Herbal Leave Recognition System Based on Dirichlet Laplacian Eigenvalues

Mahmoud Elgamal
The Custodian of the Two Holy Mosques
Institute for Hajj and Omra Research,
Umm Al -Qura University
Makkah, Saudi Arabia
Email: maelgamal@uqu.edu.sa

Mahmoud Youness R. Alaidy
College Of Computer
Computer Eng., Dept, Qassim Univ.,
Buryadh, Saudi Arbia
Email: aaiedy@qu.edu.sa

***Abstract*—Identifying and recognition of herbal plant green leaves is essential in botanical study. In[8] Thai herb leaf image recognition system used for recognition of leaves with accuracy of 93.29%, in this paper, we propose a recognition system of leaves based on the eigenvalues of Dirichlet Laplacian that used to generate three different sets of features for shape analysis and classification in binary images[4]. First leaf images are preprocessed to remove unwanted background, converted to binary form; used to build the images database, finally Queries made on the system. The correct classification rates without noise is 100% and with noise is $\sim 90\%$.**

Keywords: Eigenvalues, Finite difference method, Curve descriptor, Binary image classification, noise, leave recognition.

I. Introduction

Shape recognition is the field of computer vision which addresses the problem of finding out whether a query shape lies or not in a shape database, up to a certain invariance. Most shape recognition methods simply sort shapes from the database along some similarity measure to the query shape. Shape analysis is a key component in object recognition, matching, registration and analysis. A shape description method generates a feature vector that will uniquely characterize the silhouette of the object. This vector should in many cases, be translation-, rotation-, and size-invariant. Depending on the application at hand, a certain level of robustness and tolerance to shape deformation and noise is also required. As an important application of shape recognition, leave recognition which has significant attention in botanical study. However, by far the most popular classification in Loncaric [6] of shape techniques divides the different methods into two groups: boundary methods and global methods. Boundary methods treat the boundary or exterior points of the shape, while global methods deal with the interior points of the object. There is no clear consensus which method or category of methods works best. Each method seems to give a good result in some applications and fail in some others or in presence of noise. The method presented in this paper is a numerical non-preserving global method that attempts to use the ratios of eigenvalues of the Dirichlet Laplacian operator of a certain shape as the feature vector.
The paper structured as follows, a brief mathematical overview of the model and the evaluation of the eigenvalues in sections(II and III), feature set evaluation in section(IV), algorithmic implementation in section(V), and finally the simulation.

II. The Dirichlet Laplacian eigenvalues

Let Ω be a bounded domain in $\mathbb{R}^n$, $n \geq 2$. Consider the eigenvalue problem for the Laplace operator with Dirichlet boundary condition,

$$\begin{cases} -\Delta u = \lambda u & \text{in } \Omega, \\ u|_{\partial\Omega} = 0. \end{cases} \tag{1}$$

Here in (1), $\Delta = \dfrac{\partial^2}{\partial x^2} + \dfrac{\partial^2}{\partial y^2}$ is the Laplacian. As is well known, the Dirichlet Laplacian (Laplacian with Dirichlet boundary condition) has diverse applications in science and engineering, and we refer to Cureton and Kuttler [2] and Kuttler [5] on the detailed study of Dirichlet Laplacian in two dimensional polygons.

Let us denote the eigenvalues by $\lambda_1(\Omega), \lambda_2(\Omega), \cdots$, (we will sometimes omit explicit dependence on Ω when speaking about generic domain), where

$$0 < \lambda_1 < \lambda_2 \leq \lambda_3 \leq \cdots \longrightarrow \infty. \tag{2}$$

It is also well known that the eigenvalues of the Dirichlet Laplacian are preserved if the underlying domain Ω is translated or rotated (see Courant and Hilbert [1]). In the next section it will be discussed how to evaluate the eigenvalues.

