A Hybrid Convolutional Neural Network for Complex Leaves Identification

Daniel Ayala Niño[1(✉)], Jair Cervantes Canales[1], Farid García Lamont[1], Joel Ayala de la Vega[1], and Guillermo Calderón Zavala[2]

[1] UAEMEX (Universidad Autónoma del Estado de México), Jardín Zumpango s/n, Fraccionamiento El Tejocote, 56259 Texcoco, CP, Mexico
{danielayalanio, joelayala2001}@yahoo.com.mx, {jcervantesc, fgarcial}@uaemex.mx

[2] ColPos (Colegio de Postgraduados), Km. 36.5, Montecillo, 56230 Texcoco, CP, Mexico
cazagu@colpos.mx

Abstract. The classification of leaves has gained popularity through the years, and a great variety of algorithms has been created to target these tasks, among those is the Deep Learning approach, which simplicity of learning from raw imputed data makes this task easy to target. However, not all methods are into the complex leaves classification task. In this work we propose a different approach in the way the leaf's pictures are used to train the models, this is done by using the front and back face of a leaf as one element of the dataset. These pairs will be inputted into two shared convolutional layers, making the models to learn from a complete leaf. The results obtained in this work overpassed the accuracy obtained in related works. For this, we created a new complex leaves dataset, that consists of 6 different kinds of peach varieties, the dataset is available in this link (https://drive.google.com/drive/folders/1rWCr9DrknoK0HKFhNRavCVgZ5UKjU3hi).

Keywords: Complex leaves identification · Convolutional Neural Networks · Computer vision

1 Introduction

Plants identification has always played an important role in different disciplines, such as medicine in the development of new drugs [1, 2, 15], botany [3], agriculture [4, 9, 32]. This last one is important for food production. It is known that by 2050, global food production will increase by 60%. Thus, knowing which variety of plant is better under certain conditions, will be a clue point to improve and increase the agricultural production [5].

Since the necessity of identifying plants has increased for different reasons exposed above, the techniques for classifying them have also improved. Botanists use different leaf's characteristics (features like texture, color, morphological, venation, among others) to identify the plants, they spend too much time studying a family of plant and its varieties, this is because most of these features are hand-crafted [3].

D.-S. Huang et al. (Eds.): ICIC 2020, LNCS 12463, pp. 291–301, 2020.
https://doi.org/10.1007/978-3-030-60799-9_25

On the other hand, new technologies have targeted this aim, like machine learning and computer vision. This has been done by applying different techniques, such as extracting features from the leaves using different algorithms to get the more descriptors as possible from it [6–8, 10] and then using these features to feed a classifier. Nowadays with the increasing popularity of Deep Learning and Convolutional Neural Networks (CNN) due to its effectiveness with image classification, have demonstrated that it is not necessary to hand-engineer the features before its classification [11]. These kinds of models train using big datasets with millions of elements, and it is common knowledge that the bigger the dataset is the better the performance of the model will be [12].

Related papers have worked on big datasets classifying different kinds of plants [13, 14], but there are not too much-related papers focused on the classification of complex leaves using Deep Learning. In a previous work [7], we demonstrated that it is possible to classify complex leaves, using a Genetic Algorithm to select the features that better describe a plant, then this set of features were used to train a classifier, obtaining high rates of accuracy. Our purpose in this research is to create a model that could classify different varieties of peach and it will be able to learn from a complete leaf, not just from its back face or front face, by separate; thing that is not targeted yet in the state of the art. To accomplish this, we have created a new dataset in which these two leaf's faces are considered as one element of the dataset, and inputting these pairs into our new hybrid convolutional neural network models, which first convolutional layers are shared, this approach has overpassed our previous work's accuracy [7]. The CNN models used in this research were based on the models exposed in [16] and [24].

The paper is organized as follows. In Sect. 2 we give a brief review of the state of the art related to complex leaves classification. Then in Sect. 3, we describe the method used to classify the Peach leaves. We show the results obtained with the model described in Sect. 4. And finally, our conclusions related to the results obtained and future work is described in Sect. 5.

