Leaf Analysis for Plant Recognition

Aparajita Sahay, Min Chen
Computing and Software Systems, School of STEM
University of Washington Bothell
Bothell, WA98011, USA
{sahay87, minchen2}@uw.edu

Abstract—**Plants are essential resources for nature and people's lives. Plant recognition provides valuable information for plant research and development, and has great impact on environmental protection and exploration. This paper presents a leaf analysis system for plant identification, which consists of three main components. First, given a leaf image, a preprocessing step is conducted for noise reduction. Second, the feature extraction component identifies representative features and computes scale invariant feature descriptors. Third, the matching plant species are identified and returned using a weighted K-nearest neighbor search algorithm. The system is implemented as a Windows phone app and is tested on the LeafSnapdataset[8], an electronic field guide developed by Columbia University and University of Maryland with different combinations of species at various orientations, scales and levels of brightness. The experimental results demonstrate the effectiveness of our proposed framework in plant recognition.**

Keywords-plant recognition; leaf analysis; SIFT; KNN search

I. INTRODUCTION

Gathering information about unknown plants manually, such as going to local nursery, reading books or researching on Internet, is time consuming. Automatic plant identification system is therefore of great importance tofacilitate and speed up the process. Various plant properties can be used to identify the species, such as the dimension, branch shape and flower type [1][2][7][11]. Among them, leaf is one of the most representative properties, containing important information about the taxonomic identity of a plant[3].Moreover, leaves are present on the plants for several months in a year, whereas flowers and fruits may remain only several weeks. This is why most plant identification tools are based on Content-Based Image Retrieval techniques on leaf image databases [14].

In the literature, several plant recognition systems have been developed using leaf detection[8][18]. For instance, nearest neighbor approach is applied in [8]to compare a query leaf image with the labeled plant database using histogram intersection as the distance metric. In [18], multi-layered perceptron (MLP) is used to differentiate leaf boundaries from veins and Active Shape Model (ASM) is applied to construct a leaf shape model that classifies query image and returns leaf detection results. However, plant recognition remains a challenging task because of the following reasons: 1) real world photographs of leaves can have many variations such as background noise, scale, rotation, color and brightness; and 2)members within the same plant families may only have very subtle feature difference.

In this paper, we present an automated plant identification system with the following goals.

1. **Usability**: Develop a mobile application thatusers can conveniently carry to take photographs of leaves.
2. **Robustness**: Variations in leaf images such as rotation and scaling should not affect the outcome of the result.
3. **Portability**: Expose key capabilities of this application as a Web Service to facilitate faster future development across platforms.

II. PROPOSED FRAMEWORK

The proposed plant identification process is summarized in Fig. 1.First, the query image is pre-processed to remove unwanted noise. The second stage includes identifying interesting points and computing SIFT descriptors. Finally weighted K-Nearest Neighbor (KNN) search compares features of query image with images in database and returns top three matching plant species as the result.

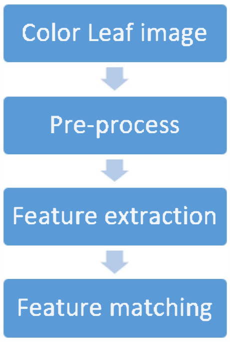

Figure 1. Overview of the proposed plant identification process

A. 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 [5]. 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 [3].

To remove noise from the image, a morphological operation called erode-dilation [9]is applied on the gray scale image.Erosion method erodes away the boundary of the foreground object. It is useful for removing small noises from the image. After erosion method, noise is removed.However; the boundary of the original image is reduced in the process. Therefore, after eroding we recover the original boundary by applying dilation algorithm on the image.

In essence, preprocessing step removes unwanted noise from the image and reduces number of false positive regions.

B. Feature Extraction

For feature extraction, instead of using general image shape features such as contour and edge [12][15], local features [13]are used, which represent highly localized information from small areas of an image and are defined around interest points. It is assumed that interest points detected through the same method on similar images will produce similar local features. In another word, local feature algorithms depend on the idea of determining some interest points in the image and implementing a local analysis on them, rather than looking at the image as a whole [16].

In leaf analysis application, local feature of an image was computed using Scale Invariant Feature Transform (SIFT) as proposed by David G Lowe [10]. SIFT feature is invariant to rotation and scale [8].

