expanded (i.e. processed).}
    for each r∈N(q) with not e(r) do begin
        temp=(g(q)+c(q,r))/2;{ Compute average cost to neighbor.}
        if r∈L and temp<g(r) then
            L=L-{r};           { Remove higher cost neighbor from list.}
        if r∉L then begin  { If neighbor not on list, }
            g(r)=temp;         { assign neighbor's average cost, }
            p(r)=q;            { set (or reset) back pointer, }
            L=L+{r};           { and place on (or return to) active list.}
        end
    end
end
```

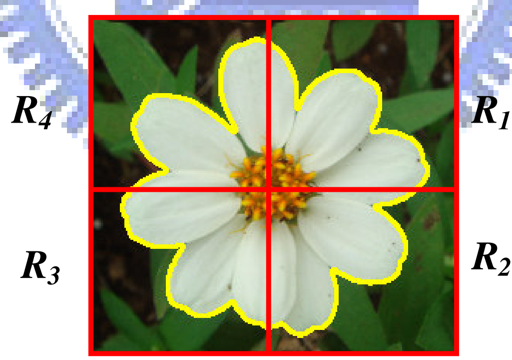



Fig. 12. Four sub-regions of the flower bounding rectangle.

## 3.2 Feature extraction

The most obvious features in flowers are shape and color. 3 shape features and

11 color features are used and the first six color features are proposed by Saitoh et al. [6].

### 3.2.1 Shape features

For the convenient explanation, we call the internal region bounded by the extracted boundary as flower region. Based upon the extracted boundary, the gravity center $(g_x, g_y)$ of the flower region is defined to be $g_x = \frac{1}{N}\sum_{i=1}^{N} x_i$ and $g_y = \frac{1}{N}\sum_{i=1}^{N} y_i$, where N is the number of pixels in the boundary and $x_i$ and $y_i$ are x-y coordinates of the $i^{th}$ pixel in the boundary. Then the three shape features are described as follows:

F1： A ratio which indicates the relevance sharpness of the petals to the flower. The feature is defined as

$$\mathrm{F1} = \frac{R_{10}}{R_{90}} \tag{3}$$

where $R_{90}$ is the average of the top 10% longest distances and $R_{10}$ is the average of the top 10% shortest distances from the boundary to the gravity.

F2： The average distance from flower points in boundary to the gravity center. The feature is defined as

$$\mathrm{F2} = \frac{1}{N}\sum_{i} D_i \tag{4}$$

$$D_i = \frac{\sqrt{(x_i - g_x)^2 + (y_i - g_y)^2} - R_{10}}{R_{90} - R_{10}} \quad (5)$$

$$\begin{cases} D_i = 1 \ \ if \ \ \sqrt{(x_i - g_x)^2 + (y_i - g_y)^2} > R_{90} \\ D_i = 0 \ \ if \ \ \sqrt{(x_i - g_x)^2 + (y_i - g_y)^2} < R_{10} \end{cases} \quad (6)$$

where $D_i$ is the normalized distance of the $i^{th}$ pixel from the boundary to gravity, N and $(x_i, y_i)$ is defined above.

F3： Roundness which indicates that how much the boundary shape is closer to a circle. The feature is defined as

$$\mathrm{F2} = \frac{4\pi S}{L^2} \quad (7)$$

where L is the boundary length of the flower and S is the total number of pixels in the flower region.

When the feature is close to 1, it means that the flower shape is close to a circle.

### 3.2.2 Color features

Flower images are represented in the RGB color model. Owing to that the images are taken in different days and weather, the RGB values are coverted into the HSV (hue, saturation and value) values [8] in order to reduce the illumination variation. The primary, secondary and thirdly flower colors and the stamen color are taken as our features. First, the HS space is divided into 12x6 cells represented by $C_i$, $i = 1, 2, \ldots, 72$.( see Fig. 13). Let $P(C_i)$ be the probability of pixels with colors in cell $C_i$. Let $DC_1$, $DC_2$, $DC_3$ be the cells with the largest, the second largest and the third largest probabilities, respectively. The coordinates of the centers of the three cells $DC_1$, $DC_2$, $DC_3$ and the three corresponding probabilities are taken as our features. The details are described as follows:

$$C_i = (H_i, S_i) \qquad i = 1, 2, \ldots, 72$$

$$(x_i, y_i) = (S_i \cos H_i, S_i \sin H_i) \tag{8}$$

where $H_i$ and $S_i$ is the center of the color cell $C_i$ which represented for the color cell, and $(x_i, y_i)$ is the coordinates of the color cell $C_i$.