2 State of the Art

There are two main approaches that most of the literature used to classify plants by using leaves and computer vision. These basically are: Using hand-crafted features to feed a classifier [17], then we have the deep learning approach using CNN to automatically extract the main information from pictures and then classify them, in some cases it is necessary to pre-process the pictures to feed a CNN, something unusual, because in most of the cases CNN's are fed with raw pictures. We will give a glimpse of the state of the art related to the methods described above.

The hand-crafted extraction method has been broadly used for plant classification since every plant has unique physical characteristics that distinguish them from others [30, 31]. Like the morphological characteristics, which could be extracted using Fourier descriptors [18], where Aakif and Khan [13] used it together with a new feature called shape-defining, this final vector was used to feed an artificial neural network (ANN), obtaining an accuracy of 96%. Another way to do it is by the texture of the leaf, like

Backes et al. [19], who uses fractal theory to classify ten types of leaves achieving a precision of 90%. Caglayan et al. extracted shape and color features to classify them using algorithms like K-Nearest Neighbor, Support Vector Machines, Naïve Bayes and Random Forest, obtaining a 96% accuracy working along with the Flavia dataset [10]. Elnemr, proposed a system that consists of five steps, which are preprocessing the image, then the image segmentation is applied, after this, the features are extracted like curvelet transform descriptors, local binary pattern and gray level co-occurrence matrix texture, then the feature selection is carried using the Neighborhood Component Feature Selection (NCFS) technique, selecting the highly discriminative features, obtaining an accuracy of 98% [20]. The approaches described above are time-consuming since the process of pre-processing the image, extracting and selecting the features must be done with each element of the dataset, and for new leaves the same process must be carried. Most of the time all features extracted are used, but not all of them contribute significantly, so that means that other algorithms must be held to select the best features, this means more time spending selecting features [6, 7]. In contrast with the Deep Learning approach where all this is done automatically.

CNN's has become the most popular algorithm for computer vision tasks, due to its excellent performance in different fields like image classification, image recognition and image segmentation. And it has showed a great performance in the plant classification task. Like Guillermo et al. [11], where three different legume species were classified (white bean, red bean and soybean), here they highlight different levels of vein details, using vein segmentation, using two kinds of setups, reaching a 96.9% accuracy, in comparison with a similar approach, done by Larese et al. [21] where 95.1% accuracy was reached, here instead of CNN's some vein measures were extracted to feed a SVM, PDA or RF. Lee et al. [22], used CNN's to classify the plants from the MalayaKew (MK) Leaf Dataset, that consist of images of leaves from 44 species classes, and reusing a pre-trained CNN ILSVRC2012. Here they used deconvolutional networks (DN) to find out which part of the leaves the CNN is focusing on, obtaining a 99.5% accuracy. CNN's are also used to create new features like the work of Ramos et al. [23], where a convolutional autoencoder (CAE) is used as feature extractor from the leaf's pictures, and this final vector is used to feed a SVM, obtaining an accuracy of 94.74%. Bing Wang and Dian Wang, proposed a method of Few-Shot learning applying the Siamese Network [24], this to create a metric space for leaf classification, where similar samples are close to each other and dissimilar samples are far away, this is due to the small datasets that were used (Flavia, Swedish and Leafsnap) obtaining a 95.32% accuracy. Lee et al. [16], also used DN to find out what a CNN is learning from leaves, and proposed hybrid-CNN's models, these hybrid models have two inputs, one input is the leaf picture, and the second input is a patch from the same leaf, this with the aim to focus on the leaf's venation. Three different hybrid models were described on this work, obtaining a 96.3% accuracy using the hybrid model Early fusion (conv-sum).

Something to highlight about the state of the art described above is that the models were not focused to classify complex leaves, which are leaves that belong to the same species, but they are from different varieties. The leaves used in the state of the art did not meet this requirement, being this our target, which consists of classifying 6 varieties of peach. Also, most of the CNN's in the literature don't pay attention to which side of the leaf they are learning from, both sides of the leaves are used as different samples of the dataset, in our approach we want the model to learn from a complete leaf, and that is its front and back face, and this is achieved by our hybrid CNN models.