The scale invariant feature transform consists of the following major stages of computation:

1. Scale-space extrema detection: The first stage of SIFT key point detection is to identify locations and scales that can be repeatedly assigned under differing views of the same object. Detecting locations that are invariant to scale change of the image can be accomplished by searching for stable features across all possible scales, using a continuous function of scale known as scale space [17]. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation.

2. Key point localization: At each candidate location, a detailed model is fit to determine location and scale. Key points are selected based on measures of their stability [6].

3. Orientation assignment: One or more orientations are assigned to each key point location based on local image gradient directions. All future operations are performed on image data that have been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations.

4. Key point descriptor: The local image gradients are measured at the selected scale in the region around each key point. The key points are detected over a complete range of scales so that small local features are available for matching small and highly occluded objects, while large key points perform well for images subject to noise and blur. Their computation is efficient, so that several thousand key points can be extracted from a typical image [10]. Fig. 2 shows SIFT for two example images.

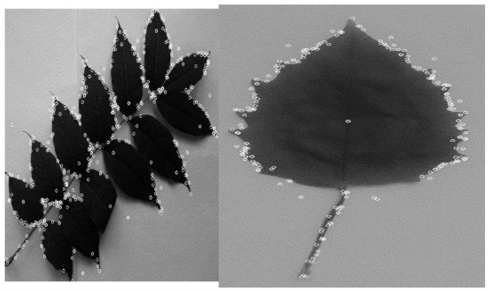

Figure 2. SIFT of example images

C. Image Matching

To identify the leaf, an instance-based learning method called nearest neighbor search was performed on feature extracted from query image to all images in database.

The image database consists of features extracted from 325 field images. The goal of nearest neighbor search in leaf recognition system is to find the best candidate match for each SIFT point in query image among all the key point descriptors from training images. The nearest neighbor is defined as the key point with minimum Manhattan distance for the invariant descriptor vector.

KNN search can be useful for in-the-field plant recognition as the database of known species can be easily updated on-the-fly to include more new samples for KNN search. KNN is also based on distance measures in feature space.But instead of comparing feature set to a class representative value, it compares to all samples of the feature training set and selects the first *k* closest ones.

To reduce the outliers in KNN matching, the following technique is used:

In KNN, *k*is set to be 2, that is to find the two nearest SIFT points in each training image instance for each SIFT point in query image.

- In an image database, only those SIFT points close to SIFT point of the query image are considered (i.e., if the distance from query image's SIFT point is less than 300 in our study). We call these SIFT points as match points.
- Ratio test is applied on match points. As per ratio test, match points are considered as valid match points only if the distance ratio between the first and second match point is greater than 0.75. Ratio test eliminates 90% of false matches and 5% of correct matches [10].
- System accuracy is further improved by using weighted KNN search. In weighted KNN, the closer neighbors are weighted more than farther ones using the distance weighted function [4][18].

As shown in Eq. (1), the algorithm is to assign weights w_i to the neighbors $f(x_i)$ $(i = 1 \ldots k)$. Here, w_i is defined in Eq. (2), which is bigger when $f(x_i)$ is closer.

$$f(x_q) = \frac{\sum_{i=1}^{k} w_i f(x_i)}{\sum_{i=1}^{k} w_i} \quad (1)$$

$$w_i = \frac{1}{d(f(x_q), f(x_i))^2} \quad (2)$$

III. SYSTEM IMPLEMENTATION

As discussed earlier, our system is a mobile application that helps users identify plants from photographs of their leaves.The system architecture is shown in Fig. 3.

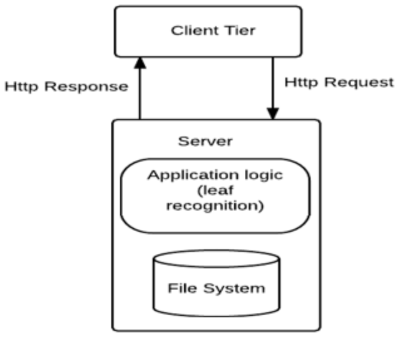

Figure 3: High Level system architecture of the system

The recognition engine consists of a backend server that accepts input images from various front-end clients. Currently, client is the application on the Windows phone device whose GUI is shown in Fig.4.

