

## Plant species identification from leaf patterns using histogram of oriented gradients feature space and convolution neural networks

Truong Quoc Bao, Nguyen Thanh Tan Kiet, Truong Quoc Dinh & Huynh Xuan Hiep

**To cite this article:** Truong Quoc Bao, Nguyen Thanh Tan Kiet, Truong Quoc Dinh & Huynh Xuan Hiep (2020) Plant species identification from leaf patterns using histogram of oriented gradients feature space and convolution neural networks, Journal of Information and Telecommunication, 4:2, 140-150, DOI: [10.1080/24751839.2019.1666625](https://doi.org/10.1080/24751839.2019.1666625)

**To link to this article:** <https://doi.org/10.1080/24751839.2019.1666625>



© 2019 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group



Published online: 22 Sep 2019.



Submit your article to this journal [↗](#)



Article views: 4919



View related articles [↗](#)



View Crossmark data [↗](#)



Citing articles: 14 View citing articles [↗](#)

# Plant species identification from leaf patterns using histogram of oriented gradients feature space and convolution neural networks

Truong Quoc Bao<sup>a</sup>, Nguyen Thanh Tan Kiet<sup>b</sup>, Truong Quoc Dinh<sup>b</sup> and Huynh Xuan Hiep<sup>b</sup>

<sup>a</sup>College of Engineering Technology, Can Tho University, Can Tho, Vietnam; <sup>b</sup>College of Information & Communications Technology, Can Tho University, Can Tho, Vietnam

## ABSTRACT

The determination of plant species from field observation requires substantial botanical expertise, which puts it beyond the reach of most nature enthusiasts. Traditional plant species identification is almost impossible for the general public and challenging even for professionals that deal with botanical problems daily, such as, conservationists, farmers, foresters, and landscape architects. Even for botanists themselves, species identification is often a difficult task. In this research, we proposed using two methods for the problem of plant species identification from leaf patterns. Firstly, we use a traditional recognition shallow architecture with extracted features histogram of oriented gradients (HOG) vector, then those features used to classifying by SVM algorithm. Secondly, we apply a deep convolutional neural network (CNN) for recognition purpose. We experimented on leaves data set in the Flavia leaf data set and the Swedish leaf data set. We want to compare a tradition method and a method consider as current state-of-the-art.

## ARTICLE HISTORY

Received 30 June 2018

Accepted 9 September 2019



## KEYWORDS

Histogram of oriented gradients; deep learning; convolutional neural networks; leaf classification

## 1. Introduction

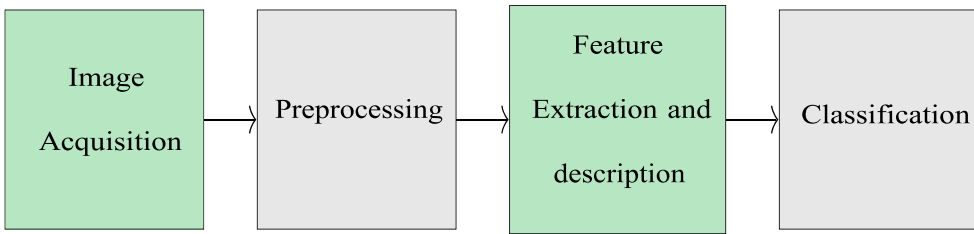
Image-based methods are considered a promising approach for species identification. A user can take a picture of a plant in the field with the build-in camera of a mobile device and analyze it with an installed recognition application to identify the species or at least to receive a list of possible species if a single match is impossible. By using a computer-aided plant identification system also non-professionals can take part in this process. An image classification process can generally be divided into the steps in Figure 1.