III. Computation of the eigenvalues

A. Finite difference method

In order to evaluate the approximate numerical solution of (1), there are several methods. Among those we choose the finite difference scheme which was first proposed in Pólya [11]. The scheme is to replace (1) by the recursive formula

$$\frac{u_{i+1,j} + u_{i,j+1} + u_{i-1,j} + u_{i,j-1} - 4u_{i,j}}{h^2} = -\lambda u_{i,j}. \tag{3}$$

Here the domain Ω is divided into squares of side h, and u_{ij} is the value of the eigenfunction corresponding to λ at the lattice point (ih, jh) (see Figure 3.1). This scheme can be written in compact form as

$$\mathcal{L}u = \lambda u,$$

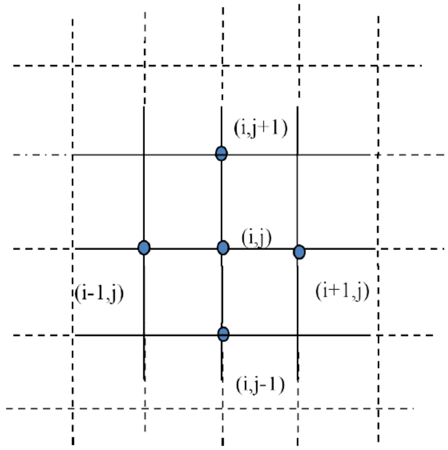


Fig. 3.1: five-stencil approximation for the Laplacian.

where

$$\mathcal{L} = \frac{1}{h^2}\begin{bmatrix} A & I_n & 0 & \cdots & 0 \\ I_n & A & I_n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & A \end{bmatrix}_{n^2\times n^2}$$

and

$$A = \begin{bmatrix} -4 & 1 & 0 & \cdots & 0 \\ 1 & -4 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -4 \end{bmatrix}_{n\times n}.$$

Here, n is inversely proportional to h and accounts for the size of the domain Ω. The eigenvalues $\lambda_1', \lambda_2', \cdots$ of this finite dimensional problem provide, in general, lower bounds for $\lambda_1, \lambda_2, \cdots$ (cf. [10]).

IV. Features Generation and Evaluation

For a given binary image Ω, ([12] and [4]) proposed the following three feature sets based on the above described eigenvalues

$$F_1(\Omega) \equiv \left\{\left(\frac{\lambda_1}{\lambda_2}, \frac{\lambda_1}{\lambda_3}, \frac{\lambda_1}{\lambda_4}, \cdots, \frac{\lambda_1}{\lambda_n}\right)\right\}, \quad (4a)$$

$$F_2(\Omega) \equiv \left\{\left(\frac{\lambda_1}{\lambda_2}, \frac{\lambda_2}{\lambda_3}, \frac{\lambda_3}{\lambda_4}, \cdots, \frac{\lambda_{n-1}}{\lambda_n}\right)\right\}, \quad (4b)$$

$$F_3(\Omega) \equiv \left\{\left(\frac{\lambda_1}{\lambda_2} - \frac{d_1}{d_2}, \frac{\lambda_1}{\lambda_3} - \frac{d_1}{d_3}, \cdots, \frac{\lambda_1}{\lambda_n} - \frac{d_1}{d_n}\right)\right\} \quad (4c)$$

Here n counts the number of the desired features to be used for the recognition scheme, and $d_1 < d_2 \leq d_3 \leq \cdots \leq d_n$ are the first n eigenvalues (counting multiplicity) of a disk. All three features are obviously size-invariant [4]. The F_1 features were first proposed by Zuliani et al. [12]. The values of $F_1(\Omega)$ and $F_2(\Omega)$ are in the unit cube, while those of $F_3(\Omega)$ are between ± 1 a useful range when using neural networks. This later descriptor is a good measure of the deviation of Ω from a disk. The optimal number of features n depends on the problem being addressed and is determined experimentally.

To test the consistency of these feature sets for a given image class, their tolerance to noise, experiments are conducted and the simulation discussed in next section.

