Weed Seeds Recognition via Support Vector Machine and Random Forest

Yilin Long and Cheng Cai*
*Department of Computer Science, College of Information Engineering, Northwest A&F University, Yangling, Shannxi, China
E-mail: cheney.chengcai@gmail.com  Tel: +86-29-87091641

Abstract—Weed seeds classification and analysis is essential to weed prevention and species hybridization for both agricultural and economical purposes. We propose to identify weed seeds with their color images using support vector machine and random forest. We extract 4 kinds of features from the images and use PCA to do dimension reduction, and then combine these features together to get another 11 kinds of features. After feature extraction we use support vector machine and random forest to do the classification task. With 10 repeated experiments, we demonstrate that both support vector machine and random forest can achieve ideal classification performance with proper features. And the complementary information provided by high-level features such as Gist, low-level features such as gradient histograms and RGB can help to obtain more discrimination power. We also discuss the unsteadiness and randomness of random forest in classification task.

I. INTRODUCTION

The classification and analysis of weed seeds are crucial activities dedicated to the added value in crop production and the filtered biological characteristics in species hybridization. There are about 8000 species of weeds in the world and the number of weed species which bring direct damage to crops or spread plant diseases and insect pests is nearly 1200 [1]. Recognizing weed seeds species in time can be very crucial to prevent the damage caused by weeds. On the other hand, many kinds of seeds can be useful to human beings, for example, some Chinese medicine is made from seeds which can serve as germplasm resources in species hybridization. For different parts of global procedure, such as seed production, cereal grading for industrialization or commercialization purposes, during scientific research for improvement of species, etc. these classification and analysis activities show their fundamental functions.

In the classification procedure, specialists are engaged. And the process of manual classification of weed seeds by experts is slow and has difficulty in quantification in its commercial and technological implications caused by a degree of subjectivity. Then it is technically and financially significant to execute computer based methods for credible and fast identification and classification of weed seeds.

Automatic systems can be based on seed images, from which classification features are easily obtained. Our predecessors have done plenty of attempts in tracking the classification problem based on seed images and the corresponding features such as seed size, shape, color and texture. Discriminant analysis, artificial neural network and Naïve Bayes classifiers were applied in weed seeds classification [2, 3, 4], but extracting features of seeds directly in these works is time consuming and difficult. Moreover, as mentioned in [5], these methods almost neglect the discriminant power of colors.

From 2009, some new methods utilized color and textual features in weed seeds classification task. PCA, 2DPCA, column-directional 2DPCA and (2D)2PCA and color PCA showed the color and textual features could achieve higher recognition accuracy than traditional features [6]. Locally Linear Embedding (LLE) method is also used in weed seeds recognition; compared with PCA, this method has a better performance [7]. And the compressive sensing theory applied in the task with dimension reduction can process weed seeds with corruption, which may help deal with real world weed seeds recognition issues [8]. However, the number of seed species in the experiment of these methods is so small that the classification result is limited.

In this paper, Support Vector Machine (SVM) and Random Forest (RF) classifier using color and textual based features are applied to weed seeds classification. The features we use include low-level pixel based features, mid-level features and high-level concept features. Low-level pixel based features include Histograms of Oriented Gradients (HOG) features [9] extracted from black and white images, and RGB features extracted from color images. Mid-level features we used are called “Sketch Tokens” extracted from color images mentioned in [10]. The high-level concept features are Gist extracted from color images mentioned in [11]. After feature extraction, we use Principal Component Analysis (PCA) to do the dimension reduction, and then we combine the four individual features together to obtain another 11 kinds of features. After these procedures, based on 15 kinds of features, we use support vector machine and random forest to track the classification problem. Compared with the previous work, support vector machine and random forest can process a test image fast and the recognition accuracy is also quite good. An interesting thing is, as mentioned in [10], linear support vector machine using the combination features of HOG and Sketch Tokens should have a higher recognition performance than that of HOG, but we find that HOG features perform better than the combination features do.
The remains of the paper are organized as follows. First, we introduce support vector machine and random forest classifier respectively. Then the feature extraction procedure is proposed. Next, we show the experiment results obtained from support vector machine and random forest. Finally, some discussions are shown.

II. CLASSIFICATION METHODS

A. Support vector machine classifier

Support Vector Machine (SVM) is a classical pattern recognition method which has been widely applied to many processes, such as pattern identification, regression analysis, function approximating, etc. [12]. Support vector machine generates a hyperplane or a set of hyperplanes in a high- or infinite-space used for classification, regression and other tasks.

In a classical two-class support vector machine classification issue, when some data points belong to one of the two given classes, the goal is to distinguish which category a new data point belongs to. In such a support vector machine, a data point is regarded as an \(n\)-dimensional vector, and we need to know whether we can separate these data points with a hyperplane whose dimension number is \((n-1)\). Obviously, there exists a huge number of hyperplanes that could classify the data. The best hyperplane is the one that represents the largest separation, or the margin between two classes. Such a hyperplane is known as the maximum-hyperplane and the corresponding linear classifier is known as the maximum margin classifier, or the perceptron of optimal stability.