Image acquisition: The purpose of this step is to obtain the image of a whole plant or its organs so that analysis towards classification can be performed. The aim of image preprocessing is enhancing image data so that undesired distortions are suppressed and image

**CONTACT** Truong Quoc Bao  tqbao@ctu.edu.vn  College of Engineering Technology, Can Tho University, Can Tho, Vietnam

© 2019 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



**Figure 1.** Generic steps of an image-based plant classification process.

features that are relevant for further processing are emphasized. The preprocessing sub-process receives an image as input and generates a modified image as output, suitable for the next step, the feature extraction. Preprocessing typically includes operations like image denoising, image content enhancement, and segmentation. These can be applied in parallel or individually, and they may be performed several times until the quality of the image is satisfactory. Feature extraction and description: Feature extraction refers to taking measurements, geometric or otherwise, of possibly segmented, meaningful regions in the image. Features are described by a set of numbers that characterize some property of the plant or the plant's organs captured in the images (aka descriptors). Classification: In the classification step, all extracted features are concatenated into a feature vector, which is then being classified.

Image tradition classification is usually based on features engineerings such as SIFT, HOG, SURF, combined with a learning algorithm in these features engineering spaces such as SVM, Neuron, and KNN. The efficiency of all approaches that depend heavily on predefined features. Image features engineering itself is a complex process that requires changes and recalculation for each problem or associated data set.

With the development of neural networks, neural network architecture has been used as an effective solution to extract high-level features from data. Deep Convolutional Neural Network architectures can accurately portray highly abstract properties with condensed data while preserving the most up-to-date characteristics of raw data. This is beneficial for classification or prediction. In recent times, CNN has emerged as an effective framework for describing features and identities in image processing. CNN can learn basic filters automatically and combine them hierarchically to describe underlying concepts to identify patterns. CNN does not need to compute features engineering that is time and effort consuming. The generalization of the method makes it a practical and scalable approach to the various application problems of classification and recognition.

Another substantially studied local feature approach is the histogram of oriented gradients (HOG) descriptor (Pham, Le, Grard, & Nguyen, 2013). The HOG descriptor, introduced by (Lowe, 2004) is similar to SIFT, except that it uses an overlapping local contrast normalization across neighbouring cells grouped into a block. Since HOG computes histograms of all image cells and there are even overlap cells between neighbour blocks, it contains much redundant information making dimensionality reduction inevitably for further extraction of discriminant features. Therefore, the main focus of studies using HOG lies in dimensionality reduction methods.

SIFT has been proposed and studied for leaf analysis by Chathura and Withanage (2015); Hsiao, Kang, Chang, and Lin (2014); Lavania and Matey (2014). A challenge

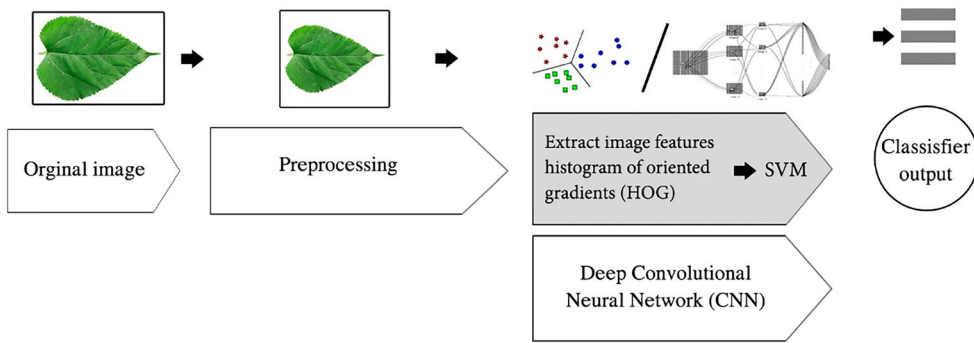
that arises for object classification rather than image comparison is the creation of a codebook with trained generic keypoints. The classification framework by Chathura and Withanage (2015) combines SIFT with the Bag of Words (BoW) model. The BoW model is used to reduce the high dimensionality of the data space. Hsiao et al. (2014) used SIFT in combination with sparse representation (aka sparse coding) and compared their results to the BoW approach. The authors argue that in contrast to the BoW approach, their sparse coding approach has a major advantage as no re-training of the classifiers for newly added leaf image classes is necessary. In Lavania and Matey (2014), SIFT is used to detect corners for classification. Nguyen, Le, and Pham (2013) studied speeded up robust features (SURF) for leaf classification, which was first introduced by Bay, Tuytelaars, and Van (2006). The SURF algorithm follows the same principles and procedure as SIFT. However, details per step are different. The standard version of SURF is several times faster than SIFT and claimed by its authors to be more robust against image transformations than SIFT (Bay et al., 2006). To reduce the dimensionality of extracted features, Nguyen et al. (2013) apply the previously mentioned BoW model and compared their results with those of Pham et al. (2013). SURF was found to provide better classification results than HOG Pham et al. (2013).