V. Implementation

In this section, an algorithm is developed based on the above discussion to evaluate F_1, F_2, and F_3 of a all images, also for the query image; compare to get the minimum value of the norm as explained in fig. (5.1). Figure (5.2) shows the flow diagram of the GUI

Input: query image i_q,
Output: the most similar image (i_{out}), from the image database(IDB).

Step 1:
- a) Read the images ($i = 1, \cdots, n$) from the stored Database images(IDB)
- b) Convert all the images in *IDB* to binary images.
- c) Evaluate F_1^i, F_2^i, & F_3^i for each images of the *IDB*.
- d) Calculate the norms of F_1^i, F_2^i, & F_3^i; $NF_1^i := \|F_1^i\|$, $NF_2^i := \|F_2^i\|$, and $NF_3^i := \|F_3^i\|$

Step 2: Let the query image q, repeat Step 1:(b - d) to calculate the norms of $NF_1^q := \|F_1^q\|$, $NF_2^q := \|F_2^q\|$, and $NF_3^q := \|F_3^q\|$

Step 3: For i := 1 to N; (N = no. of images),
- a) Calculate $NdF_j^i := (NF_j^i - NF_j^q)$, $(j = 1, 2, 3)$.
- b) Store the values of NdF_j^i in an array A

Step 4: Find $\min(A)$ and the corresponding index; which is the index of the retrieved image.

Step 5: Display the images of the query image; i_q and the retrieved image; i_{out} along side.

Step 6: Add noise to the images and apply steps 1-5.

Fig. 5.1: Leaf recognition system algorithm.

VI. Simulations

In this part we focus on testing the above described algorithm, it was implemented using Matlab version 8.0; figure(6.2). The leave images was refined by removing background using Adobe Photoshop as shown in figure() A particular feature should have a fairly constant value for all images from a particular class. The consistency of a feature can be measured using its standard deviation from the mean for that image class. To test the consistency of the three feature sets being used, experiments were conducted on different images with and without noise.

Example 1: image resolution at 256 pixels

a) noise = 0.0

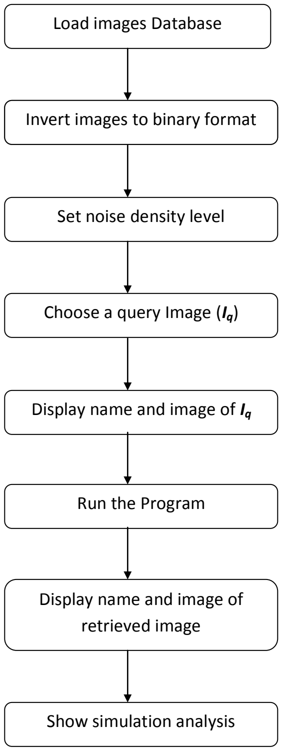


Fig. 5.2: flow-diagram of the algorithm GUI-interface.

Fig. 6.1: Leaf background removal.