Using kernel trick technology helps to create a nonlinear classifier. Applying a kernel trick to the maximum-margin hyperplane generates a nonlinear algorithm which is similar to the linear support vector machine algorithm, except that the dot product is replaced by the nonlinear kernel function. This approach helps to compute the maximum-margin hyperplane in a transformed feature space.

Compared with the two-class support vector machine, the multi-class support vector machine is more practical. The key point of this approach is to reduce the single multi-class problem into multiple binary classification problems. To achieve this target in our experiment, the “one-versus-one” strategy is applied: building binary classifiers to distinguish each pair of classes, and the classification is done by a max-wins voting strategy [13].

Parameter selection. The selection of a kernel, the parameters of the kernel, and the soft margin parameter \(C\) are all pivotal to the effectiveness of the support vector machine. Gaussian kernel with a single parameter \(\gamma\) is a common choice. Grid searching with exponentially growing sequences of \(C\) and \(\gamma\) is usually used for selecting the best combination of the parameters. And cross validation is used for checking each combination of parameter choices, then the parameters with the best cross-validation accuracy are picked.

In our experiment, the linear support vector machine and RBF support vector machine (the kernel used is Gaussian radial basis function) are used after parameter selection.

B. Random forest classifier

Recently random forests have been an essential approach for automatic image analysis, such as classification, regression, semi-supervised learning, density estimation, manifold learning, active learning, etc. [14]. Random forests are a collection of decision trees. For classification, random forests can naturally process pattern recognition problems with more than two classes.

A decision tree consists of an internal node which denotes a test on an attribute, and a leaf node which holds a class label.
Given a new instance $X$ whose class label is unknown, testing attribute values of the instance against the decision tree can trace a path from the root to a leaf node which holds the category prediction for the instance [15].

For classification, three key elements of decision tree are data partition, feature list and split functions. Initially, data partition is a complete set of training instances and their corresponding class labels. The feature list describes the attributes of the instances. Split function uses a heuristic procedure to select the “best” attribute of the given instances according to class, which has the most discrimination power in the feature list.

In random forests, constructing each tree depends on the values of a random vector sampled from the original feature vector independently. And bagging in tandem with random attribute selection can be used in building the random forests. To generate the sampled instance, random attribute selection is employed. Let $N$ be the total number of the attributes, and the number of attributes used for split function at each node is much smaller than $N$. This procedure is done via random selection. For testing a formally unseen instance, random forests use the voting strategy.

In our experiment, the bagging strategy used in constructing random forest consists of 2000 decision trees with randomly selected $\sqrt{N}$ attributes.

III. IMAGE PREPROCESSING AND FEATURE EXTRACTION

A. Image Database

We use a subset of the weed seed image dataset from the Seed Analysis Laboratory at EEA of Oliveros of INTA[5], which contains 4024 color images from 91 different species. The size of each image is 768 pixel $\times$ 512 pixel, whose detail information can be gained from [5].

B. Image pre processing

Notice that the linear support vector machine is highly sensitive to how the features are normalized [10], so how the samples are standardized is crucial to the recognition result. To achieve some good-normalized features, first we should standardize the original weed seed images.

Fig. 1 is a flowchart showing how to gain standardized weed seed images. First we change the seed directions manually, so that images of each species will have the same seed direction. Then rotation, detection, cropping and resizing are then performed to standardize the weed seed images.

Image rotation. In image rotation, the principle component analysis (PCA) is used to find the two main axes of the original seed image. Then the degree of the included angle between the longer main axe and the vertical direction is computed. Based on the degree, we rotate the original seed image in order to make the longer main axe appear in the vertical direction and gain the rotated image.

Object detection and standardization. We detect and extract the rectangle seed area (the green box in Fig. 1) through the binary image corresponding to the rotated image. And the rectangle seed area is extended to make the rectangle seed area have the height-width ratio of 3 : 2. Then, cropping and resizing procedure is executed on the rotated image, and then the final image for our experiment with 96 pixel $\times$ 64 pixel is generated.

C. Feature extraction

Features used for recognition in our experiment include GIST [11], HOG [9], Mid-Level [10], RGB features and the combination of them. And PCA is used for feature dimension reduction. The dimension of these four kinds of features are all reduced to 512 via PCA. This step can guarantee each kind of individual features has the equal influence on the recognition accuracy of the combination features. Moreover, after reducing the PCA feature dimension, the experiment speed is boosted due to the low feature dimension.

GIST features. In [11], GIST features are used for scene categorization, which represent the dominant spatial structure of a scene with low perceptual dimensions. And the GIST descriptor is insensitive to the object shape or identity for categorization. Meanwhile, this model involves a holistic representation of the scene about its probable semantic category. As showed in Fig.1, the weed seed images dataset consists of images which have single foreground object and pure black background. For this reason, GIST features were selected for weed seed image classification.

Mid-Level features. One kind of Mid-Level features called Sketch Tokens (STs) used in our experiment is advanced in [10]. Mid-Level features act as a bridge between Low-Level pixel-based information and High-Level concepts. And they are the foundation of both bottom-up processing and top-down tasks, e.g. contour classification [16] or pixel-level segmentation [17]. The STs are learned using the supervised mid-level information, and the experiment result in [10] shows that the detection accuracy improves largely in object detection due to the complementary information provided by STs to Low-Level features such as gradient histograms. In this view, Sketch Tokens is taken into account for achieving high recognition accuracy. And the combination features of Sketch Tokens and HOG are considered as well.

