



# Plant recognition based on Jaccard distance and BOW

Zhaobin Wang<sup>1</sup> · Jing Cui<sup>1</sup> · Ying Zhu<sup>2</sup>

Received: 21 December 2019 / Accepted: 8 May 2020 / Published online: 9 June 2020  
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

Plant recognition is a meaningful research that has attracted many researchers. Due to the variety of plants, it is difficult for the existing identification methods to identify their species efficiently. We propose a plant recognition method based on Jaccard distance and Bag of words (BOW). Firstly, Jaccard distance is employed to calculate the similarity between the test sample and part of the training samples of all species,  $C_1$  species with the highest similarity are selected as candidate species of the test image, which not only reduce the amount of computation but also shorten the time consumption. Secondly, BOW is employed to extract features from texture image and contour image, and support vector machine is used for training and classification. In our method, the texture and contour features of leaf images are extracted by Laws texture measure and Sobel operators respectively. The local and global features of the leaf can be described well. Some representative datasets are used to evaluate the proposed method and obtain high accuracy. Comparison with existing methods proves that the proposed method not only has a high accuracy, but also has robustness in noise environment.

**Keywords** Plant recognition · Feature extraction · Bag of words · Jaccard distance

## 1 Introduction

Plants are closely related to our life. The classification of plants plays an important role in the exploitation and protection of plant resources. With the rapid development of digital image processing and pattern recognition, more and more researchers pay attention to classification and identification of plant species. In the plant species recognition, the leaves, flowers [1], barks [2], fruits, stems and roots of plants can be used to classification. However, flowers and fruits season for only a few months of the year, it is difficult to

collect the images of flowers and fruits. Moreover, the sample images collected at different flowering periods are quite different. Leaves' images are easier to collect than images of flowers and fruits, and the shape and texture of leaves are more stable. So, most of the studies use leaf to plant recognition and classification. The characteristics of the leaf image can be represented by its shape, texture and color. And shape features include leaf margin, leaf tip, leafstalk and so on. Saleem et al. [3] proposed a novel plant identification method, which used the optimized shape features extracted from the leaf images. Munisami et al. [4] proposed a method by extracting shape and color features for plant recognition, such as the length, width, area and the perimeter of the leaf, and color histogram. Zhang et al. [5] proposed shape features which used a combination of morphological features to characterize the global shape of the leaf, and combined the global shape features with margin features. Some texture features presented by Turkoglu Muammer and Hanbay Davut [6], including region mean—LBP(RM-LBP), overall mean—LBP(OM-LBP) and ROM-LBP. They are the improved versions of the LBP method, and worked by considering the region and overall mean instead of the center pixel for coding. Savio et al. [7] proposed a new method for texture recognition based on CNs and pagerank. Discrete

---

Communicated by Y. Zhang.

---

This work was funded by National Natural Science Foundation of China (Grant No. 61201421).

---

✉ Zhaobin Wang  
zhaobin\_wang@hotmail.com

Ying Zhu  
zhuying\_365@126.com

<sup>1</sup> School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China

<sup>2</sup> Key Laboratory of Microbial Resources Exploitation and Application of Gansu Province, Institute of Biology, Gansu Academy of Sciences, Lanzhou, China

Schroedinger transform(DST) [8, 9] is used for texture recognition and leaf recognition.

Many studies that combine the shape and texture features of leaf for plant species recognition. Ali Jan Ghasab et al. [10] extracted shape, texture, morphology and color from leaf images to establish a feature search space, and employed the ant colony optimization (ACO) to obtain the best discriminant features. Liu et al. [11] combined the shape features include Hu moment invariants and Fourier descriptors, and the texture features include local binary patterns, Gabor filters and gray-level co-occurrence matrices for plant recognition. Chaki et al. [12] proposed a novel methodology, which uses Gabor filter and gray level co-occurrence matrix (GLCM) to model texture features and uses some curvelet transform coefficients with invariant moments to model shape features. VijayaLakshmi et al. [13] proposed an approach of leaf recognition by combining Haralick texture-based features, Gabor features, shape features, and color features. Zhang et al. [14] combined shape features and texture features, and principal component analysis and linear discriminant analysis are combined to reduce the dimension.

In the past years, many researchers used Bag of words (BOW) model for plant species recognition. Larese et al. [15] detected Scale-Invariant Feature Transform (SIFT) keypoints in segmented vein images, and used SIFT descriptors to build BOW model. Pires et al. [16] proposed a method which is based on Bag of Visual Words and several images local descriptors, including SIFT, dense scale-invariant feature transform (DSIFT), pyramid histograms of visual words (PHOW), speeded-up robust features (SURF), and histogram of oriented gradients (HOG). Wang et al. [17] proposed a new method of leaf recognition based on bag of words (BOW) and entropy sequence (EnS) obtained from dual-output pulse-coupled neural network (DPCNN), the improved BOW enhance the ability to represent EnS's features.

It can be seen from the above papers that the key to plant species identification and classification is whether the features extracted from leaf are stable and whether they have good recognition ability. In general, it is difficult to achieve high recognition accuracy by only texture features or shape features. Therefore, to improve the representation ability of features, We proposes a new two-stage method of plant recognition based on Jaccard distance, Laws texture feature and contour feature of the image and bag of words (BOW).

The proposed method of plant species recognition has the following advantages: (1) Jaccard distance can exclude some classes that are more dissimilar to the test images, meanwhile, it can reduce the time consumption of recognition. (2) The combination of Laws texture feature and contour feature with BOW has higher accuracy of recognition than traditional BOW. (3) This method is robust to noise and it is easy to apply to image classification.

The rest of the paper is organized as follows: Sect. 2 briefly introduces some related basic theories, including the Jaccard distance, bag of words and Laws' texture measures. Section 3 introduces the details of we proposed two-stage recognition method. Section 4 presents experimental results on several representative leaf image datasets.

## 2 Related theory

The steps of feature extraction and classification are significant in plant identification. In this section, we will introduce some theories related to the proposed method. Jaccard distance, Laws' texture energy measure and bag of words are used to coarse classification, texture feature extraction, and classification respectively.

### 2.1 Jaccard distance

In image processing, different distance metrics are used to calculate the similarity between images, such as Euclidean distance, Jaccard distance [18, 19], Gaussian kernel distance, Mahalanobis distance [20] and so on. Jaccard distance is used to measure the difference between two sets, and Jaccard similarity coefficient is used to measure the similarity between two sets. Jaccard distance is defined as 1 subtract Jaccard similarity coefficient. Therefore, the Jaccard distance between binary images can be calculated quickly. Suppose there are two binary images, set  $A$  and set  $B$ , the Jaccard similarity coefficients  $J$  and distances  $D_j$  are defined as follows:

$$J(A, B) = \frac{M_{11}}{M_{01} + M_{10} + M_{11}}, \quad (1)$$

$$D_j(A, B) = 1 - J(A, B) = \frac{M_{01} + M_{10}}{M_{01} + M_{10} + M_{11}}, \quad (2)$$

where  $M_{11}$  is the total number of dimensions with values of 1 in both  $A$  and  $B$ ,  $M_{01}$  is the total number of dimensions with value of 0 with  $A$  and value of 1 with  $B$ ,  $M_{10}$  is the total number of dimensions with value of 1 with  $A$  and value of 0 with  $B$ . In the calculation of the Jaccard distance and coefficient, removed pixels with a value of 0 in both images, that is  $M_{00}$ . It is suitable for evaluating the similarity between leaf images.

### 2.2 Laws' texture energy measures

Texture analysis is an important task in image processing, and Laws is a significant operator in texture analysis. The essential principle of Laws texture energy measure is to apply the small convolution kernel to the digital image

firstly, and then perform a nonlinear window operation to extract the high-frequency part or the low-frequency part of the image.

