This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2018.2809688, IEEE Signal Processing Letters

A Multi-Scale Fusion Convolutional Neural Network for Plant Leaf Recognition

Jing Hu, Zhibo Chen*, Meng Yang, Rongguo Zhang and Yaji Cui

Abstract—**Plant leaf recognition is a computer vision task used to automatically recognize plant species. It is very challenging since rich plant leaf morphological variations, such as sizes, textures, shapes, venation, and so on. Most existing plant leaf methods typically normalize all plant leaf images to the same size and recognize them at one scale, resulting in unsatisfactory performances. In this paper, a multi-scale fusion convolutional neural network (MSF-CNN) is proposed for plant leaf recognition at multiple scales. First, an input image is down-sampled into multiples low resolution images with a list of bilinear interpolation operations. Then, these input images with different scales are step-by-step fed into the MSF-CNN architecture to learn discriminative features at different depths. At this stage, the feature fusion between two different scales is realized by a concatenation operation, which concatenates feature maps learned on different scale images from a channel view. Along with the depth of the MSF-CNN, multi-scale images are progressively handled and the corresponding features are fused. Third, the last layer of the MSF-CNN aggregates all discriminative information to obtain the final feature for predicting the plant specie of the input image. Experiments show the proposed MSF-CNN method is superior to multiple state-of-the art plant leaf recognition methods on the MalayaKew (MK) Leaf Dataset and the LeafSnap Plant Leaf Dataset.**

Index Terms—**Plant leaf recognition; multi-scale convolutional neural network; multi-scale feature**

I. Introduction

In botany, plants are usually recognized according to the shapes of their leaves and flowers. Botanists usually use variations on leaf characteristics as a comparative tool for their study on plant recognition [1, 2]. For this, a lot of computer vision researchers have used plant leaf images as a comparative tool to recognize plant species [3–5].

The plant leaf image feature representation is a crucial component of a plant leaf recognition algorithm. There are two main types of feature representation methods for describing leaf images, hand-crafted features [6–8] and deep learning features [9–12]. In practise, the design of hand-crafted features is more dependent on the ability of computer vision experts to encode morphological characters those pre-defined by botanists [11]. However, the deep learning features are able to be automatically learned based on the end-to-end advantage of deep learning algorithms. Therefore, deep learning based plant leaf recognition methods received more and more attention.

This work was supported in part by the National Natural Science Foundation of China under the Grants 61402038.

Jing Hu is with the School of Information Science & Technology, Beijing Forestry University and College of Computer Science & Technology, Taiyuan University of Science and Technology (e-mail: 279641292@qq.com).

Zhibo Chen* (corresponding author) is with the School of Information Science & Technology, Beijing Forestry University (e-mail: zhibo@bjfu.edu.cn).

Meng Yang is with School of Information Science & Technology, Beijing Forestry University (e-mail: zhibo@bjfu.edu.cn).

Rongguo Zhang is with the College of Computer Science & Technology, Taiyuan University of Science and Technology (e-mail: rg-zh@163.com).

Yaji Cui and Yuxi Sun are with the Beijing University of Posts and Telecommunications, No.10, Xitucheng Road, Haidian District, Beijing, China (e-mail: 13683290326@163.com and sunyuxi_syx@163.com).

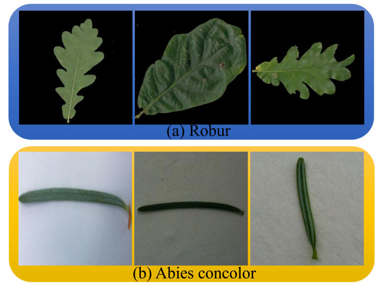

Fig. 1. Plant leaf samples from the MalayaKew (MK) Leaf Dataset [10, 11] (i.e. Robur) and the LeafSnap Plant Leaf Dataset [3] (i.e. Abies concolor).

As shown in Fig. 1, plants of different species have different sizes. For example, the robur leaf is bigger than the abies concolor leaf. Moreover, due to different growth conditions or different shooting distances, even for the same species, plants still have different sizes, as shown in Fig. 1 (a). Facing a wide variety of plant leaves, people must take the observation scale into account to find suitable scales for recognizing plants. However, most existing deep learning based plant leaf recognition methods focus on learning feature at single scale, which leaves a room for improving the accuracy.