F4： the $x$ coordinate of the center of the color cell $DC_1$,

F5： the $y$ coordinate of the center of the color cell $DC_1$,

F6： the probability, $P(C_1)$, of the color cell $DC_1$,

F7： the $x$ coordinate of the center of the color cell $DC_2$,

F8： the $y$ coordinate of the center of the color cell $DC_2$,

F9： the probability, $P(C_2)$, of the color cell $DC_2$,

F10： the $x$ coordinate of the center of the color cell $DC_3$,

F11： the $y$ coordinate of the center of the color cell $DC_3$,

F12： the probability, $P(C_3)$, of the color cell $DC_3$.

In addition, stamen color of flowers is also an important characteristic in recognition. So, the color distribution in stamen area which is a square area with length of 2/3 petal length around the gravity is taken as our feature, where petal length is defined by $R_{90}$. Let $SC$ be the cell with the largest probability of pixels with color in stamen area. The center of the cell $SC$, is taken as our features. The details are described as follows:

F13： the $x$ coordinate of the center of the color cell $SC$,

F14： the $y$ coordinate of the center of the color cell $SC$.

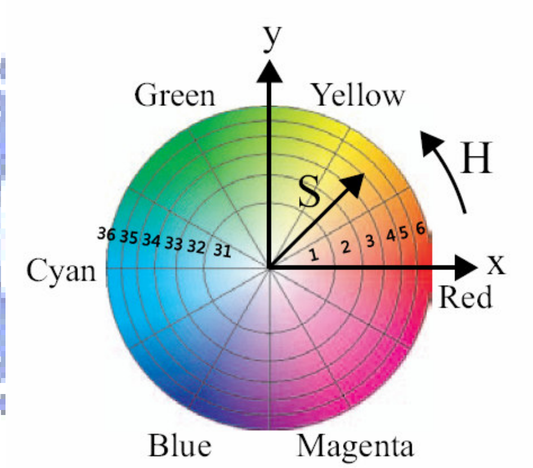



Fig. 13. The HS space divided into 12x6 cells.

## 3.3 Recognition

Since the variation of flowers, a classification method does not used for grouping species and recognition. For example, orchid may contain different colors in petal. If these instances' features are grouped together, it may mess up this color feature's discrimination ability. Hence, recognition is accomplished using the k-nearest neighbor algorithm.

### 3.3.1 K-nearest neighbor algorithm

In the k-nearest neighbor algorithm, each flower image is considered as an individual class. First, the recognition system calculates the distance between query and all flower images in the database. All features are normalized to [0, 1] and the initial weights for F1 to F14 are set according to the separability of each feature. The initialized weight $w_i$ of the $i^{th}$ feature is defined as

$$w_i = \frac{P_i}{\sum_{k=1}^{14} P_k} \qquad i=1, 2, \ldots, 14. \tag{9}$$

where $P_k$ denotes the accuracy rate of the $k^{th}$ feature.

The distance is calculated by Euclidean distance

$$d(i,q) = \frac{\sum_{k=1}^{14} w_k \left| f_{i,k} - f_{q,k} \right|}{\sum_{k=1}^{14} weight_k} \tag{10}$$

where $f_{i,k}$ denotes the $k^{th}$ feature of instance $i$, $f_{q,k}$ denotes the $k^{th}$ feature of the query image , and $w_k$ denotes the weight of the $k^{th}$ feature.

Then, the recognition system ranks the distances and the top 20 nearest neighbors are returned to the query image. These returned images are called candidates. Here, a retrieval feedback mechanism [9] is also provided for user to adjust the weights automatically.

### 3.3.2 Relevence feedback algorithm

Since the result does not always meet the user's expectation, a feedback mechanism is provided to adjust the weights of all features in response to each user's subjective point of view. A relevence feedback algorithm is provided to automatically adjust the weight of each feature according to the current retrieval result and the user's response. After revising weights, system returns new candidates to user.