The proposed method uses a  $5 \times 5$  micro-window to measure the grayscale irregularity of small areas centered on pixels. The two-dimensional convolution mask is obtained by convolving a set of one-dimensional convolution kernels of length 5. The one-dimensional convolution kernel is composed of four basic texture vectors: level (*L*), edge (*E*), spot (*S*), and ripple (*R*). The one-dimensional convolution kernel is as follows:

$$L5(\text{level}) = [1 \quad 4 \quad 6 \quad 4 \quad 1], \tag{3}$$

$$E5(\text{Edge}) = [-1 \quad -2 \quad 0 \quad 2 \quad 1], \tag{4}$$

$$S5(\text{Spot}) = [-1 \quad 0 \quad 2 \quad 0 \quad -1], \tag{5}$$

$$R5(\text{Ripple}) = [1 \quad -4 \quad 6 \quad -4 \quad 1]. \tag{6}$$

We can obtain 16 different two-dimensional convolution kernels by convolving a horizontal one-dimensional kernel with a vertical one-dimensional kernel. The two-dimensional kernels are shown in Table 1.

Laws' texture energy measures have the following steps [21]:

Step 1: Apply convolution kernels. Firstly, apply each of the 16 convolution kernels to the image of  $M$  rows and  $N$  columns which we want to texture analyze, and we can get a set of  $16 \times M \times N$  grayscale images.

Step 2: Performing windowing operation. The Texture Energy Measure (TEM) at the pixel replaces each pixel in the  $16 \times M \times N$  individual grayscale images. Adding the absolute values of the local neighborhood pixels around each pixel produces a new set of images, which called TEM images.

Step 3: Normalize features for contrast. All convolution kernels we used are zero-mean except for the L5L5 kernel. Hence, the L5L5 image can be regarded as a normalized image, and the TEM image is normalized pixel by pixel with the L5L5T image (TEM image by L5L5 convolution kernel), that is, the feature is normalized for contrast.

Step 4: Combine similar features. The directionality of textures is not very significant in many applications. Hence,

the deviation in the dimensional characteristics is eliminated by combining similar features. For instance, L5E5T and E5L5T are sensitive to vertical and horizontal edges, respectively. We can get a single feature that is sensitive to simple "edge content" by adding these TEM images together. The nine final energy maps are L5E5/E5L5, L5R5/R5L5, E5S5/S5E5, S5S5, S5R5/R5S5, R5R5, L5S5/S5L5, E5E5, and E5R5/R5E5.

### 2.3 Bag of words

The bag of words(BOW) model is a commonly used representation in the field of information retrieval [22]. When applying the BOW model to image processing, the image can be represented in the form of a document, and it is a collection of several "visual vocabulary". Therefore, we need to extract the independent visual vocabulary from the image firstly, which usually requires the following three steps: (1) feature detection; (2) feature representation; and (3) dictionary generation.

Although there are differences between different samples of the same type of the target, we can still find the common characteristics among samples. Extract common features among these different samples as the visual vocabulary for identifying these target species. SIFT algorithm is widely used to extract local invariant features from images. SIFT features are invariant to rotation, scale scaling and brightness variations, and it also has a certain degree of stability to the change of viewing angle and noise. Hence, we use these invariant features as visual vocabulary and construct a dictionary.

BOW model has the following three steps shown in Fig. 1. (1) the SIFT algorithm is applied to extract the vectors of visual words from different kinds of images, which represent local invariant feature points in these images; (2) the  $k$ -means algorithm is used to combine similar visual words and construct a dictionary containing  $K$  words; and (3) count the number of times each word in the dictionary appears in the image, and represent the image as a  $k$ -dimensional feature vector.

## 3 Proposed method

In general, leaf recognition can be divided into three steps: image preprocessing, feature extraction, and classification. The proposed method of recognition also adopts the above steps. The specific details of the method are shown in Fig. 2.

### 3.1 Image preprocessing

The original leaf images in most databases are randomly angularly oriented. So, we need to rotate the image to put the

**Table 1** Two-dimensional kernel name

	L5	E5	S5	R5
L5	L5L5	E5L5	S5L5	R5L5
E5	L5E5	E5E5	S5E5	R5E5
S5	L5S5	E5S5	S5S5	R5S5
R5	L5R5	E5R5	S5R5	R5R5

Fig. 1 Processing flow of BOW

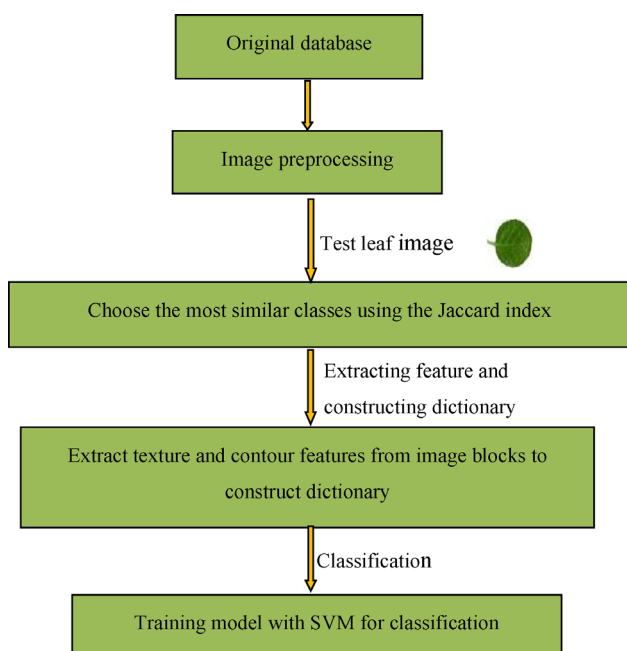
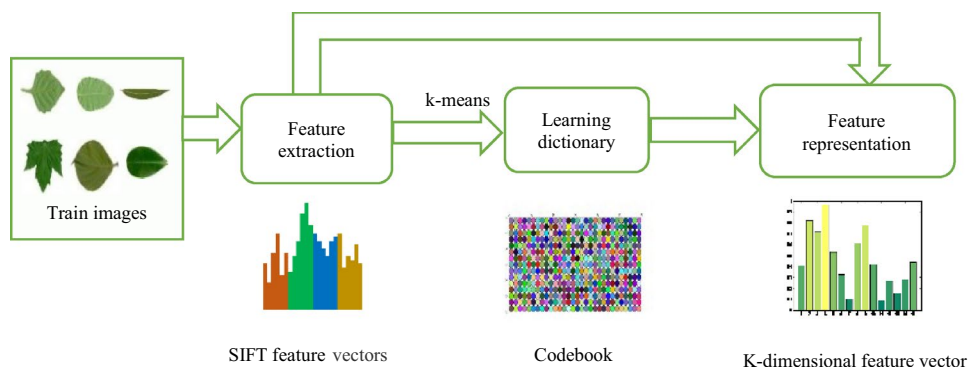


Fig. 2 Framework of the proposed method

leaf in the center of the image, with the petiole at the bottom and the tip at the top. This work makes it easier to use the Jaccard index for similarity calculations. Besides, the image is denoised by a median filter.