Figure 4. Example application GUI

The connection between the GUI on the mobile device and the identification system on the server is done by a RESTful web service which offers an attractive scalable computing architecture. The data exchange between the clients (i.e. the windows phone devices) and web service is ensured via HTTP protocol. Web service consists of a library called leaf core that implements leaf identification logic and file system that stores leaf data set.

To execute the system, web service is started first toprocess all the images in file system, currently 450 images, and their extract features. The user selects a query leaf image with a white colored background from windows phone and sends the request to the web service. Web service accepts the image in byte array and serializes the image from byte array to image. The leaf core logic web service then processes the query image and extractsits relevant features. Identification is done by running a weighted K nearest neighbors search on database images. Top 3 most closely matched leaf species are returned back to the client.The readme file and all the source code can be accessed at:
https://drive.google.com/folderview?id=0B1RcF1odDtbPZy1WLXdmSHdBVjQ&usp=sharing

IV. EXPERIMENTAL RESULTS

In order to test the system, dataset from LeafSnap[8] is used for experiment. The data set consists of different combinations of species to analyze the accuracy of algorithm. Each leaf species has twenty-leaf image in different orientation, scale and brightness. Fig. 5 shows 15 different categories of leaves.

To test the accuracy of the system,20% of the leavesin the database are used as query images. Two widely used measurement metrics, precision and recall, are used to compute the accuracy of the system, which are defined in Eq. (3)

$$Recall = \frac{Truepositive}{TruePositive + Falsenegative}$$
$$Precision = \frac{TruePositive}{Truepositive + falsepositive} \quad (3)$$

Figure 5. 15 categories of plant leaves

Three different testing setups are used as below:

- Intra-family: totally 120 images are tested, which are from all six species (20 images per species) from the same family called betula. By averaging the testing results, the precision is 76.7% and recall is 96.0%.
- Inter-family: totally 300 images are tested, which are from all 15 different leaf families (20 images per family) shown in Fig. 5. In each family, the 20 images selected belong to one species. We achived average precision of 80.7% and recall 75.6%.
- Mixed: Similar to the inter-family setup, totally 300 images are tested, which are from all 15 different leaf

families with 20 images per family. However, within each family, instead of having all 20 images coming from the same species, in mixed setup, these 20 images belong to different species in the same family. As a result, the average precision is 72.8% and recall is 68.0%.

The results are illustrated in Fig. 6. As we can see, the performance in mixed is relatively lower than the other two setups. It is due to the relatively complex setup in the experiment. In contrast, the performance of intra-family, especially the recall, is much better than others because the testing scope is limited.

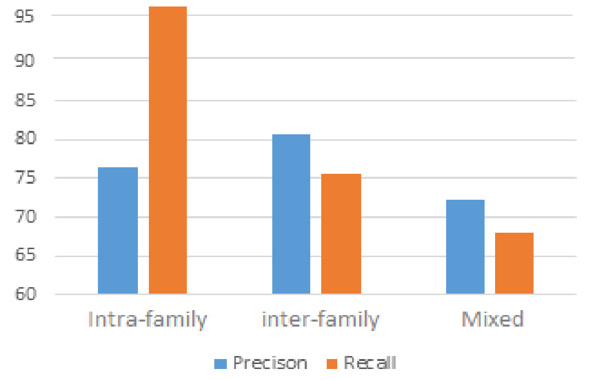

Figure 6: Performance analysis of intra-family, inter-family and mixed

We than compare the performance of weighted KNN search with that of KNN. In this experiment, 100 images are randomly selected for testing. The averaged precision and recall values are listed in Table I. As we can see Weighted KNN outperforms KNN in both metrics with over 15% improvements.

TABLE I. Performance comparison between weighted and normal KNNalgorithm

	KNN	Weighted KNN
Precision	66.3	85.2
Recall	50.9	67.5

V. CONCLUSIONS

In this paper, we present a Windows phone application for leaf species identification. The leaf identification process makes this application useful to amateur stakeholders as well as experts.