In this paper, we first propose a multi-scale fusion convolution neural network (MSF-CNN) for plant leaf recognition at multiple scales. Then, comprehensive experiments are evaluated to show the superiority of the proposed MSF-CNN method and analyze the internal mechanism of MSF-CNN.

II. Related Work

A. Hand-crafted Features

There are many hand-crafted features for plant leaf images, and they can be divided into three types, shape [6, 13–15], texture [7, 16, 17] and venation [2, 8, 18].

For describing plant leaf shape characteristics, Neto et al. [6] introduced Elliptic Fourier and discriminant analyses to distinguish different plant species based on their leaf shape. Shape context (SC) and histogram of oriented gradients (HOG) have also been used to design a leaf shape descriptor [13, 14]. Kumar et al. [3] proposed the Histogram of Curvature over Scale (HoCS) to analyse leaves. Cope et al. [15] analysed leaf margins for species classification.

For modeling plant leaf texture characteristics, Cope et al. [17] applied Gabor co-occurrence features to describe plant

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2018.2809688, IEEE Signal Processing Letters

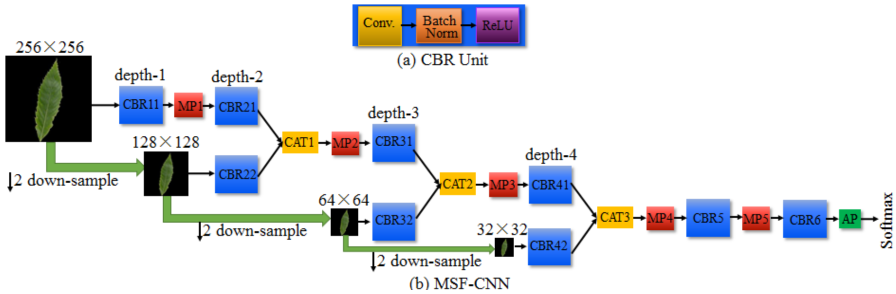

Fig. 2. The architecture of the proposed multi-scale fusion convolutional neural network (MSF-CNN). (a) CBR unit. (b) The diagram of the proposed MSF-CNN, where CAT donates a concatenation layer used to concatenate two convolutional layers from a channel view.

textures. Naresh and Nagendraswamy [7] modified the conventional local binary pattern (LBP) approach to consider the structural relationship between neighboring pixels, replacing the hard threshold approach of conventional LBP.

The plant leaf venation structure is also a widely adopted leaf characteristics. Charters et al. [8] designed a novel descriptor called EAGLE. It consists of five sample patches that are arranged to capture and extract the spatial relationships between local areas of venation. They showed that a combination of EAGLE and SURF was able to boost the discriminative ability of the feature representation. Larese et al. [18] recognised legume varieties based on leaf venation. They first segmented the vein pattern using Hit or Miss Transform (UHMT), then extract a set of features for veins using LEAF GUI measures [18].

B. Deep Learning Features

Recently, several deep learning based plant leaf recognition methods [9–12] have been proposed. Liu et al. [9] applied a traditional convolutional neural network (CNN) to extract features and then used a support vector machine (SVM) to classify leaf images. Guillermo et al. [12] first segmented the vein pattern using Hit or Miss Transform (UHMT) to obtain binary vein segmented images. Then, they trained a CNN on the binary vein segmented images, rather than on original input images. Lee et al. [10] proposed the DeepPlant network to recognize plant leaf images. Moreover, they applied the de-convolutional network (DN) [19] to gain the intuition on the chosen features from the CNN model. Based on their previous work [10], Lee et al. [11] further proposed a two-stream convolutional neural network (TwoCNN). In TwoCNN [11], the two feature learning streams are learned on whole and patch images, respectively. Although, TwoCNN is able to capture discriminative information at different scales (i.e. whole and patch), its' training progress requires a more complex sample collecting work that must prepare both whole and patch images.