### 3.2 Feature extraction

Before extracting image features, Jaccard distance is firstly employed to calculate the similarity between the test image and images from the dataset. For example, 30 images for 5 species are selected in the Flavia dataset and select an image as the test sample in the first species. Firstly, the input color image is converted to grayscale image, as shown in Fig. 3b. The image size is required to be the same, so the size of image is adjusted, as shown in Fig. 3c. The edge is detected by Sobel operator with threshold 0.1 and the image contour is extracted, as shown in Fig. 3d. Then, calculate the average

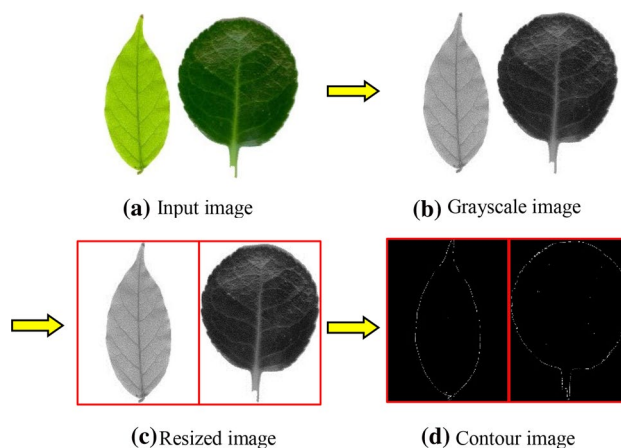


Fig. 3 Extraction process of contour image

Table 2 Average jaccard coefficients and average jaccard distances

Species no.	Average Jaccard coefficient	Average Jaccard distance
No. 1	0.0336	0.9664
No. 2	0.0049	0.9951
No. 3	0.0068	0.9932
No. 4	0.0027	0.9973
No. 5	0.0024	0.9976

Jaccard coefficient and distance of the test image and the 30 images of each of the 5 classes in the Flavia database by Eqs. (1) and (2), respectively. And obtain the classes that are more similar to the test image, as shown in Table 2.

The larger the average Jaccard coefficient is, the more similar it is to the test image. From Table 2, we can find that species No. 1 has the largest Jaccard coefficient, so the test image is the most similar to species No. 1. Although the Jaccard coefficient calculated by Sobel operator in this dataset is very small, the accuracy of calculating similarity is better

than Canny and other operators. In this method, we choose the threshold with 0.1, and different thresholds correspond to different contour images. Besides, different thresholds also affect the recognition rate, which will be explained in detail in the following sections.

According to the calculation of the average Jaccard coefficient, we can exclude some species that are not similar to

the test image, and eliminate the negative influence of these species in the identification. The average Jaccard coefficient was ranked from high to low. Top  $C_1$  species with higher average Jaccard coefficient are selected as candidate training classes from  $C$  species leaves, and  $C - C_1$  species are discarded. The pseudo code of the Jaccard index calculation is shown in Algorithm 1.

---

**Algorithm 1** Calculate image similarity using Jaccard distance

---

**Input:** The test image,  $I$ ; The train sample,  $T_{ij}$ ; The number of total species in the dataset,  $N$ ;

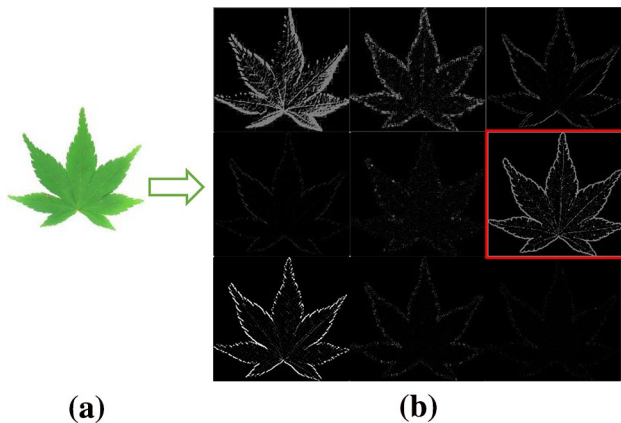
**Output:** Jaccard index matrix,  $W$

```

1:  $S_I \leftarrow \text{Sobel}(I)$ 
2: for  $i \leftarrow 1$  to  $N$  do
3:    $J_i \leftarrow 0$ 
4:   for  $j \leftarrow 1$  to 30 do
5:      $S_{ij} \leftarrow \text{Sobel}(T_{ij})$ 
6:     Apply Eq.(1) to calculate Jaccard index between test image( $S_I$ ) and train
       image( $S_{ij}$ )
7:      $J_i \leftarrow J_i + J_{ij}$ 
8:   end for
9:    $J_i \leftarrow J_i / 30$ 
10: end for
11: if  $N > 100$  then
12:    $C_1 \leftarrow 1/3N$ 
13: else
14:    $C_1 \leftarrow 1/2N$ 
15: end if
16:  $W \leftarrow$  The first  $C_1$  largest values in  $J_i$  and its corresponding species  $i$ 

```

---



**Fig. 4** Extraction process of texture image. **a** Input image; **b** the Laws' energy texture image

The nine energy maps extracted by Laws' texture are shown in Fig. 4, and the texture image that we used is in the red box. This image is the combination of S5L5 and L5S5. L5S5 measures vertical speckle content and S5L5 measures horizontal speckle content. Therefore, the total spot content will be the mean of S5L5 and L5S5.

The proposed method uses both contour and texture features. The pseudo code of feature extraction and classification is shown in Algorithm 2. Firstly, the Laws' texture and the Sobel operator extract the texture and contour images respectively. Then divide the two pictures into blocks separately, and extract SIFT features from these blocks to form feature vectors. Let  $T_{ij}(i = 1, 2, \dots, C_1, j = 1, 2, \dots, n)$  and  $S_{ij}(i = 1, 2, \dots, C_1, j = 1, 2, \dots, n)$  are the texture and shape feature vector of the  $j$ th image of the  $i$ th species respectively, where  $C_1$  is the number of candidate training species, and  $n$  is the number of training images per species. Let  $T_{ijt}$  and  $S_{ijt}$  be the texture and shape feature of  $t$ th regions respectively,

an image can be described as  $T_{ij} = [T_{ij1}, T_{ij2}, \dots, T_{ijM}]$  and  $S_{ij} = [S_{ij1}, S_{ij2}, \dots, S_{ijM}]$ , where  $M$  is the number of blocks in the image. The size of the image and the size of the block determine the value of  $M$ . The sizes of different images are not necessarily the same, so the values of  $M$  may not be equal. The feature vectors of the training set are defined as follows:

$$W_{ij} = [T_{ij}, S_{ij}]. \quad (7)$$

composed of cluster centers,  $B = [b_1, b_2, \dots, b_D] \in R^{(D \times n)}$ . The number of clustering centers ( $D$ ) is equal to the number of virtual codes. In the proposed method, the number of codes is fixed to improve the speed and performance of dictionary learning.

In addition, pyramid matching is added to the traditional BOW model and spatial information is added to the feature representation. The image is divided into blocks with fixed-size such as  $1 \times 1$ ,  $2 \times 2$ ,  $4 \times 4$ ,  $16 \times 16$ , then the number

---

### Algorithm 2 Feature extraction and classification

---

**Input:** Sample images of candidate species,  $M_i$ ; Test image,  $I$ ;

**Output:** Classification results

- 1: Contour image  $\leftarrow$  Sobel( $M_i$ )
  - 2: Texture image  $\leftarrow$  Laws texture measure( $M_i$ )
  - 3:  $S_i \leftarrow$  SIFT of BOW (Contour image)
  - 4:  $T_i \leftarrow$  SIFT of BOW (Texture image)
  - 5:  $N_i \leftarrow [S_i, T_i]$
  - 6: Construct dictionary with BOW
  - 7:  $I \leftarrow J_{max} * [S_I, T_I]$
  - 8: Classification with SVM
- 

Next, the proposed method weights the feature vector. The feature vectors of each species are multiplied by the corresponding average Jaccard coefficient after extracting the features of training images:

$$W'_{ij} = J_i W_{ij}, \quad (8)$$

where  $J_i$  is the average Jaccard coefficient of the  $i$ th species,  $i = 1, 2, \dots, C_1$ . In most cases, the species of the test image has the highest Jaccard coefficient, while in some cases the average Jaccard coefficient of the test image species is in the top three. And the differences in the Jaccard coefficients of several species which are similar to the test images are small. Hence, using the Jaccard coefficient not only reduces the complexity but also improves the recognition accuracy. When input the test image for classification, the weighted coefficient of the feature vector is the maximum Jaccard coefficient of the test image.