Ren, Wang, and Zhao (2012) propose a method for building leaf image descriptors by using multi-scale local binary patterns (LBP). Initially, a multi-scale pyramid is employed to improve leaf data utilization and each training image is divided into several overlapping blocks to extract LBP histograms in each scale. Then, the dimension of LBP features is reduced by a PCA. The authors found that the extracted multi-scale overlapped block LBP descriptor can provide a compact and discriminative leaf representation.

In Oide and Ninomiya (2000) used neural networks to classify soybean leaves using a Hopfield network and a simple perceptron. In Krizhevsky, Sutskever, and Hinton (2012) have used Deep Convolutional Neural Networks for ImageNet and their research results have created a new rush for deep learning.

Several publications have suggested the use of CNN in leaf classification in recent years. Jassmann, Tashakkori, and Parry (2015) develop an application for classifying plants, based on leaf images. The system uses a CNN in a mobile application for mobile phones to categorize the nature of the leaf, trained with the ImageCLEF data set. The proposed architecture consists of a convoluted layer, followed by a composite layer and two fully connected layers applied to the  $60 \times 80$ -pixel input image. Wu, Shang, Huang, Wang, and Zhang (2016). Proposed a simplified version of AlexNet for leaf recognition. They have used parametric linear units (PReLU) instead of ReLU. Plant disease identification includes the processing of leaf recognition. Sladojevic, Arsenovic, Anderla, Culibrk, and Stefanovic (2016) have been interested in a new method for developing a disease-identification model based on leaf classification of images, using CNN. The developmental model was able to recognize 13 healthy plant leaf diseases, with the ability to discriminate leaves from the surrounding environment.

The main objectives of this paper are (1) Description of features histogram of oriented gradients (HOG) and plant classification from leaf patterns by SVM; (2) Description of the CNN model plant classification from leaf patterns and (3) Comparison of results of two methods and discussion.



**Figure 2.** Our scheme implementation. In this work, the grayed stages were compared result with a deep convolutional network.

## 2. Methods

In [Figure 2](#). Our scheme implementation, we use two data sets leaf public are Swedish and Flavia leaf data set.

### 2.1. Plant species identification using the histogram of oriented gradients (HOG) feature space

Histogram of Oriented Gradient (HOG) was first proposed by Dalal and Triggs (2005) for human body detection but it is now one of the successful and popular used descriptors in computer vision and pattern recognition. HOG counts occurrences of gradient orientation in part of an image hence it is an appearance descriptor. HOG divides the input image into small square cells (here we used  $cell\ size = 8 \times 8$ ) and then computes the histogram of gradient directions or edge directions based on the central differences. For improved accuracy, the local histograms have been normalized based on the contrast and this is the reason that HOG is stable on illumination variation. It is a fast descriptor in compare to the SIFT and LBP due to the simple computations.

#### 2.1.1. Feature extraction

In this stage, the features of the leaf that are crucial for classifying them at the recognition stage are extracted. This is an important stage as its effective functioning improves the recognition rate and reduces the misclassification. HOG based feature extraction scheme for recognizing leaf image is proposed in this work. Every leaf image resized  $50 \times 50$  and  $128 \times 128$  pixels is used to extract the HOG feature. The proposed HOG feature is extracted by considering  $cell\ size\ 8 \times 8$  dimension and  $cell\_per\_block = 2 \times 2$  of the input leaf image.

#### 2.1.2. Support vector machine (SVM)

In machine learning, support vector machines (SVM) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked for

belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a nonprobabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible.