3 CNN Proposal

Deep learning is a machine learning algorithm that consists of multiple layers, letting the representations of multiple data abstraction. Its main purpose is to extrapolate new features from raw input representations, this is done without telling the model which features to use and how to extract them. Being this the importance of using CNN, because of its ability to classify and extract the characteristics from images without needing to tell it how to do it, and the convolutional layer helps us to reach this purpose. After each convolution layer comes a pooling layer, this kernel is creating subsamples from the inputted image, in order to reduce the computational cost and space. At the end of these convolution and pooling layers, the full connected layer follows, this last layer works as a hidden layer of an ANN [7]. In this section, the architecture used for this work is discussed.

3.1 Models

The architectures exposed here follows the same principle proposed by Lee et al. [16] and Bing Wang [24], in this work, we decided that it is important for the model to train with a dataset divided into two components, the front face of the leaf and the backside of the same, and in contrast of [24], we use this principle of the Siamese network to create a function that learns the similarities among the front side and the backside of the same leaf, that is why the comparison of leaves belonging to a different variety is not held. Figure 1 shows a representation of the models applied, as it can be seen, we have two different models that use an early convolution fusion, applying a conv-sum [16]. The input (a) of the models consists of the front face of the leaf and the input (b) is the same leaf back face. Two different datasets were created to train the models, one model was trained with pictures of 224 × 224 pixels (Fig. 1a), and the second model with a higher resolution, 416 × 416 pixels (Fig. 1b). More details about the dataset will be drawn out in Sect. 4.

The early convolution fusion consists of two streams, these streams are sharing weights, that means its updating is not independent from each other. After the first 3 conv-layers, the resulting convolutional tensor from each stream will be added up, the result will be convolved across two more units, this for the first model, for the second model, the result will be convolved across 4 units, this is because the picture is bigger and it is possible to extract more information from it. More details about the models are described on Table 1 and 2. Dropout where used in the unit 4 and 5 of model (224 × 224), for unit 4 a dropout of *0.05* was used and for unit 5 a dropout *0.1* was used.

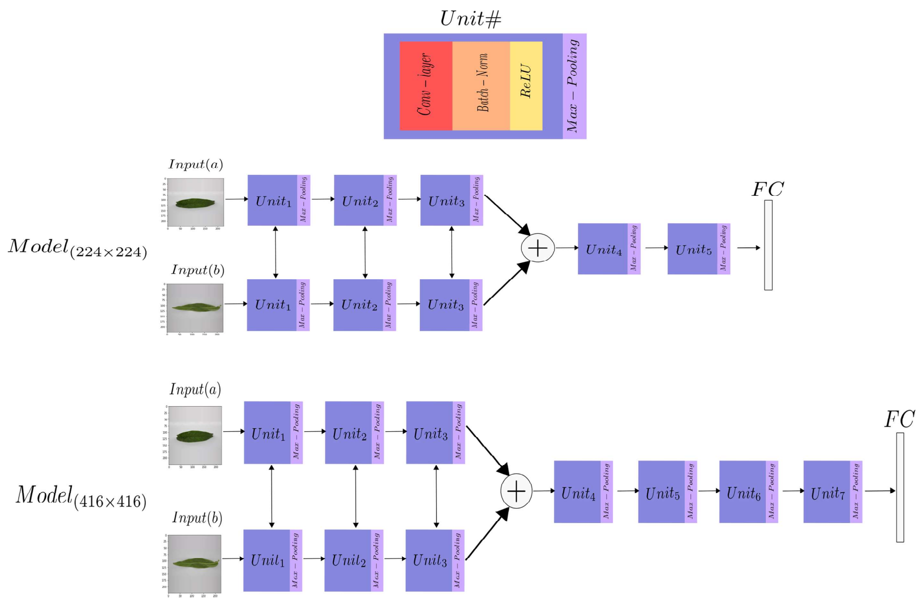

Fig. 1. a) Model (224 × 224). b) Model (224 × 224).