### 3.3 Dictionary construction

The traditional  $k$ -means dictionary learning method widely used in the field of sparse coding [23] is employed to construct a visual code dictionary. After obtaining the feature histogram of image dataset,  $k$ -means algorithm is used for cluster analysis. The virtual code of the dictionary is

of different codes is counted in each block. From left to right, count the histograms in each block at different levels. Finally, the histograms obtained from each level are concatenated, each level is given a corresponding weight, and the weights from left to right are sequentially increased.

### 3.4 Classification

The final step in plant identification is classification. Many classifiers are applied in plant identification, such K-Nearest Neighbor (KNN) [24], Support Vector Machine (SVM) [25], Random forests [26], Probabilistic Neural Network (PNN) [27] and so on [28]. SVM is a classic algorithm of machine

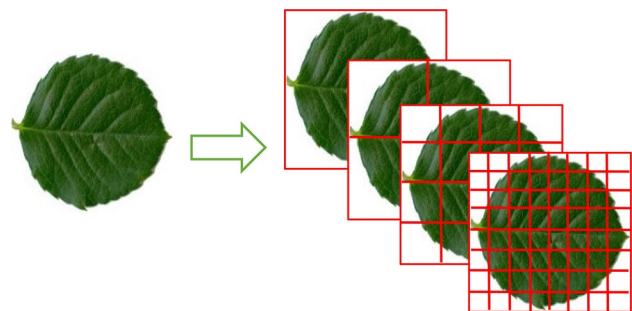


Fig. 5 Spatial pyramid for feature-pooling



Fig. 6 Typical leaf examples from the Flavia dataset



Fig. 7 Typical leaf examples from the LZU dataset

learning, which has achieved good results in many fields [29]. It has fast processing speed and the ability to process large-scale data, which makes it widely used in engineering practice. The image samples of the dataset are divided into two subsets of the training set and the testing set. To ensure the accuracy of the proposed method, we use the test method of tenfold cross validation and five-fold cross-validation.

### 3.5 Analysis of time complexity

The elapsed time of our proposed method mainly consists of three parts. The first part is the time of calculating the similarity between test image and train images. It takes time  $O(n)$  to calculate the similarity coefficient between the training samples of each species ( $i = 1, 2, \dots, n$ ) and the test images. Therefore, the total time of computing the similarity of all species ( $N$ ) is  $O(Nn)$ . The second part is the time of extracting features from the training samples of the candidate species. Divide the image into  $L$  patches according to different patch sizes and step size, and extract key points on each patch using SIFT algorithm. The time of extracting features is  $O(L^2)$  for all patches. The third part is clustering with K-means, and the time is  $O(tkm)$ , where  $t$  is the number of iterations for clustering,  $k$  is the number of clustering centers, and  $m$  is the number of samples. In the experiment, because the number of patches, clustering centers and iterations are small, the calculation of these parts are fast.

## 4 Experiments and analysis

In this section, parameter setting in the experiments is explained firstly. Then, the proposed method is tested on five leaf datasets. To verify the effectiveness of the proposed method, our proposed method is compared with some state-of-the-art leaf classification methods.

### 4.1 Parameter setting and test dataset

The setting of the parameters greatly affects the performance of the recognition. In the proposed method, we choose the patch size of sample images is 48 and the pyramid levels for pooling is 4 to extract detailed low-level features. The leaf image is divided into  $1 \times 1$ ,  $2 \times 2$ ,  $4 \times 4$  and  $16 \times 16$ , in total 277 blocks as shown in Fig. 5.

Support vector machine (SVM) has many different situations and the choice of kernel function plays a key role in the performance of SVM. In the proposed method, we choose the radial basis function as the kernel function, also known as the Gaussian kernel function. The kernel function is defined as follows:

$$k(x, y) = \exp(-\gamma \|x - y\|^2). \quad (9)$$

The radial basis function is a real-valued function whose value depends only on the distance of a specific point, as Eq. 10.

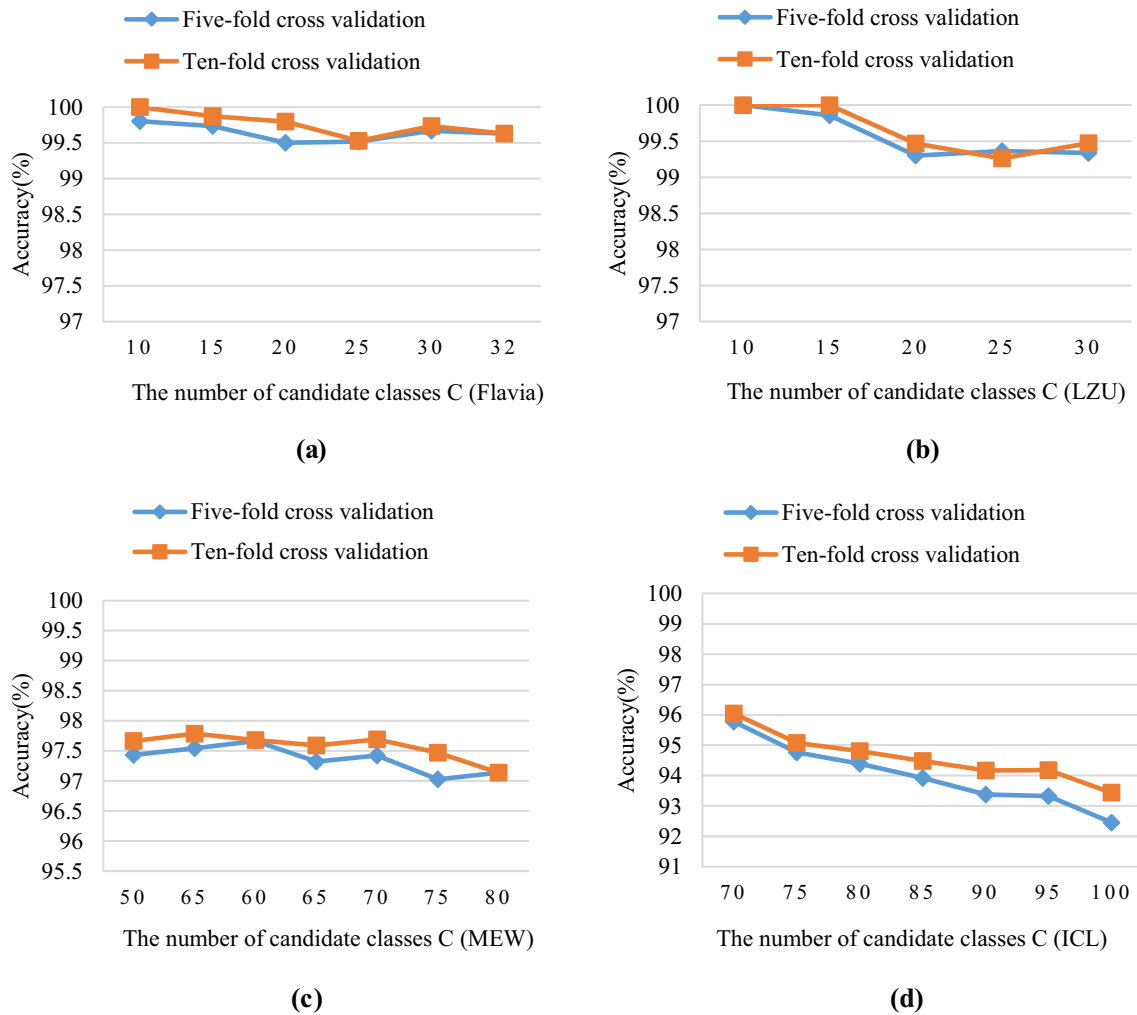
$$\Phi(x, y) = \Phi(\|x - y\|). \quad (10)$$

Currently, there are many common leaf datasets used to evaluate the performance of the recognition method. In this paper, we select five leaf datasets to evaluate the proposed identification method, they are Flavia dataset, Swedish dataset, LZU dataset, ICL dataset and MEW dataset.