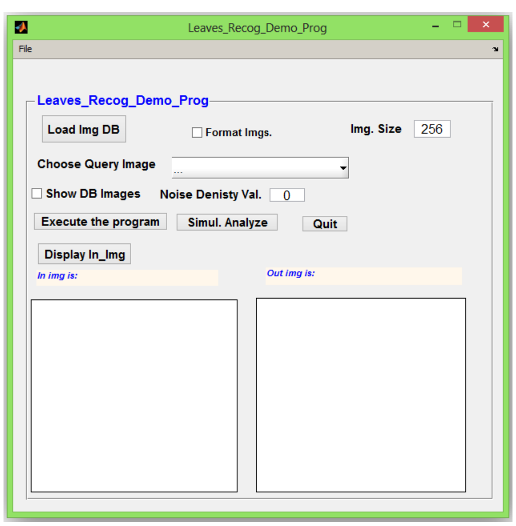
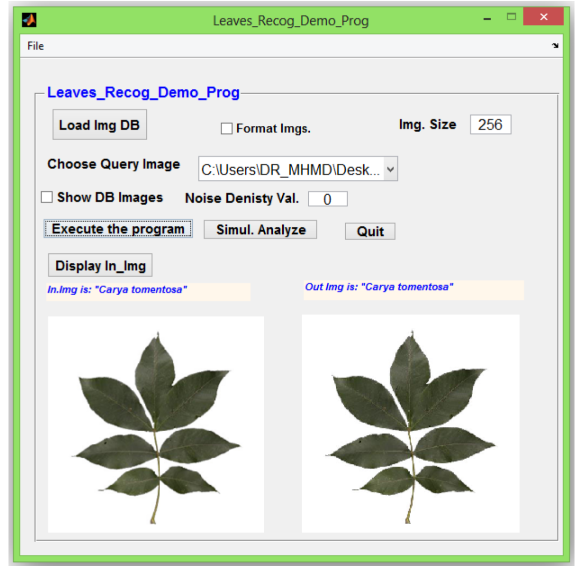
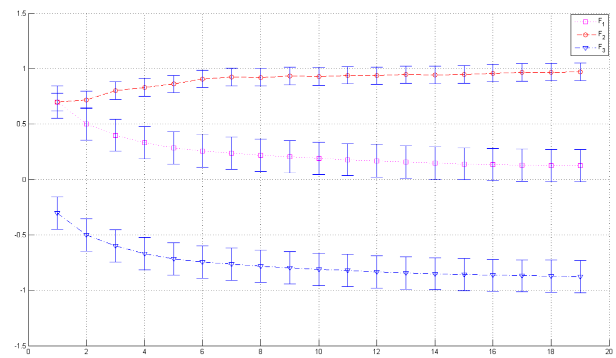


Fig. 6.2: GUI of Leave recognition System.



(a) input and retrieved images.



(b) average and standard deviation of the first 25 features from F_1, F_2, and F_3.

Fig. 6.3: output at noise=0.0.

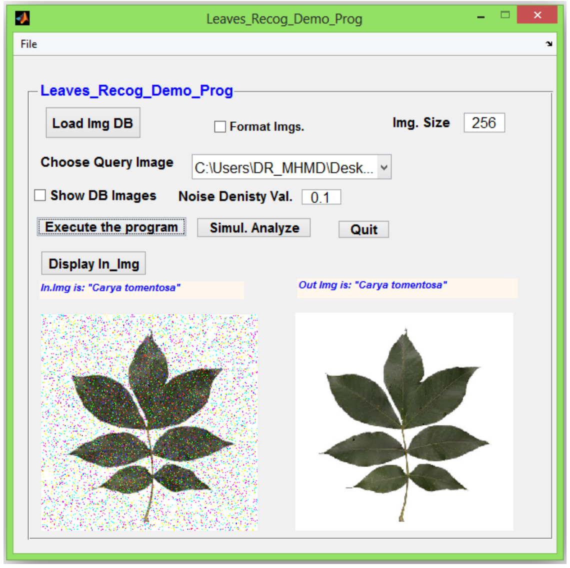
b) noise = 0.1, figure (6.4(b)) shows almost identical values of F_1 for the images without noise and the others with noise

Example 2: image resolution at 128 pixels

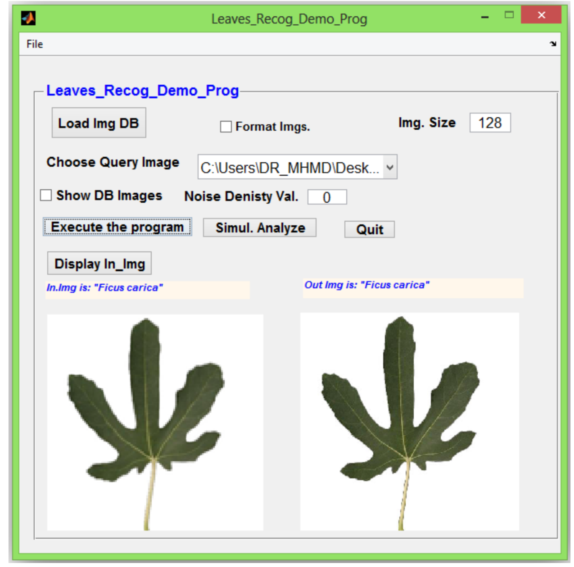
a) noise = 0.0

b) noise = 0.1, figure (6.6(b)) shows almost identical values of F_1 for the images without noise and the others with noise