Steps for relevence feedback algorithm are presented as follows:

1. The user chooses $u$ alike flower images, $A_1, A_2, ..., A_u$, and $t$ unlike flower images, $N_1, N_2, ..., N_t$, from candidates.

2. Calculating the mean of the $k^{th}$ features of alike images and the query image, $M_k$, as follows:

$$M_k = \frac{1}{u+1}(f_{q,k} + \sum_{i=1}^{u} f_{A_i,k}) \qquad k = 1,\ 2,\ ...,\ 14 \tag{11}$$

3. Calculating the standard deviation on each feature for both alike and unlike images, $\sigma_k^{alike}, \sigma_k^{unlike}$, as follows:

$$\sigma_k^{alike} = \sqrt{\frac{(f_{q,k} - M_k)^2 + \sum_{i=1}^{u}(f_{A_i,k} - M_k)^2}{u+1}} \qquad k = 1,\ 2,\ ...,\ 14 \tag{12}$$

and

$$\sigma_k^{unlike} = \sqrt{\frac{\sum_{i=1}^{t}(f_{N_i,k} - M_k)^2}{t}} \qquad k = 1,\ 2,\ ...,\ 14 \tag{13}$$

4. Revising the $k^{th}$ feature weight $w_k$

$$w_k = \frac{\sigma_k^{unlike}}{\sigma_k^{alike}} \qquad k = 1,\ 2,\ ...,\ 14 \tag{14}$$

5. Running recognition engine again with new weights to get new candidate images.

Note that when some images are selected as alike and unlike images, there will exist some features, which are similar in those alike images, these features in the unlike images will be dissimilar to the alike images. These features have high separability and should have high weights. From the above definition, we can see that for the $k^{th}$ feature, if $\sigma_k^{alike}$ is small and $\sigma_k^{unlike}$ is large, this means that the feature is similar to each other in alike-class but dissimilar with unlike-class. Then, the feature's weight will be increased according to the formula (10). Increasing this feature's weight will let retrieved images farther from those images in unlike-class and closer to the alike-ones. After adjusting all of the weights, the new result will be closer to what the user really wants [9].

# CHAPTER 4

# EXPERIMENTAL RESULTS

In this chapter, experiments are conducted to evaluate the performance of the proposed method. First, the separability of each feature is showed. Next, recognition results are presented based on two databases which are our database of 348 flower images consisting of 24 species and Zou-Nagy's [4] database of 612 flower images consisting of 102 species. The performance will be compared with Zou-Nagy's method. Besides, two databases are combined to a bigger species database consisting of 114 species and the performance of the proposed method will be shown. Finally, the effectiveness of using the feedback mechanism will be shown.

We take one feature each time in recognition and receive the top 5 accuracy rate of each feature. Top 5 accuracy means that one of the $1^{st}$, $2^{nd}$, $3^{rd}$, $4^{th}$ and $5^{th}$ candidate species is the same species of the query one. The separability of each feature in our database is shown in Table 1. Note that every database has its own initialized weight for each feature.

Each flower image in our database is resized to 400 x 300 pixels. Each flower image in the database is taken as our query image and the remaining 347 flower images are considered as training data. The result is shown in Table 2. The average processing time is 4.15 seconds which includes user interaction and boundary segmentation time.

Table 1. The initialized weight for each feature in our flower database.

| Feature | Top 5(%) Accuracy | Top5 / Sum of top 5 accuracy | Initialized weight |
|---|---|---|---|
| F1 | 58.05 | 0.0708 | 7.1 |
| F2 | 39.94 | 0.0487 | 4.9 |
| F3 | 60.63 | 0.0740 | 7.4 |
| F4 | 66.38 | 0.0810 | 8.0 |
| F5 | 67.24 | 0.0820 | 8.2 |
| F6 | 58.62 | 0.0715 | 7.2 |
| F7 | 62.93 | 0.0768 | 7.7 |
| F8 | 64.94 | 0.0792 | 7.9 |
| F9 | 43.97 | 0.0536 | 5.4 |
| F10 | 62.93 | 0.0768 | 7.7 |
| F11 | 60.34 | 0.0736 | 7.4 |
| F12 | 45.11 | 0.0550 | 5.5 |
| F13 | 63.51 | 0.0775 | 7.7 |
| F14 | 65.23 | 0.0796 | 8.0 |
| Sum of Accuracy | 819.82 | | |

Table 2. Performance on our own database.