Flavia dataset [30] is a very common leaf dataset that contains 1907 samples from 32 species and 50 to 73 samples per species, and most of them are common plants in the Yangtze Delta, China. Many researchers test the performance of plant recognition using Flavia dataset [10]. We randomly selected 30 samples of each species as the training set and 20 samples as the testing set. Some examples are shown in Fig. 6.

Swedish dataset [31] is also a common dataset used to test. The dataset contains 15 species, each species consisting of 75 sample images, for a total of 1125 leaf images. We randomly selected 50 samples of each species as the training set and 25 samples as the testing set.

LZU dataset is a leaf dataset collected by Lanzhou University. This dataset contains 30 kinds of plants at Lanzhou University of Lanzhou, Gansu Province, China. The number of leaf images varied for each species, for a total of 4221 leaf images for 30 species in this dataset. We randomly selected



**Fig. 8** Relationship between recognition accuracy and the different number of candidate classes C. **a** Experiments on Flavia dataset; **b** experiments on LZU dataset; **c** experiments on MEW dataset; **d** experiments on ICL dataset

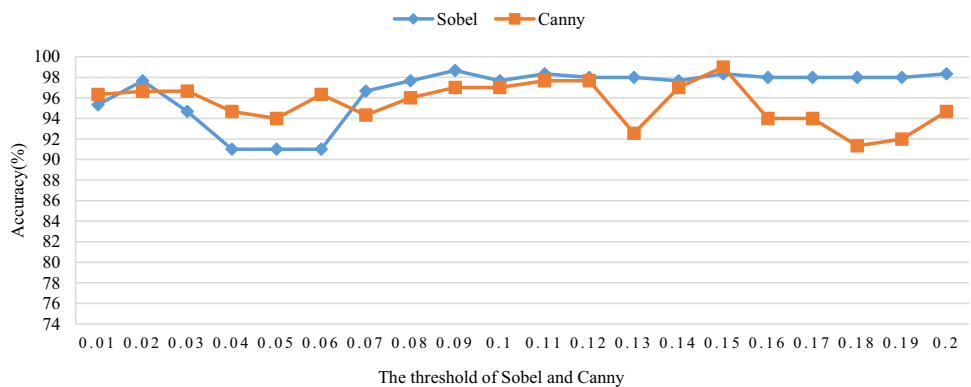
30 samples of each species as the training set and 20 samples as the testing set. Some examples are shown in Fig. 7.

ICL dataset is collected by the Intelligent Computing Laboratory (ICL) of the Institute of Intelligent Machines, Chinese Academy of Sciences. This dataset contains 6000

leaf images from 200 plant species with 30 leaf images per species. We randomly selected 20 samples of each species as the training set and 10 samples as the testing set.

MEW (Middle European Woods) dataset [32] is a large dataset which contains 153 kinds of Middle European woody

**Fig. 9** Relationship between the threshold of Sobel and Canny and accuracy in Flavia dataset





plants and a total of 9745 samples. In these experiments, we only selected 50 sample images for each species, 30 samples for training and 20 samples for testing.

Flavia dataset, Swedish dataset, LZU dataset, ICL dataset and MEW dataset are used to evaluate the proposed method. We mainly use ICL dataset and MEW dataset for parameter setting.

### 4.2 Effect of number of Candidate Classes C

From [19] and [33], the number of candidate classes  $C$  can be half or one-third of the number of species in the dataset. In fact, the size of number  $C$  affects the complexity of the model and the training time. Hence, it is very important to determine the number of candidate species. We use Flavia dataset, LZU dataset, MEW dataset and ICL dataset to test the impact of the number of candidate classes on identification accuracy respectively. Set the number of candidate classes  $C$  as  $\lceil T/3 \rceil$ ,  $\lfloor T/2 \rfloor$  in MEW dataset and ICL dataset. Set  $\lceil T/3 \rceil$ ,  $T$  in Flavia dataset and LZU dataset, where  $T$  is the total number of species in the dataset. We discuss the impact of the number of  $C$  by the proposed method in two situations: (1) five-fold cross validation, i.e., divide all the data into 5 parts, take one part for testing and the rest of 4 parts for training each time. A total of five tests and the results are averaged; (2) ten-fold cross validation, i.e., divide all the data into 10 parts, take one part for testing and the rest of 9 parts for training each time. A total of ten tests and the results are averaged. The results of four datasets are shown in Fig. 8.

As shown in Fig. 8a–c, the number of candidate classes  $C$  in the Flavia dataset, LZU dataset and MEW dataset has little effect on the recognition accuracy. However, in the ICL dataset, the recognition accuracy decreases as the number of candidate classes increases. When there are 70 kinds of candidate classes, the accuracy can reach 96%, but when the number of candidate classes is 100, the accuracy is 93.5%. At the same time, considering that the complexity will decrease with the decrease in the number of candidate

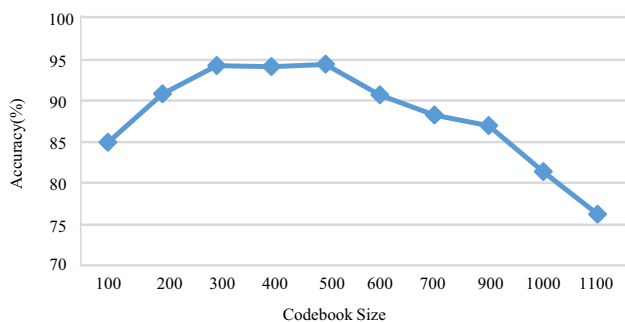


Fig. 10 Relationship of codebook size  $D_s$  and accuracy in ICL dataset

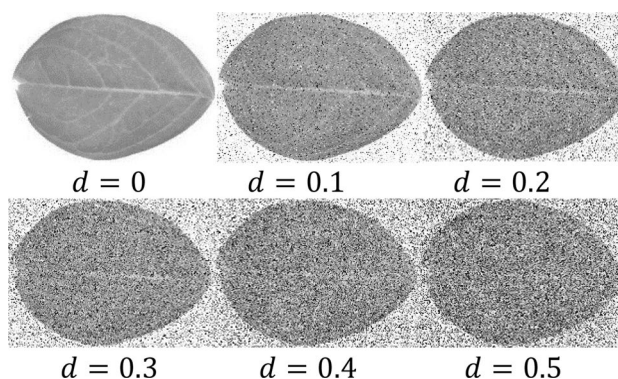


Fig. 11 Images with salt and pepper noise and  $d$  is changed from 0 to 0.5

classes, we select approximately one third of the number of total species as the number of candidate species.

### 4.3 Effect of threshold

Sobel and Canny operators are very important operators in pixel image edge detection. The edge images extracted by Sobel and Canny operators with different thresholds are different, which affects the similarity results of the calculation by Jaccard coefficients. We observe the influence of the change between the Sobel and Canny operator and the difference of the threshold on the recognition accuracy by changing the thresholds of the Sobel and Canny operators respectively. We set the threshold as  $0.01p$ ,  $p \in [1, 20]$ . And we choose the Flavia dataset for testing as shown in Fig. 9.

As shown in Fig. 9, with the increase of Sobel operator threshold, the recognition accuracy is unstable before the threshold value is 0.07, while after that the recognition accuracy is stable at around 98%.

However, with the change of the threshold of the Canny operator, the recognition accuracy is very unstable. The recognition accuracy is up to 99% and the minimum is 91%.