## 2.2. Convolutional neural network

### 2.2.1. Convolutional neural network (CNN)

Convolutional Neural Network is an effective identification method, developed in recent years, that caused widespread attention. Now, CNN has become one of the most efficient methods in the field of pattern classification and recently, has been used more widely in the field of image processing (Krizhevsky et al., 2012; LeCun, Bengio, & Hinton, 2015), and it can reach a better performance than traditional methods (Chatfield, Lempitsky, Vedaldi, & Zisserman, 2011). through wide verification. CNN consists of one or more pairs of convolutional and max pooling layers. A convolutional layer applies a set of filters that process small local parts of the input where these filters are replicated along with the whole input space. A max-pooling layer generates a lower resolution version of the convolutional layer activations by taking the maximum filter activation from different positions within a specified window. This adds translation invariance and tolerance to minor differences of positions of objects parts. Higher layers use more broad filters that work on lower resolution inputs to process more complex parts of the input. Top fully connected layers finally combine inputs from all positions to do the classification of the overall inputs. This hierarchical organization generates good results in image processing tasks.

### 2.2.2. Convolutional neural network structure

- *Convolution layer*, the convolution operation extracts different features of the input. The first convolution layer extracts low-level features like edges, lines, and corners. Higher-level layers extract higher-level features.
- *Non-linear layers*, Neural networks in general and CNNs, in particular, rely on a non-linear ‘trigger’ function to signal distinct recognition of likely features on each hidden layer. CNNs may use a variety of specific functions such as rectified linear units (ReLU) and continuous trigger (non-linear) functions to efficiently implement this non-linear triggering. A ReLU implements the function  $y = \max(x, 0)$ , so the input and output sizes of this layer are the same.
- The pooling/subsampling layer reduces the resolution of the features. It makes the features robust against noise and distortion.
- Fully connected layers are often used as the final layers of a CNN. These layers mathematically sum a weighting of the previous layer of features, indicating the precise mix of ‘ingredients’ to determine a specific target output result. In case of a fully connected layer, all the elements of all the features of the previous layer get used in the calculation of each element of each output feature.

Training is performed using a ‘labeled’ data set of inputs in a wide assortment of representative input patterns that are tagged with their intended output response. Training uses

**Table 1.** The proposed CNN architecture.

Attribute Type	L1		L2		L3	L4	L5
	Conv	MaxPool	Conv	MaxPool	Conv	Conv	
Filter size	$8 \times 8$	$7 \times 7$	$5 \times 5$	$6 \times 6$	$2 \times 2$	$3 \times 3$	Softmax
Stride	2	2	2	2	1	1	
Number of filter	100	100	250	250	100	100	
Output size	$61 \times 61$	$28 \times 28$	$13 \times 13$	$4 \times 4$	$3 \times 3$	$1 \times 1$	

general-purpose methods to iteratively determine the weights for intermediate and final feature neurons.

### 2.2.3. Deep learning proposal

CNN architectures vary with type of images and especially when input image sizes are different. In this paper, the size of input images is considered to be  $128 \times 128$  pixels. The proposed architecture is described in Table 1. After each Conv layer, a ReLU activation function is used and for each pooling layer, MaxPooling approach is applied. The fully connected layers are defined as convolutional layers with the filter size of  $1 \times 1$  as it is conventional in MatConvnet (Vedaldi & Lenc, 2015). The final layer has  $n$  units corresponding to  $n$  and category of leaf datasets. After all layers, a SoftMax loss is placed.

The CNN model is shown in Table 1. There are 5 layers that include: [Conv – ReLu – Max pool] → [Conv – ReLu – Max pool] → [Conv – ReLu] → [Conv → FC] → Softmax.

## 3. Experiment and results

### 3.1. Experiment data sets

In order to test the performance of the classification system, we selected two standard sets:

- *Swedish leaf data set*: The Swedish leaf data set has been captured as part of a joined leaf classification project between the Linköping University and the Swedish Museum of Natural History (Soderkvist, 2001). The data set contains images of isolated leaf scans on a plain background of 15 Swedish tree species, with 75 leaves per species (1125 images in total). This data set is considered very challenging due to its high inter-species similarity.
- *Flavia data set*: This data set contains 1907 leaf images of 32 different species and 50–77 images per species. Those leaves were sampled on the campus of the Nanjing University and the Sun Yat-Sen arboretum, Nanking, China. Most of them are common plants of the Yangtze Delta, China (Wu et al., 2007). The leaf images were acquired by scanners or digital cameras on a plain background. The isolated leaf images contain blades only, without petioles.