| Time (s) | Recognition rate (%) | | | | | number of images | number of species |
|---|---|---|---|---|---|---|---|
| | Top 1 | Top 2 | Top 3 | Top 4 | Top 5 | | |
| 4.15 | 89.1 | 95.4 | 97.4 | 99.4 | 100 | 348 | 24 |

The proposed method is also conducted on Zou-Nagy's [4] database collected from [10]. All images have size of 300 x 240 pixels. Each species contains six images. Some pictures are quite out of focus, and several pictures contain multiple, tiny, overlapping flowers. The result is shown in Table 3. From the table, we can see that the processing time of our method is twice faster than Zou-Nagy's method. And the Top-3 recognition rate (96.2%) is much higher than Zou-Nagy's (79%) with 8.5 seconds user's rose-curve adjustments before labeling the flower to the class.

Table 3. Performance on Zou-Nagy's database

| | Time (s) | Recognition rate (%) | | | | |
|---|---|---|---|---|---|---|
| | | Top 1 | Top 2 | Top 3 | Top 4 | Top 5 |
| Our method | 4.2 | 74.4 | 92.2 | 96.2 | 97.4 | 98.4 |
| Zou-Nagy's method (before labeling) | 8.5 | 52 | | 79 | | |
| Zou-Nagy's method | 10.7 | | | 93 | | |

These two databases are combined into a bigger one and the performance of the proposed method is shown in Table 4. 684 flower images collected from 114 distinct species are used to test the proposed algorithm.

Finally, an example is further presented to show the effectiveness of using the feedback mechanism (see Figs. 14 and 15). In Fig. 14, the first glory bush is used as the query image and the others are the twenty most similar images search by the proposed method without user's feedback. Similar color and shape flowers are

Table 4. Performance on combined database

| Time (s) | Recognition rate (%) | | | | | number of images | number of species |
|---|---|---|---|---|---|---|---|
| | Top-1 | Top-2 | Top-3 | Top-4 | Top-5 | | |
| 4.5 | 74 | 90.1 | 94.2 | 96.2 | 97.2 | 684 | 114 |

retrieved. If the third image is selected as the alike one and the fifth image as the unlike one, and then the feedback mechanism is applied, a better retrieval result is obtained (see Fig. 15). We have 18 images of the glory bush in the database and 5 images of the glory bush are retrieved initially. After feedback mechanism, 16 images of the glory bush will be retrieved and the candidate images from top 1 to top 10 is the same species with the query one consequently.

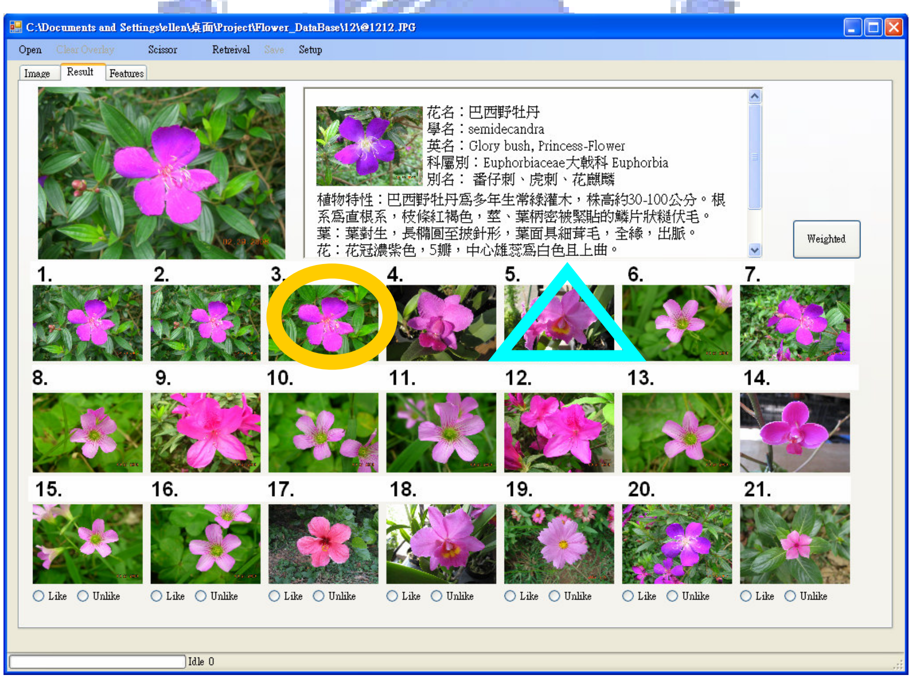



Fig. 14. The initial retrieval result of the glory bush without using the feedback mechanism.