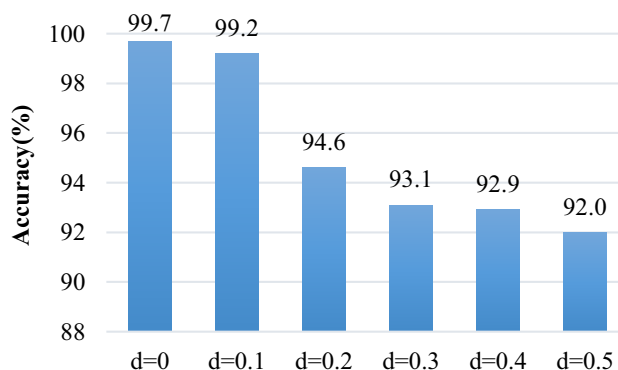


Fig. 12 Average accuracy with different noise density

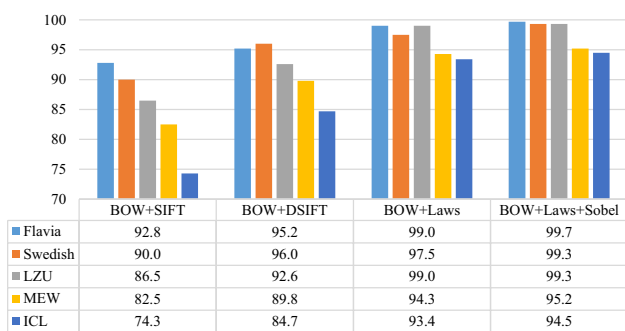


Fig. 13 Comparison of different method in five leaf datasets

In comparison, the Sobel threshold has better stability and higher recognition accuracy. In this proposed method, we choose the Sobel operator with the threshold of 0.1 to extract the edge information from the sample and calculate similarity between sample images.

#### 4.4 Effect of codebook size

The calculation of codebooks costs a lot of time. In order to reduce the computation and complexity, we need to build a smaller codebook with higher recognition accuracy for identification and classification efficiently. To study the effect of dictionary size  $D_s$  on recognition accuracy and to determine  $D_s$  in this proposed method, we set the dictionary size as  $100n, n \in [1, 10]$ . In Fig. 10, we can observe that the accuracy increases gradually while  $D_s$  grows from 100 to 300 and it tends to be stable when  $D_s$  is 400 and 500. After that, recognition accuracy gradually decreases with  $D_s$  increases. Since the recognition results are close when the dictionary size  $D_s$  is 300, 400 and 500, and it is inefficient to learn a large codebook. Therefore, we set the  $D_s$  to 350 in all datasets.

#### 4.5 Robustness to noise

To reflect the performance of the proposed method, we add salt and pepper noise to the image of the Flavia dataset

to observe the change of recognition accuracy. As Fig. 11 shows, salt and pepper noise of different noise density is added to the image.  $d$  is the noise density, and the value of  $d$  is from 0 to 0.5, that is, the percentage of noise value in the image area is from 0 to 50%. As the noise density increases, the picture clarity decreases. The average accuracy of 10 randomized trials is shown in Fig. 12.

When  $d = 0.1$ , the accuracy dropped from 99.7 to 99.2%, and when  $d$  further increased to 0.2, the accuracy dropped sharply to 94.6%. When  $d$  increased from 0.3 to 0.5, the accuracy drops gently from 93.1 to 92.0%. The results show that the proposed method has good performance even if the noise density is large, indicating that the method has better anti-noise ability.

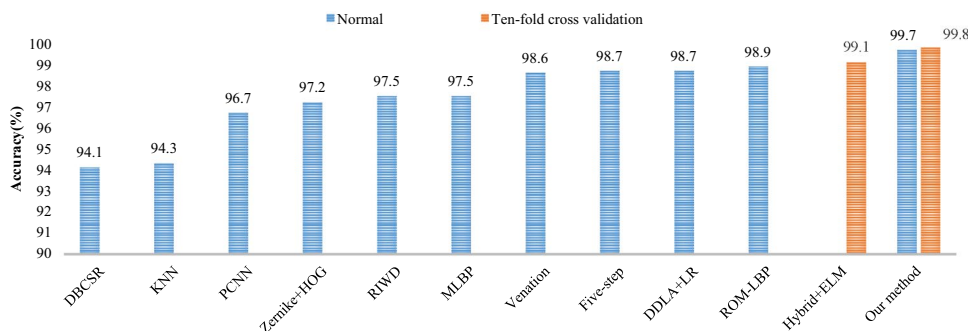
#### 4.6 Comparison with other methods

In this section, we compare some existing methods with the proposed method. BOW+SIFT is the method based on BOW in Ref. [34] and BOW+DSIFT is the improved method based on BOW in Ref. [16]. BOW+Laws is the method that removes the contour feature extracted by Sobel operator from our proposed method. BOW+Laws+Sobel is the method we proposed. We compare these four methods with five datasets as shown in Fig. 13. It is obvious that the method we proposed has the highest accuracy in five datasets. BOW+DSIFT has significantly improved compared with BOW+SIFT, especially in the ICL dataset. The accuracy of BOW+Laws also improved significantly compared with the accuracy of BOW+DISFT. And the accuracy of BOW+Laws+Sobel is slightly better than BOW+Laws which only extract texture features.

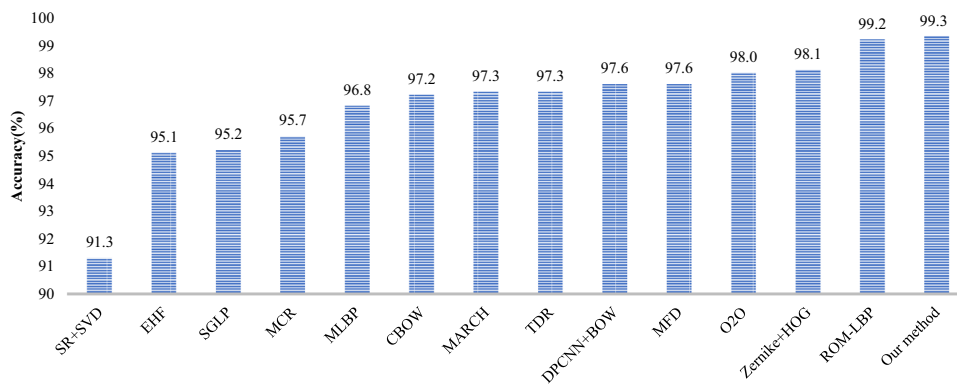
##### 4.6.1 Test on Flavia dataset

The results of the comparison with the existing different methods on the Flavia dataset are shown in Fig. 14. The accuracy of all comparison methods is above 94%, and the accuracy of our method is 99.7%. In Ref. [35], ten-fold cross validation is used to test the performance of Hybrid features, and the result of our method is 99.8% which is

Fig. 14 Comparison of proposed method with existing approaches using Flavia dataset



**Fig. 15** Compariinson of proposed method with existing methods in Swedish dataset



**Table 3** Comparison of proposed method with existing methods in MEW dataset

Method	Number of species	Training samples	Testing samples	Total samples	Accuracy (%)
MEW [32]	153	1/2	1/2	–	84.9
	153	All	1	–	88.9
PCNN [38]	153	–	–	–	91.2
Proposed method	153	30	20	7650	95.2

higher than 99.1% of Hybrid features. Demisse et al. [36] presented a deformation based representation approach for curved shapes(DBCSR) obtained the lowest accuracy. In Ref. [37], the shape and edge features are extracted from leaf images and K-NN classifier is used for classification. Wang et al. [38] used PCNN to extract the leaf features and combined with SVM. The method of Ref. [39] used Zernike Moments and Histogram of Oriented Gradients(HOG) to extract the shape features and texture features respectively. RIWD (Rotation Invariant Wavelet Descriptors) [40]and MLBP (Modified Local binary patterns) [41] have the same accuracy in Flavia. Ref. [42] proposed a new venation detection method. DDLA+LR [43] is the method using a dual deep learning architecture with logistic regression classifier, Ref. [3] presented a new five-step algorithm, and their accuracy are same. ROM-LBP [6] is the method based on LBP. The results of comparison show that our method is superior to other methods in the Flavia dataset.