Considering multi-scale features, both Du et al. [20] and Rasti et al. [21] proposed multi-scale convolutional neural networks (MSCNNs). There MSCNNs is composed of multiple different scale feature learning branches. In [20], each branch learns on different sized image patches, while in [21], each branch learns on different sized input images. The main difference among these MSCNNs and our MSF-CNN is that the multi-scale features learned by these MSCNNs are fused at one inter fusion layer, while MSF-CNN are step-by-step fused.

TABLE I
THE PARAMETER DETAILS OF THE PROPOSED MSF-CNN. h, w AND c REPRESENT HEIGHT, WIDTH, CHANNEL SIZES,RESPECTIVELY.

name	output size ($h/w/c$)	filter size ($h \times w$)/ stride/pading	pooling size ($h \times w$)/ stride/pading
CBR1	256/256/64	$3 \times 3/1/1$	-
MP1	128/128/64	-	$3 \times 3/1/1$
CBR21	128/128/64	$3 \times 3/1/1$	-
CBR22	128/128/64	$3 \times 3/1/1$	-
CAT1	128/128/128	-	-
MP2	64/64/128	-	$3 \times 3/1/1$
CBR31	64/64/128	$3 \times 3/1/1$	-
CBR32	64/64/128	$3 \times 3/1/1$	-
CAT2	64/64/256	-	-
MP3	32/32/256	-	$3 \times 3/1/1$
CBR41	32/32/160	$3 \times 3/1/1$	-
CBR42	32/32/160	$3 \times 3/1/1$	-
CAT3	32/32/320	-	-
MP4	16/16/320	-	$3 \times 3/1/1$
CBR5	16/16/224	$3 \times 3/1/1$	-
MP5	8/8/224	-	$3 \times 3/1/1$
CBR6	8/8/256	$3 \times 3/1/1$	-
AP	4/4/256	-	$4 \times 4/1/1$

Therefore, the proposed MSF-CNN is calculation economical, since it does not require multiple feature learning branches.

III. Multi-Scale Fusion Convolutional Neural Network

A. MSF-CNN Architecture

For conveniently describing, we construct our deep feature learning architecture with three basic units, CBR, MP (max pooling) and AP (average pooling) units, as shown in Fig. 2. As shown in Fig. 2 (a), one can see that the CBLR unit consists of convolutional, batch normalization [22] and ReLU [23] layers. In the proposed MSF-CNN, the size of filters in all convolutional layers is 3×3, the size of max pooling windows in all MP layer is also 3×3, and the size of average pooling window in the AP layer is 4×4. Moreover, the strides for convolutional and MP layers are 1 and 2 pixels, respectively. The bilinear interpolation method is applied to realize the image down-sampling. The parameter details of the proposed multi-scale fusion convolutional neural network (MSF-CNN) architecture is listed in Table I.

As shown in Fig. 2 (b), four scale (i.e. 256×256, 128×128, 64×64 and 32×32) input images are step-by-step feeded into the multi-scale fusion convolutional neural network (MSF-CNN) architecture to learn discriminative features at different depths. To be more specifically, 256×256, 128×128, 64×64 and 32×32 sized input images are feeded into the deep learning architecture at the 1st, 2nd, 3rd and 4th depths, respectively. Furthermore, the feature fusion between

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2018.2809688, IEEE Signal Processing Letters

two different scales is realized by a concatenation operation, which concatenates feature maps (e.g. outputs of CBL21 and CBL22) from a channel view. Based on this multi-scale fusion mechanism, multi-scale images are progressively handled at different depths, and the last layer aggregates all discriminative information to obtain the final features.

B. Objective Function

The Softmax [24] is applied as the objective function, which is formulated as follows:

$$J(W) = -\frac{1}{N}[\sum_{n=1}^{N}\sum_{c=1}^{C}\ell(y_n == c)\log\frac{e^{W_c^{\mathrm{T}}X_i}}{\sum\limits_{p=1}^{C}e^{W_p^{\mathrm{T}}X_i}}] \quad (1)$$

where $(X_1, y_1), X_2, y_2), ..., X_N, y_N)$ is the training set; x_n is n-th training sample and $y_n \in 1, 2, 3, ..., C$ is the corresponding label. N and C represents the numbers of the training samples and classes; $\ell(\cdot)$ is an indicator function. Note that Eq. (1) is cooperated with the proposed MSF-CNN architecture and they are optimized by using the stochastic gradient descent [24] algorithm and deep learning toolbox is Matconvnet [25].