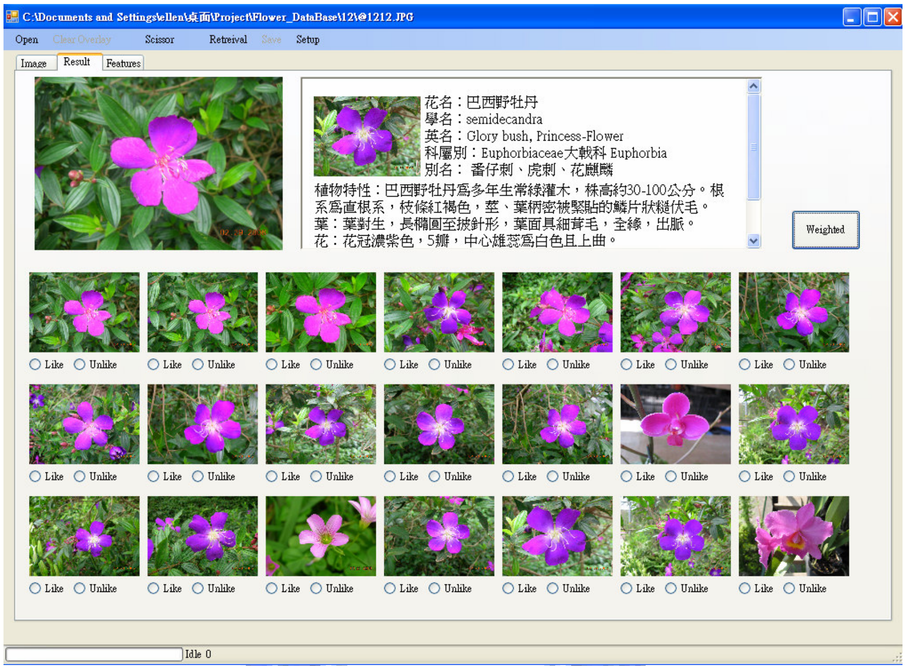



Fig. 15. The retrieval result of the glory bush after using feedback mechanism.

A recognition system written in C# language on a PC is built. Figs. 16 and 17 show the main interface. Fig. 16 is the interface for user’s interactive operations and Fig. 17 is the interface for recognition result. After the candidate flower images are retrieved, if there is one image looks like the query one then clicking on the image, the information about the flower (name, scientific name and other relative species information) will be showed to the user. Therefore, the system only need one image in the retrieval results which is the same species with the query one that can recognize the query flower correctly.

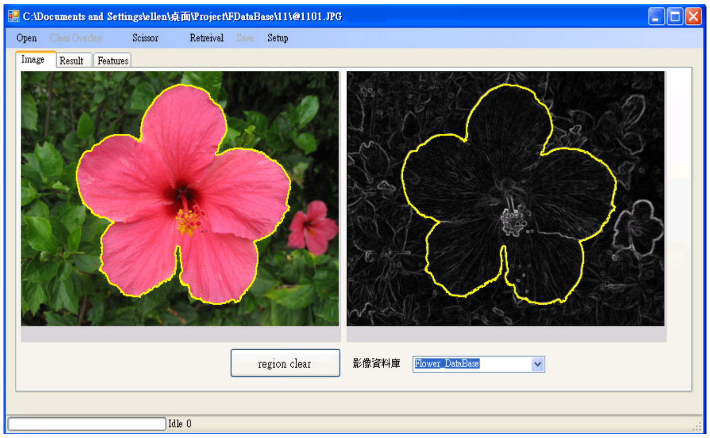



Fig. 16. Interface for user interactive operations.