**Table 2.** Partition data.

Data set	Number layers	Number Samples	Train		Val 20%	Test 20%
			60%	Augmentation data		
Swedish	15	1125	900	3600	225	225
Flavia	32	1907	1145	4580	381	381
Flavia + Swedish	47	5825	2045	8180	606	606

### 3.2. Augmentation data

For reason limitation of over-model phenomena of the model due to insufficient data. Augmentation data is an effective solution. We increased the number of training images by creating three copies of each image after reflection and rotation. Thus, each original images create three image augmentation. Data partitioned for the experiment is shown in [Table 2](#).

In this study, the input data of the model is scanned or taken from leaves of the trees, then model the training. The experimental process is as follows:

- *Standardized image size*: the image resize to  $128 \times 128$ px to match the input of the network. To resize images to the desired size of  $128 \times 128$  pixels, first images are resized such that the larger dimension of them is equal to 128, then the smaller one is padded with pixels having the value of 0.
- *Image Partition*: Each category is partitioned as shown in [Table 2](#) for experimental purposes.
- *Initialization parameter*: Learning rate: set to 0.00007 (greater than fast convergence network but not very good error rate, smaller than slow convergence network); Weight-Decay constant (anti-overfitting) = 0.0005; and Momentum constant = 0.9.

The above constants are chosen based on experimental results for the proposed model by trial And error Method. Training time depends on computer resources with GPU or CPU, Matlab software and Matconvnet tool.

### 3.3. Experiment results

- With HOG-SVM model, we use 80% image to training, 20% image test. We use training images are  $50 \times 50$ ,  $128 \times 128$  and cell size  $8 \times 8$  to extract HOG feature. This paper employed a multiclass SVM classifier as a classification tool of HOG feature space developed for a complete dataset of leaf. The HOG feature of dimension  $1 \times 900$  or  $1 \times 8100$  for each individual leaf image feature space. Hence the final dimension of the feature space developed for the proposed SVM based classification purpose is of  $N \times 900$  or  $N \times 8100$  (N is numbers of images in data set).

The experiment results are aggregated into each test data set and detail in [Table 3](#).

- With the CNN model, datasets are separated to 80%, 20% and 20% as training, validation and test sets respectively, then augmentation training. Input data is  $128 \times 128$ , the layer L4 (in [Table 1](#)) is feature vector  $1 \times 100$  for classifying by Softmax function.

**Table 3.** Experiment results of HOG-SVM model.

Image size Datasets	Accuracy test set	
	$50 \times 50$	$128 \times 128$
Swedish (15 layers)	<b>98</b>	95
Flavia (32 layers)	91	<b>92</b>
Swedish + Flavia	88.7	94.17



**Table 4.** Experiment results of CNN model.

CNN Model		
Rate Train/Val/Test 60-20-20	Augmentation data	Accuracy test set CNN
Swedish (15 layers)	No	96.15
	Yes	<b>98.22</b>
Flavia (32 layers)	No	93.8
	Yes	<b>95.5</b>
Swedish + Flavia	No	94.17
	Yes	95.6

**Table 5.** Experiment results the best of two models.

Datasets	HOG-SVM	CNN
Swedish (15 layers)	98	98.22
Flavia (32 layers)	92	95.5
Swedish + Flavia	88.7	95.6

The experiment results are aggregated into each test data set and detail in [Table 4](#). From the experiment results we in [Table 5](#).

