Complex Background Leaf-based Plant Identification Method Based on Interactive Segmentation and Kernel Descriptor

Thi-Lan Le
MICA, Hanoi Univ. of Technology
VietNam
Thi-Lan.Le@mica.edu.vn

Duong-Nam Duong
Hanoi Univ. of Science and Technology
VietNam
duongksclck55@gmail.com

Van-Toi Nguyen
Thai Nguyen Univ.
Thai Nguyen, VietNam
Van-Toi.Nguyen@mica.edu.vn

Hai Vu
MICA, Hanoi Univ. of Technology
VietNam
Hai.Vu@mica.edu.vn

Van-Nam Hoang
MICA, Hanoi Univ. of Technology
VietNam
Van-Nam Hoang@mica.edu.vn

Thi Thanh-Nhan Nguyen
Thai Nguyen Univ.
Thai Nguyen, VietNam
Thanh-Nhan.Nguyen@mica.edu.vn

ABSTRACT

This paper presents a plant identification method from the images of the simple leaf with complex background. In order to extract leaf from the image, we firstly develop an interactive image segmentation for mobile device with tactile screen. This allows to separate the leaf region from the complex background image in few manipulations. Then, we extract the kernel descriptor from the leaf region to build leaf representation. Since the leaf images may be taken at different scale and rotation levels, we propose two improvements in kernel descriptor extraction that makes the kernel descriptor to be robust to scale and rotation. Experiments carried out on a subset of ImageClef 2013 show an important increase in performance compared to the original kernel descriptor and automatic image segmentation.

Categories and Subject Descriptors

I.4.8 [**Image processing and Computer Vision**]: Scene Analysis - Object recognition

General Terms

Algorithms

Keywords

plant identification; kernel descriptor; complex background

1. INTRODUCTION

EMR'15, June 23–26, 2015, Shanghai, China.

DOI: http://dx.doi.org/10.1145/2764873.2764877.

Plant identification is a process that aims at matching a given specimen plant to a known taxon. This is a difficult and time consuming task even for the botanist experts. Recently, with the advanced research in computer vision community, a number of works have been dedicated to plant identification based on images. The main aim of these works is to build the computer aided for plant identification using image processing techniques. Among different plant's organs, the leaf is most widely used. However, most current works for leaf-based plant identification base on the assumption that leaf image is taken in relatively simple background [11]. In this paper, we propose a method for plant identification for complex background images of the simple leaf. The contributions of this paper are two-fold. Firstly, we develop an interactive image segmentation for mobile device with tactile screen. This allows to separate the leaf region from the complex background image in few manipulations. This allows to tackle with one of the most difficult tasks in computer vision (image segmentation) in reality. Secondly, we propose two improvements in kernel descriptor (KDES) extraction in order to build a scale and rotation invariant descriptor for leaf identification.

2. RELATED WORKS

There are two types of leaf: compound leaf (composed of a number of leaflets) and simple leaf. While most of the works for simple leaf focus on extracting the shape and the venation of the leaf [7], the works for compound leaf try to find out the configuration of the leaflet [4]. Since our work focuses on simple leaf, in this section, we make an analysis of the works dedicated to the simple leaf. Since the leaf image can be taken in complex background, the first step of the leaf-based plant identification is leaf image segmentation. However, the image segmentation is challenging issue of the computer vision. Therefore, the most current works for simple leaf-based plant identification bases on the assumption that the leaf image is taken in a relatively simple background. Very fews works handle the complex background leaf segmentation [13], [6]. In [13], the authors have proposed a method named automatic marker-controlled watershed segmentation method. However, this method bases on the assumption that the image is bimodal. Recently, in [6], the authors have proposed an automatic image segmentation for leaf images. For this, a segmentation step based on a light polygonal leaf model is first performed, and later used to guide the evolution of an active contour. However, this work is limited to simple and lobed tree leaves. While working with plant identification based on leaf image, the most crucial part is leaf representation, in which, we need to define and to decide the robust features in order to obtain the best leaf representation. A number of features have been proposed for leaf representation that has been shown in ImageClef2013 [5]. However, the performance of the proposed methods have been reduced when applying in complex background images.