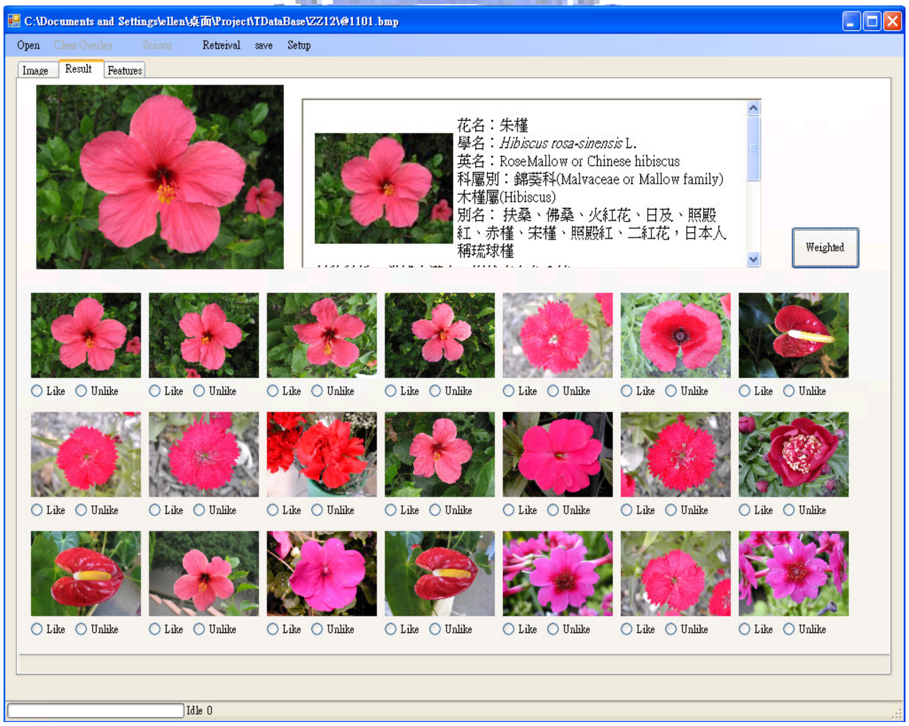



Fig. 17. Interface for recognition result.

# CHAPTER 5

# CONCLUSIONS

In this thesis, we have presented an interactive method for flower recognition which can deal with photographs with fewer constraints. First, a simple user interactive process is applied to bind the searching area for segmentation. Then, the system detects the edge points automatically and runs the segmentation engine. Fourteen features are extracted from the segmented flower image. In the end, a similarity measure is provided to do recognition. Furthermore, to make the recognition result more close to the user's need, a relevance feedback algorithm is provided to automatically determine the importance of each feature. Based on the algorithm, a system can be built to recognize flower with photographs. Experiments designed for assessment of performance showed the accuracy and efficiency of the proposed method.

# REFERENCES

[1] T. Saitoh and T. Kaneko, "Automatic Recognition of Wild Flowers," *Proc. International Conference on Pattern Recognition,* Vol. 2, pp. 507-510, 2000.

[2] M. Das, R. Manmatha, and E-M. Riseman, "Indexing Flower Patent Images Using Domain Knowledge," *IEEE Intelligent Systems*, Vol. 14, pp. 24-33, 1999.

[3] A. Hong, G. Chen, J. Li, Z. Chi, and D. Zhang, "A Flower Image Retrieval Method Based on ROI Feature," *Journal of Zhejiang University (Science)* Vol. 5, pp. 764–772, 2004.

[4] J. Zou and G. Nagy, "Evaluation of Model-Based Interactive Flower Recognition," *Proc. International Conference on Pattern Recognition*, Vol. 2, pp. 311-314, 2004.

[5] M-E. Niksback and A. Zisserman, "A Visual Vocabulary for flower classification," *Proc. Computer Vision and Pattern Recognition*, Vol. 2, pp. 1447-1454, 2006.

[6] T. Saitoh, K. Aoki and T. Kaneko, "Automatic Recognition of Blooming Flowers," *Proc. International Conference on Pattern Recognition*, Vol. 1, pp. 27-30, 2004.

[7] E-N. Mortensen and W-A. Barrett, "Intelligent Scissors for Image Composition," *Proc. Computer Graphics and Interactive Techniques*, pp. 191-198, 1995.

[8] HSL and HSV, http://en.wikipedia.org/wiki/HSI_color_space.

[9] J. L. Shih and L. H. Chen, "A Context-based approach for color image retrieval,"

*International Journal of Pattern Recognition and Artificial Intelligence,* Vol. 16, pp. 239-255, 2002.

[10] http://www.ecse.rpi.edu/doclab/flowers.