#### 4. Conclusions

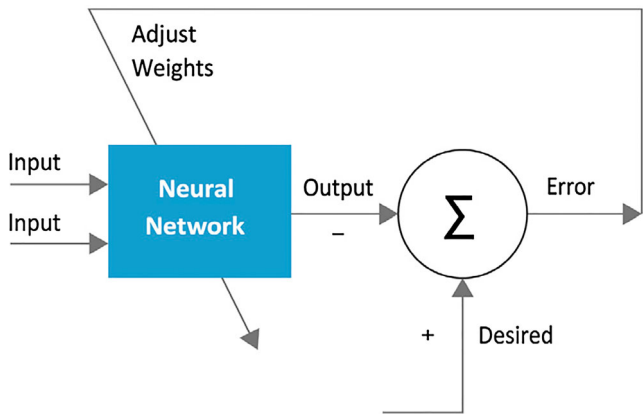
In this paper, we proposed two solutions for classification leaf by using image processing combined with a shallow architecture and a deep architecture to classify the image leaf. With the recognition of shallow architecture, we must find parameters by hand or hand-designed feature extraction. We use HOG features for recognition the result in [Table 3](#). With recognition, deep architecture combine with the effect of horizontal reflection and rotation augmentation of datasets further improves the results in [Table 4](#). The accuracy of shallow architecture depends on choosing the input size of the image, as well as the feature vector length. Deep architecture with the same input image excitation, has the feature vector is shorter and they are found by model itself for greater accuracy. The results showed in [Table 5](#) that the proposed architecture for CNN-based leaf classification closely competes with the latest extensive approaches on devising leaf features and classifiers.

**Table 6.** Comparison of results of leaves recognition methods on the Swedish data set.

Descriptor	Feature	Classifier	Accuracy
Our method (CNN)	Shape + vein	CNN	<b>98.22</b>
Our method (HOG)	Shape	SVM	<b>98</b>
IDSC	Shape	SVM	93.73
IDSC	Shape	k-NN	94.13
TOA	Shape	k-NN	95.20
TSL	Shape	k-NN	95.73
TSLA	Shape	k-NN	96.53
LBP	Shape	SVM	96.67
I-IDSC	Shape	1-NN	97.07
MARCH	Shape	1-NN	97.33
DS-LBP	Shape + texture	Fuzzy k-NN	<b>99.25</b>

**Table 7.** Statistical results of leaves recognition methods on the Flavia data set.

Descriptor	Feature	Classifier	Accuracy
Our method (CNN)	Shape + vein		<b>95.5</b>
Our method (HOG)	Shape	SVM	<b>92</b>
HOG	Shape		84.70
SIFT	Shape		87.50
SMSD, <i>Avein/Aleaf</i>	Shape + vein	PNN	90.31
SMSD, FD, CM	Color + shape	k-NN, DT	91.30
SMSD	Shape	PNN	91.40
SMSD, CM	Shape + color	RF (k-NN, NB, SVM)	93.95
SMSD, <i>Avein/Aleaf</i>	Shape + vein	SVM (k-NN)	94.50
SIFT	Shape	SVM	<b>95.47</b>
SURF	Shape	SVM	<b>95.94</b>
SMSD, FD	Shape	BPNN	<b>96.00</b>
SMSD, CM, GLCM, <i>Avein/Aleaf</i>	Shape + color + texture + vein	SVM	<b>96.25</b>



**Figure 3.** Training of neural networks (Samer, Rishi, & Chris, 2015).

Comparisons of the results were compiled in Wäldchen and Mäder (2017) with the method of leaves recognition on the Swedish and Flavia data sets in the following Tables 6 and 7. Here, the accuracy is (Number of correct identities/Total leaf of test set) × 100%.

From experiment results with Swedish and Flavia data sets, we can confirm that the CNN-based neural network depth model, which we propose, works very well on classification problem of leaves based on the shape of veins. This result once again confirms the effectiveness and simplicity of the CNN depth geometry model for real-world problems with large data. The recognition process is done by simply building the model and determining the appropriate parameters. The effectiveness of the classification process, recognition is no longer too dependent on finding and identifying image features, a process that takes a lot of time and effort.

Looking forward, we will be collecting a larger, more diverse image to broaden the scope of our research. Besides, we some additional detection and identification methods are used to compare advantages and disadvantages to the proposed method. Next, we aim to install applications on mobile devices that are widely available Figure 3.

**Disclosure statement**

No potential conflict of interest was reported by the authors.