Recently, Le et al. [7] have proposed to use kernel descriptor presented by Bo et al. [2] for leaf identification. This descriptor outperforms the state of the art descriptors such as SURF. However, this work is limited to simple background images and the kernel descriptor is not invariant to scale and rotation.

3. COMPLEX BACKGROUND LEAF-BASED PLANT IDENTIFICATION METHOD

3.1 Overview

Fig. 1 shows the diagram of our method that consists of three main steps: interactive leaf segmentation, leaf representation and classification based on kernel descriptor.

Interactive image segmentation: An interactive segmentation method aims to separate a leaf from the complex background. The fact that conventional automatic segmentation methods such as thresholds are often ineffective for image captured in a natural/out-door scene. Some others robust segmentation methods such as graph-cut require a lot of computational time. Therefore, fully automatic leaf segmentation is an unsolved problems. In this work, we exploit a well-known technique that is watershed algorithms [10] for interactive segmentation. Using this technique, an interested leaf is feasibly extracted from the complex background.

Leaf representation: This step takes a leaf region image (hereafter called *image*, for short) as input and returns a descriptor of the candidate leaf. It is composed of three sub-steps:(1) Pixel-level feature extraction (2) Patch-level feature extraction and (3) Image-level feature extraction.

Leaf classification: Once the leaf is represented by a descriptor vector, any classifier could be applied for classification task. In this paper, following the strategy originally proposed [2], we will use Multi-class SVM. In following sections, we focus to present in detail the interactive segmentation and the three steps of leaf representation.

3.2 Interactive segmentation

The proposed interactive segmentation scheme is shown in Fig. 2

The watershed method has been used in image segmentation since the early 90's. It is analogous to placing a water source in each regional minimum and then flooding the relief. Barriers are built to prevent the different sources meeting. In interactive segmentation works, the markers based watershed approach is often employed with two markers: an

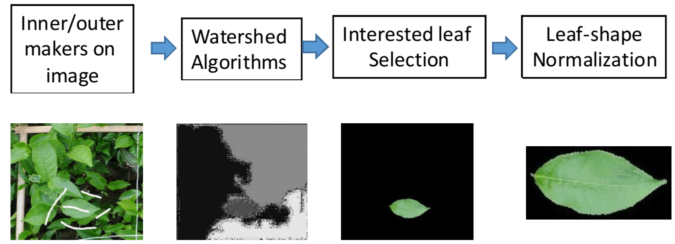

Figure 2: The proposed interactive segmentation scheme

external and internal markers. An external marker coarsely marks boundary in order to quickly separate the foreground regions from background ones. Internal markers point out seed (a center area) of the interested regions. Meyer et al. use markers in conjunction with region growing to prevent over segmentation [?]. An Otsu threshold is employed to automatically create markers. The practical implementations show that the higher the marker precision, the better the segmentation results. In our work, we define manual makers thanks to the convenience of the user's touch on a smartphone's screen. At the first step, user manually mark regions around an interested leaf. More makers created, more regions have been segmented. At the second step, the regions with same gradient levels as marker regions are selected. User selects the region including the interested leaf. It is simple point out the region of the interest.

3.3 Leaf-based plant identification

3.3.1 Pixel-level features extraction

According to [2], a number of features can be computed at pixel level such as pixel value, texture, and gradient. In [7], it was argued that gradient is the best feature for leaf recognition; accordingly, in this paper we use gradient at pixel level.

The gradient vector at a pixel z is defined by its magnitude $m(z)$ and orientation $\theta(z)$. In [2], the orientation $\widetilde{\theta}(z)$ is defined as following:

$$\widetilde{\theta}(z) = [sin(\theta(z)) \; cos(\theta(z))] \qquad (1)$$

3.3.2 Patch-level features extraction

The patch-level feature extraction consists of two task. The first task aims at generating a set of patches from image and the second one is to calculate the patch feature. In [7], the authors generate the uniform patches from leaf image that makes the kernel descriptor is not invariant to image size. Moreover, in path-level feature extraction, the value of gradient angle is not normalized. As results, the kernel descriptor is not invariant in rotation. In this paper, we propose to use the adaptive size of patches and to normalize the gradient angle in order to build a kernel descriptor that is invariant to the scale and rotation.