**4.6.2 Test on Swedish Dataset**

We choose thirteen existing methods to compare with our methods in Swedish dataset. Zhang et al. [44] combined SR (sparse representation) and SVD (singular value decomposition) for plant recognition. In Ref. [45], Guo-dong et al. presented an algorithm of extract height functions for feature description. Supervised global-locality preserving projection (SGLP) is a new manifold learning method for plant leaf recognition proposed by Shao [46]. The MCR method proposed by Yu et al. [47], they extracted the leaf contour and venation features on multiple scales. In Ref. [48], Zeng et al.

proposed a shape recognition algorithm based on CBOW, which combined curvature and BOW. MARCH (termed multiscale arch height) is a multiscale shape descriptor proposed by Wang et al. [49]. Yang et al. [50] presented a new shape description approach called triangle-distance representation (TDR) for plant leaf recognition. Wang et al. [17] combined DPCNN(Dual-output Pulse-coupled Neural Network) and BOW. Yang et al. [51] proposed a novel multiscale Fourier descriptor based on triangular features (MFD) which is used to identify shapes. In Ref. [52], a novel post-processing method, online to offline(O2O), to improve the efficiency of shape retrieval is proposed. Our proposed method achieved the highest accuracy 99.3%, and the method of SR+SVD achieved the lowest accuracy. The accuracy of MLBP, CBOW, MARCH, TDR, DPCNN+BOW and MFD are similar. ROM-LBP has good performance compared with other methods, but lower than our method. In Fig. 15, the comparison results show that the proposed method is superior to these existing methods.

**4.6.3 Test on MEW dataset**

The species number in the MEW dataset is large and each species contains a large number of images. The comparison results with other methods are shown in Table 3. Table 3 lists the number of species used for testing, the number of training and testing samples for each species, the total number of samples, and the recognition results. Combining contour features and Fourier descriptors proposed by Novotny and Suk [32] has lower accuracy. In this method, the sample images of each species are divided into two equal parts as training

**Table 4** Comparison of proposed method with existing methods using ICL dataset

Method	Number of species	Training samples	Testing samples	Total samples	Accuracy (%)
RM-LBP [6]	31	–	–	350	71.4
OM-LBP [6]	31	–	–	350	76.6
ROM-LBP [6]	31	–	–	350	83.7
PCA+LDA [14]	42	45	–	–	87.8
SGLP [46]	100	–	–	5000	95.3 (Five-fold CV)
JSDR [19]	200	24	6	6000	93.4 (Five-fold CV)
LWSRC [33]	200	15	15	6000	81.4 (Half-fold CV)
TMMG [5]	200	15	15	6000	83.4
CS [53]	200	15	15	6000	89.4 (Half-fold CV)
	200	20	10	6000	94.5
Proposed method	200	15	15	6000	91.8 (Half-fold CV)
	200	24	6	6000	95.4 (Five-fold CV)

set and testing set respectively. PCNN proposed by Wang et al. in 2016 has higher accuracy than contour combined with Fourier descriptor. The method we proposed obtains the highest accuracy of 95.2%.

#### 4.6.4 Test on ICL dataset

In this contrast experiment, Turkoglu and Hanbay [6] proposed different approaches based on LBP (RM-LBP, OM-LBP and ROM-LBP) for the recognition of plant leaves using extracted texture features. The method of PCA+LDA [14] combined principal component analysis and linear discriminant analysis to reduce the dimension of the features. Zhang et al. [19] proposed a two-stage method that Jaccard distance based sparse representation (JSDR). Zhang et al. [33] combined local mean-based clustering and sparse representation based classification (LWSRC). TMMG [5] is the method proposed by Zhang et al., they proposed margin features and shape features and fused them together. Zhao et al. [53] presented a counting-based shape descriptor (CS) and captured the global and local shape information of leaf. They selected three different subsets of ICL for experiments, and the following results are the average of the three experiments.

These literatures selected different samples for testing, and the details are shown in Table 4. SGLP and JSDR used five-fold-cross validation to test on ICL dataset, and CS and LWSRC used half-fold-cross validation to test. Hence, we use five-fold cross-validation and half-fold cross-validation in our method to compare with other methods. Experimental results show that Our method is higher than JSDR 2% and higher than SGLP 0.1% in five-fold cross-validation. In the half-fold-cross validation, CS was significantly higher than LWSRC, while our method achieved a 1.7% improvement in accuracy compared with CS method. The number of

species and samples used in the method based on LBP and PCA+LDA is small, they have less recognition difficulty but lower accuracy. It is obvious that the method we proposed is better than other methods.

## 5 Conclusion

In this paper, we proposed a method combined Jaccard distance and BOW for leaf recognition. Jaccard distance is used to exclude the most dissimilar classes, not only reduce the amount of computation but also shorten the time consumption. BOW is used to extract features from texture image and contour image by Laws' texture measure and Sobel operator, the local and global features of the image are described. Conducted comparative experiments in many aspects, including parameter setting and robustness verification. Besides, we compared and analyzed our method with the existing method in four datasets. The experimental results show that our method has better recognition results in small and large datasets.

There is still room for improvement in our method. For example, the accuracy of calculating similar classes for test images using Jaccard distance cannot reach 100%, and then impact the recognition accuracy of whole dataset. We can try to make improvements to make the similarity calculation more precise. In addition, we only use texture features and contour features in this method, and we hope to add some new features in future studies to get better results.

## Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

1. Seeland, M., Rzanny, M., Alaqraa, N., Wäldchen, J., Mäder, P.: Plant species classification using flower images—a comparative study of local feature representations. *PLoS One* **12**(2), e0170629 (2017)
2. Bertrand, S., Ameer, R.B., Cerutti, G., Coquin, D., Valet, L., Tougne, L.: Bark and leaf fusion systems to improve automatic tree species recognition. *Ecol. Inform.* **46**, 57–73 (2018)
3. Saleem, G., Akhtar, M., Ahmed, N., Qureshi, W.: Automated analysis of visual leaf shape features for plant classification. *Comput. Electron. Agric.* **157**, 270–280 (2019)
4. Munisami, T., Ramsurn, M., Kishnah, S., Pudaruth, S.: Plant leaf recognition using shape features and colour histogram with k-nearest neighbour classifiers. *Proced. Comput. Sci.* **58**, 740–747 (2015). [second International Symposium on Computer Vision and the Internet (VisionNet'15)]
5. Zhang, X., Zhao, W., Luo, H., Chen, L., Fan, J.: Plant recognition via leaf shape and margin features. *Multimed. Tools Appl.* **78**(7), 27463–27489 (2019)
6. türkoğlu, M., Hanbay, D.: Leaf-based plant species recognition based on improved local binary pattern and extreme learning machine. *Phys. A Stat. Mech. Appl*
7. Cantero, S.V.A.B., Goncalves, D.N., Leonardo, F.: Importance of vertices in complex networks applied to texture analysis. *IEEE Trans. Cybern.* **20**, 20 (2018)
8. Florindo, J.B., Bruno, O.M.: Discrete schroedinger transform for texture recognition. *Inf. Sci.* **415–416**, 142–155 (2017)
9. Florindo, J.B.: Dstnet: Successive applications of the discrete schroedinger transform for texture recognition. *Inf. Sci.* **20**, 20 (2020)
10. Ghasab, M.A.J., Khamis, S., Mohammad, F., Fariman, H.J.: Feature decision-making ant colony optimization system for an automated recognition of plant species. *Expert Syst. Appl.* **42**(5), 2361–2370 (2015)
11. Liu, N., Kan, J.M.: Improved deep belief networks and multi-feature fusion for leaf identification. *Neurocomputing* **216**, 460–467 (2016)
12. Chaki, J., Parekh, R., Bhattacharya, S.: Plant leaf recognition using texture and shape features with neural classifiers. *Pattern Recognit. Lett.* **58**(C), 61–68 (2015)
13. Vijayalakshmi, B., Mohan, V.: Kernel-based pso and frvm: an automatic plant leaf type detection using texture, shape, and color features. *Comput. Electron. Agric.* **125**, 99–112 (2016)
14. Zhang, L., Zheng, Y., Zhong, G., Wang, Q.: Research on leaf species identification based on principal component and linear discriminant analysis. *Cluster Comput.* **20**, 20 (2017)
15. Larese, M.G., Granitto, P.M.: Finding local leaf vein patterns for legume characterization and classification. *Mach. Vis. Appl.* **27**(5), 1–12 (2015)
16. Pires, R.D.L., Gonçalves, D.N., Oruê, J.P.M., Kanashiro, W.E.S., Machado, B.B., Gonçalves, W.N.: Local descriptors for soybean disease recognition. *Comput. Electron. Agric.* **125**(C), 48–55 (2016)
17. Wang, Z., Sun, X., Yang, Z., Zhang, Y., Ying, Z., Ma, Y.: Leaf recognition based on dpcnn and bow. *Neural Process. Lett.* **47**(6), 1–17 (2017)
18. Xu, Y., Zhu, Q., Fan, Z., Zhang, D., Mi, J., Lai, Z.: Using the idea of the sparse representation to perform coarse-to-fine face recognition. *Inf. Sci.* **238**(7), 138–148 (2013)
19. Zhang, S., Wu, X., You, Z.: Jaccard distance based weighted sparse representation for coarse-to-fine plant species recognition. *PLoS One* **12**(6), e0178317 (2017)
20. Salleh, S.S., Aziz, N.A.A., Mohamad, D., Omar, M.: Combining Mahalanobis and Jaccard to improve shape similarity measurement in sketch recognition. In: *Uksim International Conference on Computer Modelling and Simulation*, pp. 319–324 (2011)
21. Laws, K.I.: Textured image segmentation. Technical Report USCCIP-940
22. Tang, L., Zhou, C.S., Zhang, L.: 20clustered domain colors and bag of words algorithm based image retrieval. *Appl. Mech. Mater.* **321–324**, 956–960 (2013)
23. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2, pp. 2169–2178 (2006)
24. Suresha, M.S.KN, Thirumalesh, B.V.: Recognition of diseases in paddy leaves using KNN classifier. In: *International Joint Conference on Neural Networks*, pp. 663–666 (2017)
25. Ruberto, C.D., Putzu, L.: A fast leaf recognition algorithm based on SVM classifier and high dimensional feature vector. In: *International Conference on Computer Vision Theory and Applications*, pp. 601–609 (2014)
26. Caglayan, A., Guclu, O., Can, A.: Image analysis and processing. In: *IEEE International Symposium on Signal Processing and Information Technology*, pp. 161–170 (2013)
27. Wu, S.G., Bao, F.S., Xu, E.Y., Wang, Y.X., Chang, Y.F., Xiang, Q.L.: A leaf recognition algorithm for plant classification using probabilistic neural network. In: *IEEE International Symposium on Signal Processing and Information Technology*, pp. 11–16 (2008)
28. Mittal, P., Kansal, M., kaur Hajji, H.: Combined classifier for plant classification and identification from leaf image based on visual attributes. In: *2018 International Conference on Intelligent Circuits and Systems (ICICS)*, pp. 184–187 (2018)
29. Wang, Z., Sun, X., Ma, Y., Zhang, H., Ma, Y., Xie, W.: Plant recognition based on intersecting cortical model. In: *International Joint Conference on Neural Networks*, pp. 975–980 (2014)
30. Wu, S.G., Bao, F.S., Xu, E.Y., Wang, Y.X., Xiang, Q.L.: A leaf recognition algorithm for plant classification using probabilistic neural network. In: *IEEE International Symposium on Signal Processing and Information Technology*, pp. 11–16 (2008)
31. Söderkvist, O.: Computer vision classification of leaves from Swedish trees. *Teknik Och Teknologier*
32. Novotný, P., Suk, T.: Leaf recognition of woody species in central europe. *Biosyst. Eng.* **115**(4), 444–452 (2013)
33. Zhang, S., Wang, H., Huang, W.: Two-stage plant species recognition by local mean clustering and weighted sparse representation classification. *Cluster Comput.* **20**(2), 1517–1525 (2017)
34. Hsiao, J.K., Kang, L.W., Chang, C.L., Lin, C.Y.: Comparative study of leaf image recognition with a novel learning-based approach. In: *Science and Information Conference*, pp. 389–393 (2014)
35. Turkoglu, M., Hanbay, D.: Recognition of plant leaves: an approach with hybrid features produced by dividing leaf images into two and four parts. *Appl. Math. Comput.* **352**(C), 1–14 (2019)
36. Demisse, G.G., Aouada, D., Ottersten, B.: Deformation based curved shape representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**, 1 (2018)
37. Kumar, P.S. V. V. S.R., Rao, K.N.V., Raju, A.S.N., Kumar, D.J.N.: Leaf classification based on shape and edge feature with K-NN classifier. In: *International Conference on Contemporary Computing and Informatics*, pp. 548–552 (2017)
38. Wang, Z., Sun, X., Zhang, Y., Ying, Z., Ma, Y.: Leaf recognition based on pcnn. *Neural Comput. Appl.* **27**(4), 899–908 (2016)
39. Tsolakidis, D.G., Kosmopoulos, D.I., Papadourakis, G.: Plant leaf recognition using zernike moments and histogram of oriented gradients. *Artif. Intell. Methods Appl.* **20**, 20 (2014)

40. Yousefi, E., Baleghi, Y., Sakhaei, S.M.: Rotation invariant wavelet descriptors, a new set of features to enhance plant leaves classification. *Comput. Electron. Agric.* **140**, 70–76 (2017)
41. Naresh, Y.G., Nagendraswamy, H.S.: Classification of medicinal plants: an approach using modified lbp with symbolic representation. *Neurocomputing* **173**, 1789–1797 (2016)
42. Kolivand, H., Fern, B.M., Saba, T., Rahim, M.S.M., Rehman, A.: A new leaf venation detection technique for plant species classification. *Arab. J. Sci. Eng.* **44**(4), 3315–3327 (2019)
43. Pearline, A., Kumar, S.: Ddla:dual deep learning architecture for classification of plant species. *IET Image Process.* (2019). <https://doi.org/10.1049/iet-ipr.2019.0346>
44. Zhang, S., Zhang, C., Zhen, W., Kong, W.: Combining sparse representation and singular value decomposition for plant recognition. *Appl. Soft Comput.* **67**, S156849461830111X (2018)
45. Sun, G.D., Zhang, Y., Ping, L.I., Mei, S.Z., Zhao, D.X.: Feature description of exact height function used in fast shape retrieval. *Opt. Precis. Eng.* **25**(1), 224–235 (2017)
46. Shao, Y.: Supervised global-locality preserving projection for plant leaf recognition. *Comput. Electron. Agric.* **20**, 20 (2019)
47. Yu, X., Xiong, S., Gao, Y., Yang, Z., Yuan, X.: Multiscale crossing representation using combined feature of contour and venation for leaf image identification. In: International Conference on Digital Image Computing: Techniques and Applications, pp. 1–6 (2016)
48. Zeng, J., Liu, M., Fu, X., Gu, R., Leng, L.: Curvature bag of words model for shape recognition. *IEEE Access.* **20**, 11 (2019)
49. Wang, B., Brown, D., Gao, Y., Salle, J.L.: March: Multiscale-arch-height description for mobile retrieval of leaf images. *Inf. Sci.* **302**, 132–148 (2015)
50. Chengzhuan, Y., Hui, W.: Plant species recognition using triangle-distance representation. *IEEE Access*
51. Yang, C., Yu, Q.: Multiscale fourier descriptor based on triangular features for shape retrieval. *Signal Process. Image Commun.* **20**, 20 (2019)
52. Zheng, Y., Yan, B., He, W.: O2o method for fast shape retrieval. *IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc.* (2019). <https://doi.org/10.1109/TIP.2019.2919195>
53. Zhao, C., Chan, S.S.F., Cham, W.K., Chu, L.M.: Plant identification using leaf shapes—a pattern counting approach. *Pattern Recogn.* **48**(10), 3203–3215 (2015)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.