The system relies on computer vision for several key aspects, such as preprocessing the image by removing noise from the image, extracting scale and rotation invariant feature set from the image, and retrieving the most similar species matches using weighted KNN search on a data set of labelled images. Using this application, user can photograph a single leaf on a white colored background and submit it as the input query image. Application will then analyze features of the leaf and identify the plant species at real time.

REFERENCES

[1] Arora, A., et al.,"A plant identification system using shape and morphological features on segmented leaflets: Team IITK, CLEF 2012," *CLEF (Online Working Notes/Labs/Workshop)*, 2012.

[2] Bama, B. S. et al.,"Content based leaf image retrieval (CBLIR) using shape, color and texture features," *Indian Journal of Computer Science and Engineering* 2.2,pp. 202-211, 2011.

[3] Caballero, C.& Aranda, M. C.,"Plant species identification using leaf image retrieval," *Proceedings of the ACM International Conference on Image and Video Retrieval*, pp. 327-334, 2010.

[4] Gou, J., et al.,"A new distance-weighted k-nearest neighbor classifier," *Journal of Information and Computational Science* 9.6, pp. 1429-1436, 2012.

[5] Grand-Brochier, M., Vacavant, A., Cerutti, G., Bianchi, K., & Tougne, L.,"Comparative study of segmentation methods for tree leaves extraction,"*Proceedings of the ACM International Workshop on Video and Image Ground Truth in Computer Vision Applications*, no. 7, 2013

[6] Gwo, C. Y.& Wei, C. H., "Plant identification through images: Using feature extraction of key points on leaf contours," *Applications in plant sciences*, vol. 1, no.11, 2013.

[7] Kadir, A.,"Leaf Identification Using Fourier Descriptors and Other Shape Features," *Gate to Computer Vision and Pattern Recognition*, vol. 1, no. 1, pp. 3-7, 2015.

[8] Kumar, N., Belhumeur, P. N., Biswas, A., Jacobs, D. W., Kress, W. J., Lopez, I. C., & Soares, J. V.,"Leafsnap: a computer vision system for automatic plant species identification,"*Computer Vision–ECCV*, pp. 502-516, 2012.

[9] Li, Y., Zhu, Q., Cao, Y., & Wang, C., "A leaf vein extraction method based on snakes technique,"*Proceedings of IEEE International Conference on Neural Networks and Brain*, vol. 2, pp. 885-888, 2005.

[10] Lowe, D. G.,"Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no.2,pp. 91-110, 2004.

[11] Manh, A. G., Rabatel, G., Assemat, L., & Aldon, M. J., "AE—automation and emerging technologies: weed leaf image segmentation by deformable templates," *Journal of agricultural engineering research*, vol. 80, no. 2, pp. 139-146, 2001.

[12] Mouine, S., Yahiaoui, I., & Verroust-Blondet, A., "Advanced shape context for plant species identification using leaf image retrieval," *Proceedings of the ACM international conference on multimedia retrieval*, no. 49, 2012.

[13] Prasad, S., Kudiri, K. M., & Tripathi, R. C, "Relative sub-image based features for leaf recognition using support vector machine,"*Proceedings of the ACM International Conference on Communication, Computing & Security*, pp. 343-346, 2011.

[14] Quan, L., Tan, P., Zeng, G., Yuan, L., Wang, J., & Kang, S. B., "Image-based plant modeling,"*ACM Transactions on Graphics (TOG)*, vol. 25, no. 3, pp. 599-604, 2006.

[15] Rui, Y., She, A. C., & Huang, T. S.,"Modified Fourier descriptors for shape representation-a practical approach," *Proceedings of First International Workshop on Image Databases and Multi Media Search*,pp. 22-23, 1996.

[16] Shabanzade, M., Zahedi, M., & Aghvami, S. A., "Combination of local descriptors and global features for leaf recognition," *Signal & Image Processing: An International Journal*, vol. 2, no. 3, pp. 23-31, 2011.

[17] Witkin, A. P. "Scale-space filtering," *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 1019-1022, 1983.

[18] Xia, C., Lee, J. M., Li, Y., Song, Y. H., Chung, B. K., & Chon, T. S.,"Plant leaf detection using modified active shape models," *Biosystems Engineering*, vol. 116, no.1, pp. 23-3, 2013.