Generate a set of patches from image with adaptive size
As discussed, in the original work [7], the authors generated patches with a fixed size for all images in the dataset with different resolutions. For low resolution images, the number of generated patches will be very limited, producing a poor representation of the image. Beside, the feature vec-

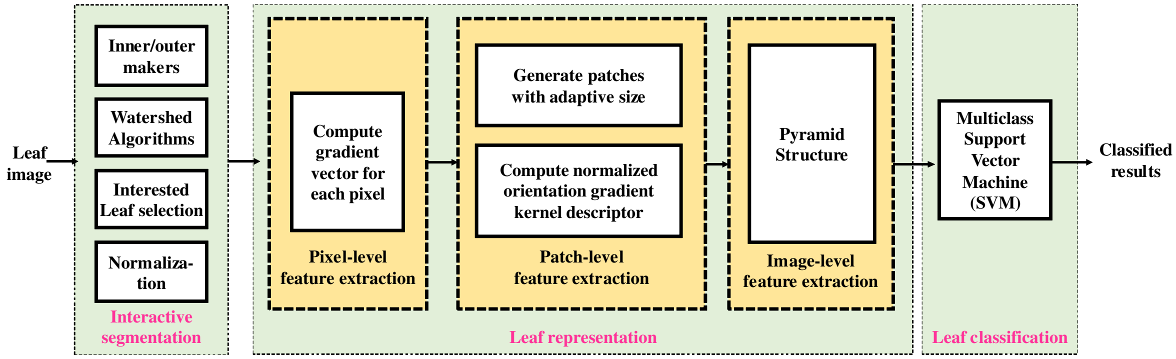

Figure 1: Overview of our method consisting of three main steps: interactive leaf segmentation, leaf representation and classification.

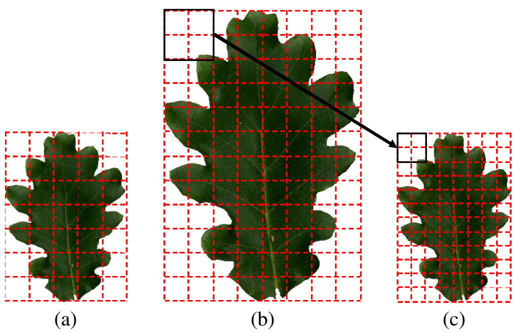

Figure 3: (a,b) two images of the same leaf with different sizes are divided using uniform patch; (b,c): two images of the same leaf with different sizes are divided using adaptive patch.

tors of two images of the same leaf at two scales will be highly different. As a consequence, the original KDES is not invariant to scale change. Fig. 3 (a,b) illustrate two images of the same leaf with different sizes are divided using uniform patch. The above analysis motivates us to make an adaptive patch size in order to get a similar number of patches along both horizontal axis and vertical axis. Suppose that the given number of patches is $np_x \times np_y$ (np_x patches along horizontal axe and np_y patches along vertical axe). The number of grid cells $ngrid_x \times ngrid_y$ is defined as: $ngrid_x = np_x + 1, ngrid_y = np_y + 1$. With an image has size of $w \times h$, the adaptive grid cell size along horizontal axe $gridsize_x = \frac{w}{ngrid_x}$ and the adaptive grid cell size along vertical axe $gridsize_y = \frac{h}{ngrid_y}$. The adaptive patch has the size of $patchsize_x \times patchsize_y$ where $patchsize_x = 2gridsize_x$ and $patchsize_y = 2gridsize_y$. A patch is constructed from 4 cells of the grid. The overlap of two adjacent patches along horizontal axe or along vertical axe is a region of two cells of the grid. By this way, the size of patches is directly proportional to the size of the image. Fig. 3(b,c) illustrates the advantage of the proposed adaptive patch and the representation of image based on patch-level features will be robust to scale change.

Compute patch-level feature

Patch-level features are computed based on the idea of kernel method. Derived from match kernel representing the similarity of two patches, we can extract feature vector for the patch using approximative patch-level feature map, given a designed patch level match kernel function.