RGB features. We select the raw pixel values of an image in RGB color space for the recognition task. These raw pixel features have several advantages: the resulting representation can be computed very fast, which does not discard any informational contents, and in addition, these features are also not application specific. Due to these benefits, RGB features are joined in the classification mission.

The combination of GIST, HOG, Mid-Level and RGB features. After the feature dimension reduction, we combine these 4 kinds of features together to generate 11 new features. Now we have 15 kinds of features in total, and each kind of features per image is normalized to have the unit Euclidean norm.

IV. EXPERIMENT AND DISCUSSION
A. Experiment results

We use LIBSVM [18] for the support vector machine recognition task, and TreeBagger class in Matlab Statistics Toolbox\(^1\) for Random Forest recognition task. To obtain a steady experiment result, we experiment on randomly selected training and testing dataset 10 times, and then average these results together. For each experiment, we randomly select 28 color images from each species for training and the remains for testing. Table I shows the average recognition accuracy of the 10 experiments.

The best recognition rate is 95.265% using the linear support vector machine with the combination features from Gist, Hog and RGB. And at most cases the support vector machine classifier achieves better performance than does the random forest.

B. Discussion

From Table I we can draw some conclusions. Firstly, some combination features show the complementary information which helps to strengthen the discrimination power to obtain better classification performance than do such single features as GistHog, GistRgb, GistHogRgb, HogRgb. However, the combination feature HogMidLevel does not exhibit the power of the complementary information from each other mentioned in [10] to achieve higher recognition accuracy. Secondly, the performance of the random forest classifier is not as good as that of the support vector machine, this may be due to the unsteadiness and the randomness character of the random forest. In our experiment, we find that the recognition result of the random forest with the same parameter and the same feature is not constant during multiple attempts. So this unsteadiness and randomness may exert some influence on the final result.

V. Conclusions

In this paper, the supported vector machine and random forest for weed seeds recognition are discussed. The experiment results show that by using the support vector machine, the combination features from Gist, Hog and RGB yield quite good performance, while the single features show less discrimination power. And the performance of the random forest is a little inferior to that of the support vector machine due to the unsteadiness and the randomness of random forest. This issue needs to be improved in future work.

In conclusion, we have described the support vector machine and random forest in weed seeds recognition task. Using different features, we have demonstrated quite good performance of the support vector machines. And to obtain a better random forest classifier for weed seeds recognition, we need to do more jobs in the future.

ACKNOWLEDGMENT

This work is partially supported by National Natural Science Foundation of China (Project 61202188).

REFERENCES


<table>
<thead>
<tr>
<th>Features</th>
<th>Linear SVM Accuracy (%)</th>
<th>RBF SVM Accuracy (%)</th>
<th>RF Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gist</td>
<td>91.208</td>
<td>91.593</td>
<td>91.98</td>
</tr>
<tr>
<td>GistHog</td>
<td>94.161</td>
<td>94.16</td>
<td>92.50</td>
</tr>
<tr>
<td>GistHogMidLevel</td>
<td>91.792</td>
<td>91.801</td>
<td>89.49</td>
</tr>
<tr>
<td>GistHogMidLevelRgb</td>
<td>93.974</td>
<td>93.993</td>
<td>91.42</td>
</tr>
<tr>
<td>GistHogRgb</td>
<td>95.265</td>
<td>95.21</td>
<td>94.10</td>
</tr>
<tr>
<td>GistMidLevel</td>
<td>87.319</td>
<td>87.376</td>
<td>87.31</td>
</tr>
<tr>
<td>GistRgb</td>
<td>93.711</td>
<td>93.975</td>
<td>92.60</td>
</tr>
<tr>
<td>GistRgbMidLevel</td>
<td>91.818</td>
<td>91.707</td>
<td>89.98</td>
</tr>
<tr>
<td>Hog</td>
<td>92.934</td>
<td>92.887</td>
<td>90.16</td>
</tr>
<tr>
<td>HogMidLevel</td>
<td>89.223</td>
<td>89.185</td>
<td>85.75</td>
</tr>
<tr>
<td>HogMidLevelRgb</td>
<td>92.897</td>
<td>92.895</td>
<td>89.41</td>
</tr>
<tr>
<td>HogRgb</td>
<td>94.723</td>
<td>94.667</td>
<td>93.53</td>
</tr>
<tr>
<td>MidLevel</td>
<td>73.592</td>
<td>74.077</td>
<td>67.48</td>
</tr>
<tr>
<td>MidLevelRgb</td>
<td>88.097</td>
<td>88.08819</td>
<td>83.90</td>
</tr>
<tr>
<td>Rgb</td>
<td>90.236</td>
<td>90.76</td>
<td>88.85</td>
</tr>
</tbody>
</table>

\(^1\) Documents available at http://www.mathworks.com/help/stats/treebagger.html