IV. Experiment and Analysis

A. Dataset and Evaluation Protocol

MalayaKew (MK) Leaf Dataset [10, 11] includes three subsets, named MK-D1, MK-D2 and MK-D3. The MK-D1 subset includes 2288 training images and 528 testing images, and each image describes a whole leaf. The MK-D2 subset consists of 34672 training samples and 8800 testing samples, and each sample is local leaf patch. The MK-D3 subset is mixture version of MK-D1 and MK-D2. To be more specific, the training set of MK-D3 includes 1324 whole leaf images and 3960 leaf patch images, which ensures each plant is recorded with one whole image and 3 local patches. Moreover, the testing set of MK-D3 is same with that of the MK-D1. The organization the training set of MK-D3 is convenient for training a two-stream convolutional neural network (TwoCNN) [11]. In [11], the two feature learning streams are learned on whole and patch images, respectively. On the MK Leaf Dataset, the top-1 classification accuracy is computed to evaluate performances, which is calculated as follows.

$$Accuracy\ rate = T_r/T_n, \quad (2)$$

where T_r is the number of true species predictions and T_n is total number of images evaluated.

LeafSnap Plant Leaf Dataset [3] consists of 7719 field images taken by mobile devices and 23147 lab images captured using a high-quality camera. In this paper, only the field images are applied to evaluate the proposed method, since the field is more closer to reality. Because the Leafsnap does not provide the partition of training, validation and testing sets. In this paper, the experiment are constructed by randomly dividing the field set into 2 parts: half for training and the other half for testing. Moreover, the testing set is further divided into two parts: half for gallery and half probe sets. This procedure is repeated 5 times and the widely adopted average cumulative match curve (CMC) [26, 27] is calculated as the final result. The Euclidean distance is used as the similarity measurement between a pair of plant images.

TABLE II
Accuracy rate (%) comparison on MK-D1. DL represents a deep learning based method.

Method	DL	Accuracy Rate (%)
MSF-CNN	Y	**99.05**
DeepPlant+SVM [11]	Y	98.1
DeepPlant+MLP [11]	Y	97.7
Combine (SVM (linear)) [4]	N	95.1
LeafSnap (SVM (RBF)) [3]	N	42.0
LeafSnap (NN) [3]	N	58.9
HCF (SVM (RBF)) [4]	N	71.6
HCF-ScaleRobust (SVM (RBF)) [4]	N	66.5
SIFT (SVM (linear)) [28]	N	58.8

TABLE III
Accuracy rate (%) comparison on MK-D2.

Method	DL	Accuracy Rate (%)
MSF-CNN	Y	**99.82**
DeepPlant+MLP [11]	Y	99.5
DeepPlant+SVM [11]	Y	99.3

B. Training Configuration

All images are scaled to 256×256 pixels, and each image is further augmented by the horizontal mirror operation. We initialize the weights in each layer based on a normal distribution $N(0, 0.01)$, and the biases are initialized to 0. The ℓ_2 regularization weight in the softmax objective function is set to 0.0005. The size of mini-batch is 128 images, and each mini-batch is randomly selected from the whole dataset. The momentums are set to 0.9. The learning rates start with 0.01 and gradually decreased as the training progress and the minimum learning rate is 0.0001.

C. Comparison with State-of-the-Art Methods

1) Results on MK-D1: From Table II, one can find that among those non deep learning (DL) based plant leaf recognition methods, the best result (i.e. accuracy rate is 95.1%) is obtained by the Combine (SVM (linear)) [4]. However, it is beaten by the three deep learning (DL) based plant leaf recognition methods, as shown in Table II. Moreover, among the three DL based plant leaf recognition methods, the proposed MSF-CNN method outperforms the DeepPlant+SVM [11] and DeepPlant+MLP [11] with 0.95% and 1.35% accuracy rates, respectively. In additionally, compared with DeepPlant [11], we do not train a MLP or SVM classifier on the learned features to improve the accuracy rate.