The gradient match kernel is constructed from three kernels that are gradient magnitude kernel $k_{\widetilde{m}}$, orientation kernel k_o and position kernel k_p. In [2], gradient match kernel is defined as follows:

$$K_{gradient}(P,Q) = \sum_{z\in P}\sum_{z'\in Q} k_{\widetilde{m}}(z,z')k_o(\widetilde{\theta}(z),\widetilde{\theta}(z'))k_p(z,z') \quad (2)$$

where P and Q are patches of two different images that we need to measure the similarity. z and z' denote the 2D position of a pixel in the image patch P and Q respectively. $\theta(z)$ and $\theta(z')$ are gradient orientations at pixel z and z' in the patch P and Q respectively.

Using directly the gradient orientation $\widetilde{\theta}(z)$ in orientation kernel, the patch level features extracted from the match kernel will not be invariant to rotation. We then propose to normalize gradient orientation before applying in match kernel. Specifically, inspired from the idea of SIFT descriptor [8], we compute a dominant orientation of the patch and normalize all gradient vectors to this orientation. We propose two ways to determine the dominant orientation $\overline{\theta}(P)$ of the patch P. First, we use the dominant orientation of the patch as proposed in [8]. Second, we compute a vector sum of all the gradient vectors in the patch. The normalized gradient angle of a pixel z in P thus becomes:

$$\omega(z) = \theta(z) - \overline{\theta}(P) \quad (3)$$

Then according (Eq.1), the normalized orientation of a gradient vector will be:

$$\widetilde{\omega}(z) = [sin(\omega(z))\ cos(\omega(z))] \quad (4)$$

Finally, we define the gradient match kernel with normalized orientation as follows:

$$K_{gradient}(P,Q) = \sum_{z\in P}\sum_{z'\in Q} k_{\widetilde{m}}(z,z')k_o(\widetilde{\omega}(z),\widetilde{\omega}(z'))k_p(z,z') \quad (5)$$

Let $\varphi_o(.)$ and $\varphi_p(.)$ the feature maps for the gradient orientation kernel k_o and position kernel k_p respectively. Then, the approximative feature over image patch P is constructed as:

$$\overline{F}_{gradient}(P) = \sum_{z \in P} \widetilde{m}(z)\phi_o(\widetilde{\omega}(z)) \otimes \phi_p(z) \tag{6}$$

where $\otimes$ is the Kronecker product, $\phi_o(\widetilde{\omega}(z))$ and $\phi_p(z)$ are approximative feature maps for the kernel k_o and k_p, respectively. The approximative feature maps are computed based on a basis method of kernel descriptor. The basic idea of representation based on kernel methods is to compute approximate explicit feature map for kernel match function [9, 12, 3].

3.3.3 Image-level features extraction

Once patch-level features are computed for each patch, the remaining work is computing a feature vector representing the whole image. To do this, a spatial pyramid structure dividing the image into cells using horizontal and vertical lines at several layers. In the following, we present in detail each step to build the final descriptor of the image.

Let C be a cell that has a set of patch-level features $X = \{x_1, ..., x_p\}$ then the feature map on this set of vectors is defined as:

$$\overline{\phi}_S(X) = \frac{1}{|X|}\sum_{x \in X} \phi(x) \tag{7}$$

Where $\phi(x)$ is approximative feature maps for the kernel $k(x, y)$.

Given an image, let L be the number of spatial layers to be considered. In our case $L = 3$. The number of cells at layer l-th is (n_l). $X(l, t)$ is set of patch-level features falling within the spatial cell (l, t) (cell t-th in the l-th level). A patch is fallen in a cell when its centroid belong to the cell. The feature map on the pyramid structure is:

$$\begin{aligned} \overline{\phi}_P(X) \quad = \quad & [w^{(1)}\overline{\phi}_S(X^{(1,1)}); ...; w^{(l)}\overline{\phi}_S(X^{(l,t)}); \\ & ...; w^{(L)}\overline{\phi}_S(X^{(L,n_L)})] \end{aligned} \tag{8}$$

In Eq.8, $w^{(l)} = \frac{\frac{1}{n_l}}{\sum_{l=1}^{L}\frac{1}{n_l}}$ is the weight associated with level l. Fig. **??** shows image-level feature extraction on the pyramid structure. Until now, we obtain the final representation of the whole image, that we call image-level feature vector. This vector will be input of a Multiclass SVM for training and testing.