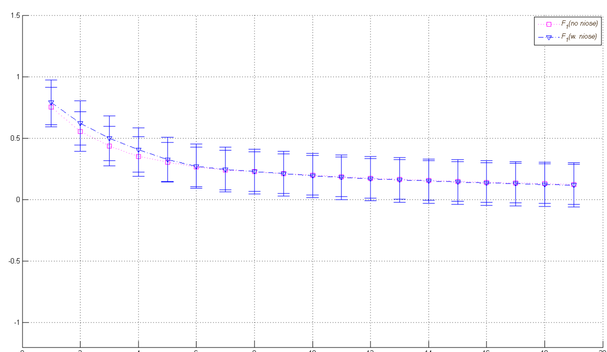
Example 3:



(a) input and retrieved images.

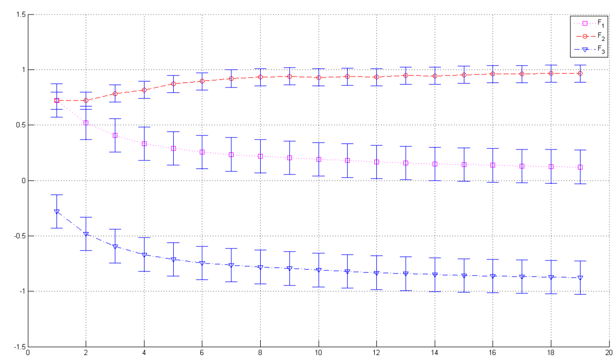


(a) input and retrieved images.



(b) average and standard deviation of F_1 for images with/without noise.

Fig. 6.4: output at noise=0.1.



(b) average and standard deviation of the first 25features from F_1, F_2, and F_3.

Fig. 6.5: output at noise=0.0.

A. Performance Evaluation

We evaluate the performance of the proposed method in terms of *precision*, *recall*, and *accuracy* see[7]. Image retrieval system has the goal to retrieve relevant images while not retrieving irrelevant ones. The measures of performance used in image retrieval borrowd from the field of *document information retrieval* and are based on two primary figures of merit: *precision* and *recall*.

- *Precision*(P) is the number of relevant documents retrieved by the system divided by the total number of documents retrieved(i.e., true positives plus false alarms).

$$P = \frac{TP}{TP + FP} \tag{5}$$

- *Recall*(R) is the number of relevant documents retrieved by the system divided by the total number of relevant documents in the data base(which should have been retrieved).

$$R = \frac{TP}{TP + FN} \tag{6}$$

Precision can be interpreted as a measure of exactness, whereas recall provides a measure of completeness.