2) Results on MK-D2: The accuracy rate performance comparison of the proposed MSF-CNN and DeepPlant [10] on the MK-D2 dataset is shown in Table III. Due to more training samples and deduction background interferences, both MSF-CNN and DeepPlant [10] acquire better results than those on the MK-D1 dataset. Moreover, one can see that the proposed MSF-CNN method still consistently beats DeepPlant+MLP [10] and DeepPlant+SVM [10] methods on MK-D2.

3) Results on MK-D3: As shown in Table IV, it can be seen that MSF-CNN obtains the highest accuracy rate 97.35%, outperforming the second (i.e. TwoCNN-EF (conv-sum) [11]) and third (Finetuned AlexNet [24]) places with 1.05% and 1.75% accuracy rates, respectively. Besides, the TwoCNN-EF (conv-sum) [11]) method ask for simultaneously feeding whole and patch leaf images into the two stream CNN, which means that the TwoCNN-EF (conv-sum) method requires a more complex sample collecting preparation. Furthermore, the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2018.2809688, IEEE Signal Processing Letters

TABLE IV
ACCURACY RATE (%) COMPARISON ON MK-D3.

Method	DL	Accuracy Rate (%)
MSF-CNN	Y	**97.35**
TwoCNN-EF(conv-sum) [11]	Y	96.3
Finetuned AlexNet [24]	Y	95.6
TwoCNN-EF(cascade) [11]	Y	95.5
TwoCNN-LF(ave) [11]	Y	94.5
TwoCNN-LF(mav) [11]	Y	94.1

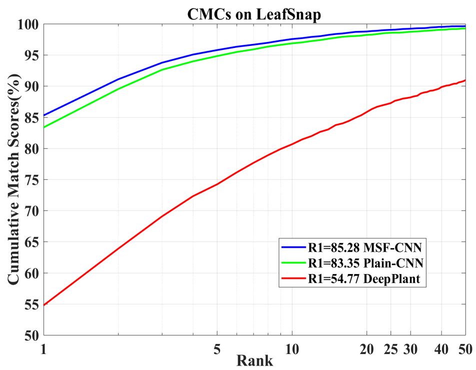

Fig. 3. CMC curves and rank-1 identification (R1) rates on LeafSnap.

Finetuned AlexNet [24] method is with the help of pre-trained AlexNet learned on the large scale ImageNet database. Contrary to these methods, the proposed method does not require a complex sample collecting preparation or pre-training.

4) Results on LeafSnap: The performance comparison on the LeafSnap dataset is shown in Fig. 3. Different from directly citing results presented in [11], in this section, we re-trained the DeepPlant [10] on LeafSnap, using the same training configuration introduced in [11]. In the our training progress, we tried our best to train the DeepPlant network, and we found that DeepPlant was very difficult in converge, and 10000 mini-batches are implemented to obtain the DeepPlant model. We believe this is because the DeepPlant network does apply batch normalization layers [22] and includes too many parameters in the fully connection layers. One can see that the rank-1 identification rate of the proposed MSF-CNN method is 85.28%, which is 30.51% higher than that of the DeepPlant method. Moreover, one can find that MSF-CNN obtains a better CMC curve than DeepPlant.

D. Analysis of the Proposed Method

In the follows, we make a comprehensive performance analysis to show the role of multi-scale fusion mechanism in the proposed MSF-CNN method and the generalization ability of the proposed MSF-CNN mehtod. For the convenience of describing, the deep learning configuration that keeps the same basic architecture (e.g. CBR1, CBR21, CBR31, CBR41, CRB5 and CBR6, etc.) with the proposed MSF-CNN but does not use the multi-scale fusion mechanism is denoted as Plain-CNN.

1) Role of Multi-scale Fusion: As shown in the Table V, the proposed MSF-CNN consistently outperforms Plain-CNN. For example, the accuracy rates of MSF-CNN are 0.96%, 0.40% and 2.08% higher than those of Plain-CNN on the MK-D1, MK-D2 and MK-D3 databases, respectively. Moreover, one can find that the MSF-CNN also obtains a higher rank-1 identification rate (i.e. 85.28%) and a better CMC curve than those of Plain-CNN on the LeafSanp dataset, as show in Fig. 3. Based on these results, it can be clearly illuminated

TABLE V
ACCURACY RATE (%) COMPARISON OF THE PROPOSED MSF-CNN AND PLAIN-CNN METHODS ON DIFFERENT DATASETS.