4. EXPERIMENTAL RESULTS

4.1 Dataset

In order to evaluate the proposed method, we extract a subset of ImageClef2013 dataset for plant identification task. ImageClef2013 is a benchmark dataset for plant identification evaluation. This dataset consists of images of 250 species of herbs and trees living in France with different views or organs of plants. This is challenging dataset for plant identification task due to the large intra-class variability and inter-class similarity. Concerning leaf images, there are two types of images: SheetAsBackground (containing scans and scan-like photographs of leaves in a front of a generally white uniform background) and NaturalBackground (cluttered natural background). In our experiments, we use both SheetAsBackground and NaturalBackground images. However, we select species images from ImageClef2013 using the following constraints: (1) the leaf of this species is

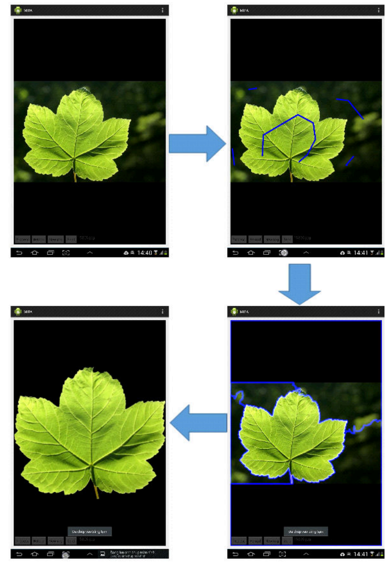

Figure 4: Interactive segmentation developed for mobile devices. Top left: original image, top right: markers, bottom right: boundary with Watershed, bottom left: segmented leaf

simple leaf since our work is not dedicated to compound leaf; (2) the number of training images is greater than a threshold (in our experiments, the chosen value is 20) in order to make a balanced dataset. As result, 80 species satisfy the above mentioned constraints. For this, some species have a high number of testing images (e.x. 106 for Laurus nobilis) and some species have very few testing images (e.x. 1 for Alnus cordata). As consequence, our working dataset contains 5540 and 1660 images of 80 species for training and testing respectively.

4.2 Results

We have implemented our method using client-server architecture. The client is mobile device, in our case, this is Android mobile device. In the client side, we implemented the interactive image segmentation. The user can take a picture of the plant or can use an existing picture in the album to do plant identification. In the server side, we reimplemented the kernel descriptor of Bo et al. [2] with our improvements. Fig. 4 shows the interface of the interactive image segmentation developed for Android mobile device.

In order to evaluate the proposed plant identification method, we have performed four experiments. The aim of the first experiment is to evaluation our method. For this, we firstly use the developed interactive segmentation in order to separate the leaf region from the background for all images in the training and testing dataset (5540 and 1660 respectively). Then, we apply our improved kernel descriptor for leaf representation. The purpose of the second experiment is to see the performance of the original kernel descriptor on the interactive segmented leaf databases. For this, we use the same database as the first experiment and apply the original kernel descriptor [7]. The third and the second experiments is to evaluate the improved kernel descriptor and original kernel descriptor on the automatic leaf image database. For this, we apply an automatic image segmentation method.