- *Accuracy*(A) is the probability that the retrieval is correctly performed

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

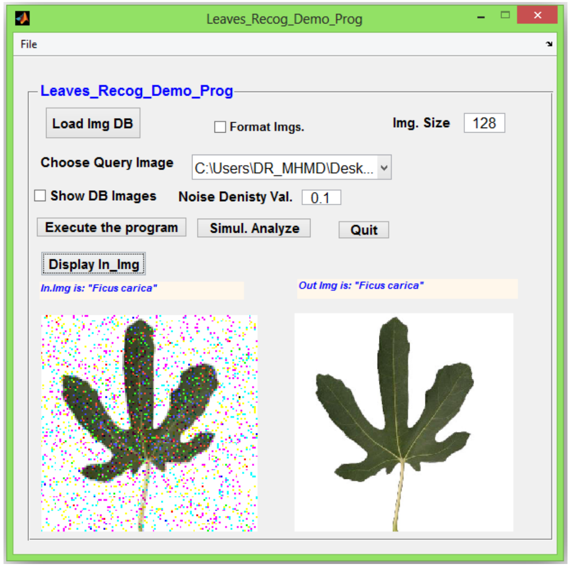
where,

TP(True Positive) - correctly classified positive,

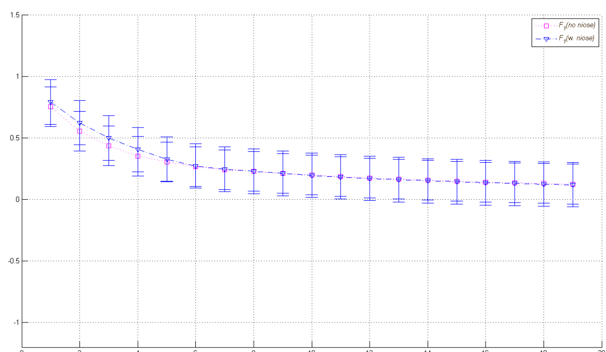
TN(True Negative) - correctly classified negative,

FP(False Positive) - incorrectly classified negative, and

FN(False Negative) - incorrectly classified positive.



(a) input and retrieved images.



(b) average and standard deviation of F_1 for images with/without noise.

Fig. 6.7: Mis-recognition at 0.2 noise level and 64 image resolution.

	TP	*TN*	*FP*	*FN*	*P(%)*	*R(%)*	*A(%)*
Noise = 0	200	50	0	0	100	100	100
Noise = 0.1	200	50	15	11	93%	94.8%	90.6%

Fig. 6.8: Performance of the used techniques.

VII. Conclusion

The three sets of features based on the eigenvalues of Dirichlet Laplacian, was used to develop a user friendly leave recognition system. The system used successfully to classify images with a high degree of accuracy and using a relatively small number of features. At first it was run on leave database images for the purpose of recognition. Initially without noise and the obtained result was good and then a noise was add to the images but still showed a good result but when increasing the noise level the input and output was different.

References

[1] R. Courant and D. Hilbert, *Methods of Mathematical Physics*, Second ed., Interscience Publishers, New York, 1965.

[2] L. M. Cureton and J. R. Kuttler, Eigenvalues of the Laplacian on regular polygons and polygons resulting from their dissection, J. Sound Vib., **220**(1999), 83-98.

[3] Leafsnap: An Electronic Field Guide,"http://leafsnap.com/species/", 2013.

[4] M. A. Khabou, L. Hermi and M. B. H. Rhouma, Shape Recognition Using Eigenvalues of the Dirichlet Laplacian, Pattern Recognition, **40**(2007), 141-153.

[5] J. R. Kuttler and V. G. Sigillito, Eigenvalues of the Laplacian in two dimensions, SIAM Rev. **26**(2)(1984), 163-193.

[6] S. Loncaric, "A survey of shape analysis techniques", Pattern recognition **31**(1998), 983-1001.

[7] Olson, David L.; and Delen, Dursun (2008); "Advanced Data Mining Techniques", Springer, 1st edition (February 1, 2008), page 138, ISBN 3-540-76916-1.

[8] C. Pornpanomchai, S. Rimdusit, P. Tanasap and C. Chaiyod, "Thai Herb Leaf Image Recognition System (THLIRS)", Kasetsart journal(natural science), 45 : 551 - 562 (2011).

[9] S. Pramanik, S. K. Bandyopadhyay, D. Bhattacharyya, and T. Kim, Identification of Plant Using Leaf Image Analysis ,2010.

[10] G. Pólya, Sur une interprétation de la méthode des différences finies qui peut fournir des bornes supérieures ou infrieures, C. R. Acad. Sci. Paris **235**(1952), 995-997.

[11] G. Pólya, Esimates for eigenvalues, in: Studies in Mtahematics and Mechanics Presented to Richard von Mises, Academic Press, New York, (1954), 200-207.

[12] M. Zuliani, C. Kenny, S. Bhagavathy and B. S. Manjunath, Drums and curve descriptors, UCSB Visition Research Lab Preprint, 2004.