Method	Dataset	Accuracy Rate (%)
MSF-CNN	MK-D1	**99.05**
Plain-CNN	MK-D1	98.11
MSF-CNN	MK-D2	**99.82**
Plain-CNN	MK-D2	99.42
MSF-CNN	MK-D3	**97.35**
Plain-CNN	MK-D3	95.27

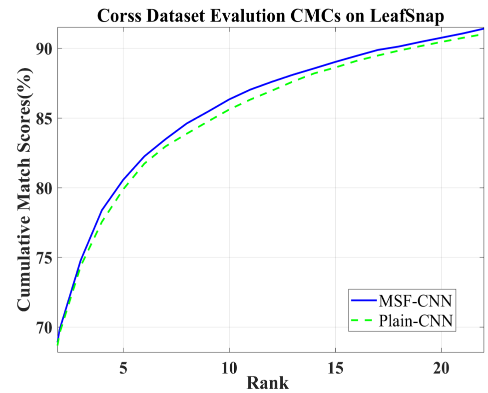

Fig. 4. Cross dataset (i.e. training on MK-D2 and testing on LeafSanp) evaluation average cumulative match characteristic (CMC) curve.

that the proposed multi-scale fusion mechanism is effective for improving the plant leaf recognition accuracy.

2) Generalization Ability: As shown in Table V, one can see that both MSF-CNN and Plain-CNN methods obtain nearly 100% accuracy rates. Therefore, we further compare the generalization ability of MSF-CNN and Plain-CNN with a cross dataset CMC evaluation. More specifically, both MSF-CNN and Plain-CNN are trained using the training set of the MK-D2 dataset, while they are tested on non-overlap LeafSnap dataset. As shown in Fig. 4, one can find that the proposed MSF-CNN method is able to beat the Plain-CNN method, under the cross dataset evaluation condition. This result illuminates that with the help of the proposed multi-scale fusion mechanism, the proposed MSF-CNN method is able to obtain a better generalization ability.

V. CONCLUSION

In this paper, a multi-scale fusion convolutional neural network (MSF-CNN) is proposed for plant leaf recognition. In the proposed MSF-CNN, an input image is down-sampled into multiple low resolution images and these input images are step-by-step feeded into the MSF-CNN architecture to learn discriminative features at different depths. The feature fusion between two different scales is realized by a concatenation operation, which concatenates feature maps learned on different scale images from a channel view. The last layer of MSF-CNN aggregates all discriminative information to obtain the final feature to predict the plant specie of the input image. On two public plant leaf datasets, MalayaKew (MK) and LeafSnap, comprehensive experiments have shown that the proposed the MSF-CNN method is superior to multiple state-of-the-art plant leaf recognition methods. Moreover, a cross dataset evaluation (i.e. training on the MK dataset and testing on the LeafSnap dataset) is also implemented to validate the superiority of the propose MSF-CNN method on generalization ability.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2018.2809688, IEEE Signal Processing Letters

References

[1] J. S. Cope, D. Corney, J. Y. Clark, P. Remagnino, and P. Wilkin, "Review: Plant species identification using digital morphometrics: A review," *Expert Systems with Applications An International Journal*, vol. 39, no. 8, pp. 7562–7573, 2012.

[2] J. Clarke, S. Barman, P. Remagnino, K. Bailey, D. Kirkup, S. Mayo, and P. Wilkin, "Venation pattern analysis of leaf images," in *International Conference on Advances in Visual Computing*, 2006, pp. 427–436.

[3] N. Kumar, P. N. Belhumeur, A. Biswas, D. W. Jacobs, W. J. Kress, and I. C. Lopez, "Leafsnap: a computer vision system for automatic plant species identification," in *European Conference on Computer Vision*, 2012, pp. 502–516.

[4] D. Hall, C. Mccool, F. Dayoub, and N. Sunderhauf, "Evaluation of features for leaf classification in challenging conditions," in *IEEE Winter Conference on Applications of Computer Vision*, 2015, pp. 797–804.