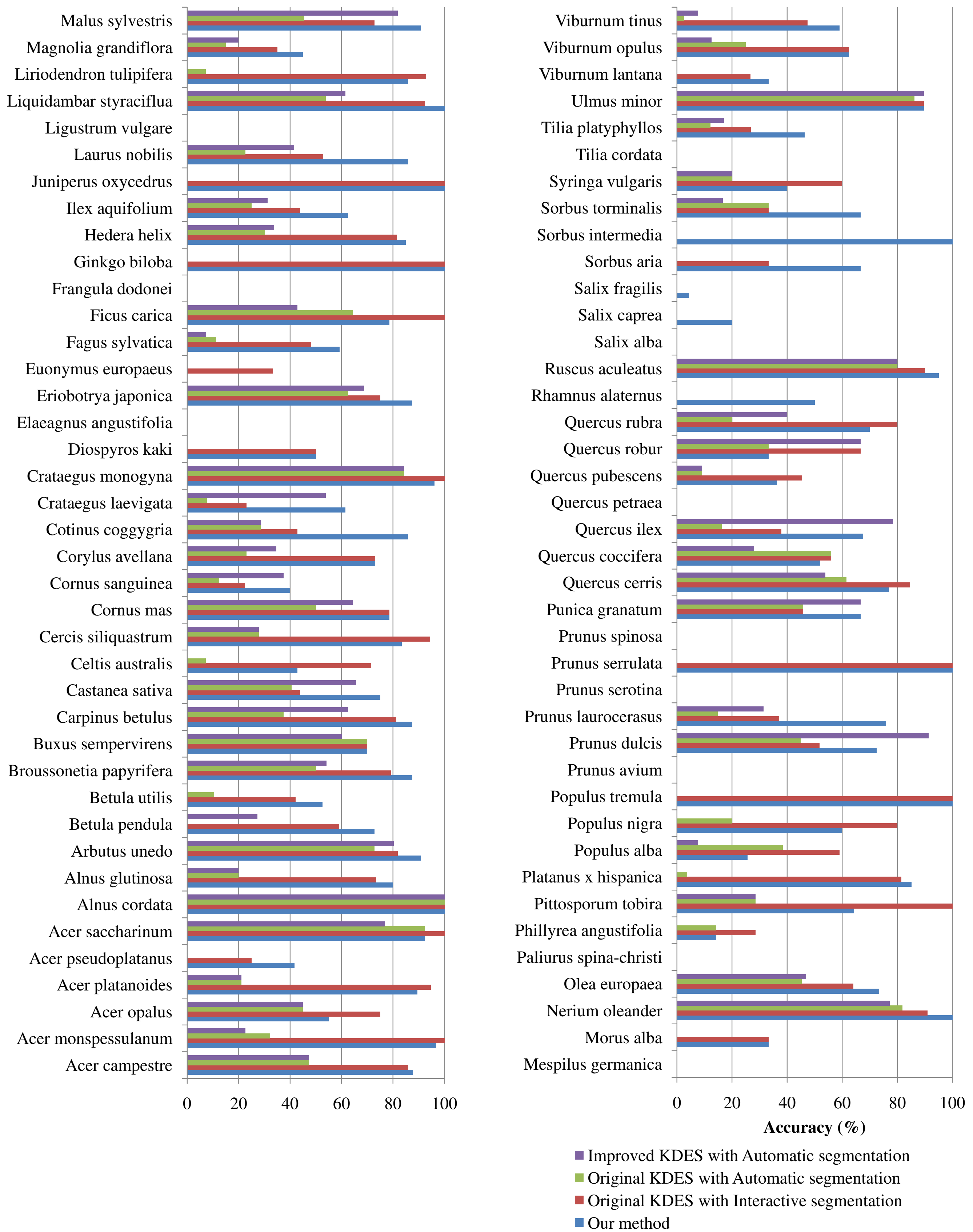

Figure 5: Detail accuracies obtained for 80 classes in the 4 experiments. For some classes such as Mespilus germanica, the obtained accuracy in the 4 experiments is 0%.

We have chosen the method based on salient region [1] in on the testing dataset. For the training dataset, we use the same as two first experiments. It is worth to note that in all of four experiments, we keep the same parameter of multi-class SVM. We compute the accuracy that is defined by the ratio of the number correct classified images and the number of testing images for each class. Then we compute the average accuracy for all of classes. The average accuracy obtained with the four above mentioned experiments are shown in Tab. 1. The results show that our improvements on kernel descriptor extraction make a significant increase of the performance on both interactive and automatic segmented images. Moreover, the proposed method based on a combination between improved kernel descriptor and interactive segmentation obtains the best result. In [7], the authors obtained only 63.4 % of accuracy for 70 classes of species with simple background images (the scan and scan-like background). From the result we can see that, even with the robust feature such as kernel descriptor, if we apply the automatic image segmentation, the result is far from the users' expectation. Moreover, in the point of view of the real application for users, it is acceptable if users have to define few number of markers (in our experiments, the number of marker is from 1 to 3). When using the mobile device, this becomes easy task even for novice users thanks to the convenience of the user's touch on a mobile device's screen. Fig. 5 shows the detail accuracy for each class. From the result, we can observe that in the simple cases (the class is highly distinctive), the original descriptor gives a acceptable result. In these cases, the improved kernel descriptor does not show its robustness. However, for the difficult cases, the improved kernel descriptor allows to improve the accuracy. For example, in [7], the authors have stated that the original descriptor does not work with *Salix fragilis* because of the poor quality of the images. In this case, our method allows to recognize this species even with a low accuracy. The obtained result have also shown that for certain species such as *Mespiluls germanica*, the proposed method can not work because of the high intra-class variability. In this case, we need to collect more images about these species in order to capture different aspects of their leaves.