## Notes on contributors

**Truong Quoc Bao** is Working at Can Tho University, Can Tho, Vietnam.

**Nguyen Thanh Tan Kiet** is Working at Can Tho University, Can Tho, Vietnam.

**Truong Quoc Dinh** is Working at Can Tho University, Can Tho, Vietnam.

**Huynh Xuan Hiep** is Working at Can Tho University, Can Tho, Vietnam.

## References

- Bay, H., Tuytelaars, T., & Van, G. L. (2006). *Surf: Speeded up robust features*. European conference on computer vision, Springer, Berlin, 404–417.
- Chatfield, K., Lempitsky, V. S., Vedaldi, A., & Zisserman, A. (2011). *The devil is in the details: An evaluation of recent feature encoding methods*. BMVC, 2.
- Chathura, P. H., & Withanage, D. (2015). *Computer assisted plant identification system for android*. 2015 Moratuwa engineering research conference (MERCon), 148–153.
- Dalal, N., & Triggs, B. (2005). *Histograms of oriented gradients for human detection*. CVPR, 200.
- Hsiao, J. K., Kang, L. W., Chang, C. L., & Lin, C. Y. (2014). *Comparative study of leaf image recognition with a novel learning-based approach*. 2014 Science and information conference (SAI), 389–393.
- Jassmann, T. J., Tashakkori, R., & Parry, R. M. (2015). *Leaf classification utilizing a convolutional neural network*. Southeastcon 2015, IEEE, 1–3.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *Imagenet classification with deep convolutional neural networks*. Advances in neural information processing systems, 1097–1105.
- Lavanaia, S., & Matey, P. S. (2014). *Leaf recognition using contour based edge detection and sift algorithm*. 2014 IEEE international conference on computational intelligence and computing research (ICCIC), 1–4.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521, 436–444.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110.
- Nguyen, Q. K., Le T. L., & Pham, N. H. (2013). *Leaf based plant identification system for android using surf features in combination with bag of words model and supervised learning*. 2013 International conference on advanced technologies for communications (ATC), 404–407.
- Oide, M., & Ninomiya, S. (2000). Discrimination of soybean laeflet shape by neural networks with image input. *Computers and Electronics in Agriculture*, 29(1–2), 59–72.
- Pham, N. H., Le, T. L., Grard, P., & Nguyen, V. N. (2013). *Computer aided plant identification system*. 2013 International conference on computing, management and telecommunications (ComManTel), 134–139.
- Ren, X. M., Wang, X. F., & Zhao, Y. (2012). *An efficient multi-scale overlapped block LBP approach for leaf image recognition*. Proceedings of the 8th international conference on intelligent computing theories and applications (ICIC'12), Springer, Berlin, 37–243.
- Samer, H., Rishi, K., & Chris, R. (2015). *Using convolutional neural networks for image recognition*. IP Group, Cadence. Retrieved from [https://ip.cadence.com/uploads/901/cnn\\_wp-pdf](https://ip.cadence.com/uploads/901/cnn_wp-pdf)
- Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Depth neural networks based recognition of plant diseases by leaf image classification. *Computational Intelligence and Neuroscience*, 11, 1–11.
- Soderkvist, O. J. O. (2001). *Computer vision classification of leaves from Swedish trees*. Linkoping: Linkoping University.
- Vedaldi, A., & Lenc, K. (2015). *Matconvnet: Convolutional neural networks for matlab*. Proceedings of the 23rd ACM international conference on Multimedia, ACM, 689–692.
- Wäldchen, J., & Mäder, P. (2017). Plant species identification using computer vision techniques: A systematic literature review, in archives of computational methods in engineering. ISSN: 1134-3060 (Print) 1886-1784 (Online). Retrieved from <https://www.researchgate.net/publication/312147459>

- Wu, S. G., Bao, F. S., Xu, E. Y., Wang, Y. X., Chang, Y. F., & Xiang, Q. L. (2007). *A leaf recognition algorithm for plant classification using probabilistic neural network*. 2007 IEEE international symposium on signal processing and information technology, IEEE, 11–16.
- Wu, Y. H., Shang, L., Huang, Z. K., Wang, G., & Zhang, X. P. (2016). Convolutional neural network application on leaf classification. In D. S. Huang, V. Bevilacqua, & P. Premaratne (Eds.), *Intelligent computing theories and application*. ICIC 2016, Lecture Notes in Computer Science (Vol. 9771, pp. 12–17). Springer.