[5] C. Kalyoncu and nsen Toygar, "Geometric leaf classification ," *Computer Vision and Image Understanding*, vol. 133, pp. 102–109, 2015.

[6] G. E. Meyer, D. D. Jones, and A. K. Samal, "Plant species identification using elliptic fourier leaf shape analysis," *Computers and Electronics in Agriculture*, vol. 50, no. 2, pp. 121–134, 2006.

[7] Y. G. Naresh and H. S. Nagendraswamy, "Classification of medicinal plants: An approach using modified lbp with symbolic representation," *Neurocomputing*, vol. 173, pp. 1789–1797, 2016.

[8] J. Charters, Z. Wang, Z. Chi, A. C. Tsoi, and D. D. Feng, "Eagle: A novel descriptor for identifying plant species using leaf lamina vascular features," in *IEEE International Conference on Multimedia and Expo Workshops*, 2014, pp. 1–6.

[9] Z. Liu, L. Zhu, X. P. Zhang, X. Zhou, L. Shang, Z. K. Huang, and Y. Gan, *Hybrid Deep Learning for Plant Leaves Classification*. Springer International Publishing, 2015.

[10] S. H. Lee, C. S. Chan, P. Wilkin, and P. Remagnino, "Deepplant: Plant identification with convolutional neural networks," in *2015 IEEE International Conference on Image Processing*, 2015, pp. 452–456.

[11] S. H. Lee, C. S. Chan, S. J. Mayo, and P. Remagnino, "How deep learning extracts and learns leaf features for plant classification," *Pattern Recognition*, vol. 71, pp. 1–13, 2017.

[12] G. L. Grinblat, L. C. Uzal, M. G. Larese, and P. M. Granitto, "Deep learning for plant identification using vein morphological patterns," *Computers and Electronics in Agriculture*, vol. 127, pp. 418–424, 2016.

[13] S. Mouine, I. Yahiaoui, and A. Verroust-Blondet, "Advanced shape context for plant species identification using leaf image retrieval," in *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, 2012, pp. 1–8.

[14] X. Y. Xiao, R. Hu, S. W. Zhang, and X. F. Wang, "Hog-based approach for leaf classification," in *Advanced Intelligent Computing Theories and Applications and International Conference on Intelligent Computing*, 2010, pp. 149–155.

[15] J. S. Cope and P. Remagnino, *Classifying Plant Leaves from Their Margins Using Dynamic Time Warping*. Springer Berlin Heidelberg, 2012.

[16] Z. Tang, Y. Su, J. E. Meng, J. Zhou, J. Zhou, and J. Zhou, "A local binary pattern based texture descriptors for classification of tea leaves," *Neurocomputing*, vol. 168, no. C, pp. 1011–1023, 2015.

[17] J. S. Cope, P. Remagnino, S. Barman, and P. Wilkin, "Plant texture classification using gabor co-occurrences," in *International Conference on Advances in Visual Computing*, 2010, pp. 669–677.

[18] R. M. Craviotto, M. R. Arango, C. Gallo, and P. M. Granitto, "Automatic classification of legumes using leaf vein image features," *Pattern Recognition*, vol. 47, no. 1, pp. 158–168, 2014.

[19] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[20] C. Du and S. Gao, "Image segmentation-based multi-focus image fusion through multi-scale convolutional neural network," *IEEE Access*, vol. 5, no. 99, pp. 15 750–15 761, 2017.

[21] R. Rasti, H. Rabbani, A. Mehridehnavi, and F. Hajizadeh, "Macular oct classification using a multi-scale convolutional neural network ensemble," *IEEE Transactions on Medical Imaging*, 2017.

[22] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448–456.

[23] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *International Conference on International Conference on Machine Learning*, 2010, pp. 807–814.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 2, p. 2012, 2013.

[25] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 689–692.

[26] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *Workshop on Performance Evaluation of Tracking and surveillance*, 2007.

[27] J. Zhu, H. Zeng, S. Liao, Z. Lei, C. Cai, and L. Zheng, "Deep hybrid similarity learning for person re-identification," *arXiv preprint arXiv:1702.04858*, 2017.

[28] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1794–1801.