Table 1: Accuracy obtained in four experiments.

Method	Accuracy
Our method	**71.5**
Original KDES with Interactive segmentation	63.4
Improved KDES with Automatic segmentation	42.3
Original KDES with Automatic segmentation	35.5

5. CONCLUSIONS

In this paper, we have proposed a new method for leaf-based plant identification combining the interactive segmentation method and kernel descriptor. Different experiments performed on the leaf dataset of ImageClef 2013 have proved the robustness of the proposed method. However, this method is still limited to the simple leaf images. Since plant identification belongs to the fine-grained classification problem. The methods proposed such as [14] for this problem can be considered for plant identification in our future works.

6. REFERENCES

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1597–1604, June 2009.

[2] Liefeng Bo, X Ren, and Dieter Fox. Kernel Descriptors for Visual Recognition. *NIPS*, pages 1–9, 2010.

[3] Liefeng Bo and Cristian Sminchisescu. Efficient Match Kernel between Sets of Features for Visual Recognition. *NIPS*, pages 1–9, 2009.

[4] Guillaume Cerutti, Laure Tougne, Céline Sacca, Thierry Joliveau, Pierre-Olivier Mazagol, Didier Coquin, Antoine Vacavant, et al. Late information fusion for multi-modality plant species identification. *CLEF Working Notes*, 2013.

[5] Hervé Goëau, Alexis Joly, Pierre Bonnet, Vera Bakic, Daniel Barthélémy, Nozha Boujemaa, and Jean-François Molino. The imageclef plant identification task 2013. In *Proceedings of the 2Nd ACM International Workshop on Multimedia Analysis for Ecological Data*, MAED '13, pages 23–28, New York, NY, USA, 2013. ACM.

[6] Julien Mille Antoine Vacavant Didier Coquin Guillaume Cerutti, Laure Tougne. Guiding active contours for tree leaf segmentation and identification. In *Proceedings of the Conference on Multilingual and Multimodal Information Access Evaluation*, 2011.

[7] Thi-Lan Le, Duc-Tuan Tran, and Ngoc-Hai Pham. Kernel descriptor based plant leaf identification. *4th International Conference on Image Processing Theory, Tools and Applications*, 2014.

[8] D.G. Lowe. Object recognition from local scale-invariant features. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 2:1150–1157, 1999.

[9] Subhransu Maji, Alexander C Berg, and Jitendra Malik. Efficient classification for additive kernel SVMs. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):66–77, January 2013.

[10] F. Meyer and S. Beucher. Morphological segmentation. *Journal on Visual Commun. Image Represent*, 1(1):21–46, 1990.

[11] Yahiaoui I. Boujemaa N. Zagrouba E. Mzoughi, O. Advanced tree species identification using multiple leaf parts image queries. In *2013 20th IEEE International Conference on Image Processing (ICIP)*, pages 3967 – 3971, 2013.

[12] Andrea Vedaldi and Andrew Zisserman. Efficient additive kernels via explicit feature maps. *IEEE transactions on pattern analysis and machine intelligence*, 34(3):480–92, March 2012.

[13] Xiao-Feng Wang, De-Shuang Huang, Ji-Xiang Du, Huan Xu, and Laurent Heutte. Classification of plant leaf images with complicated background. *Applied Mathematics and Computation*, 205(2):916 – 926, 2008.

[14] Ning Zhang, Jeff Donahue, Ross B. Girshick, and Trevor Darrell. Part-based r-cnns for fine-grained category detection. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*, pages 834–849, 2014.