Table 1. Model (224 × 224)

Unit1 ConvL1	MaxPool 1	Unit2 ConvL2	MaxPool2	Unit 3 ConvL3	MaxPool3	Unit4 ConvL4	Max-Pool4	Unit5 ConvL5	Max-Pool5	FC
5X5X18	3X3	5X5X30	3X3	5X5X60	3X3	5X5X100	2X2	5X5X90	2X2	720

Table 2. Model (416 × 416)

Unit 1 Conv L1	Max Pool 1	Unit2 ConvL 2	Max Pool 2	Unit 3 ConvL 3	Max Pool 3	Unit4 ConvL4	Max-Pool 4	Unit5 ConvL 5	Max-Pool 5	Unit6 ConvL 6	Max-Pool 6	Unit6 ConvL7	Max-Pool 7	FC
5X5 X18	3X3	5X5X3 0	3X3	5X5X6 0	3X3	5X5X10 0	2X2	5X5X9 0	2X2	5X5X9 0	2X2	5X5X18 0	2X2	720

He initialization [25] were used for all the conv-layers, this is given that ReLU activation is applied [26]. For the FC layer, the Xavier initialization [27] were used, given that Softmax activation is applied on the FC. The He initializer is similar to the Xavier initializer, except that Xavier initialization uses a scaling factor for the weights, as it can be observed on Eq. 1 (He) and Eq. 2 (Xavier).

$$Var(w_i) = sqrt(\frac{2}{fan_{in}}) \tag{1}$$

$$Var(w_i) = sqrt(\frac{1}{fan_{in}}) \tag{2}$$

Where fan_{in} is the number of input units in the weight tensor.

Given that this is a multi-class classification objective, the objective function used was the Categorical Cross-Entropy, also called as Softmax loss. This is a Softmax activation, Eq. 3, plus a cross-entropy loss, Eq. 4, where s_i and t_i are the ground truth. The point is to train the CNN to output a probability over the C classes.

$$f(y_i) = \frac{e^{s_i}}{\sum_j e^{s_i}} \tag{3}$$

$$CE = -\sum_i^C t_i \log\left(f(s)_i\right) \tag{4}$$

The Categorical Cross-Entropy is given as follows.

$$CE = -\log(\frac{e^{s_p}}{\sum_j^C e^{s_j}}) \tag{5}$$

Where s_p is the CNN score for the positive class.

4 Experiments

The experiments held in this section were carried out to test the efficiency and precision of the models. Metrics like accuracy, ROC, F-measure, sensitivity, and specificity were used to evaluate the models, more details about the metrics will be described in Subsect. 4.2. These models were implemented on Python 3.7, using TensorFlow 2.0.0, using a GPU to speed up the training process.

4.1 Data Set

For this work, a new dataset was created with the help of the Fruit Department of the Postgraduate School, Texcoco. As mentioned above, the dataset consists of two elements, the front face of the leaf and its back face, the pictures were taken under a controlled environment, with a white background and artificial light, Fig. 2. These leaves belong to 6 different kinds of experimental peaches hold by the institution. No image preprocessing was applied, and no data augmentation techniques were used. Every pair of leaf picture is pointing to the same side, as it can be appreciated in Fig. 2, intuitively, this would make easier the classification for the model. Different image size was used to train each model, one dataset consists of 224 × 224 pictures, and the second dataset of 416 × 416 pictures. The dataset distribution is described in Table 3, for the training process, the dataset was divided 10% for testing and the rest for training. For the test, the same "two-input" models were used, both leaf's faces are essential for the plant identification, in contrast of [24], where the model trained using two different pictures where both could be from the same plant or one from the positive target and the second from a negative target, to make the model find similarities of one class in different angles and to distinguish it from other classes. Thus, during the test only one picture is needed, to feed a one stream CNN and RNN, and this is possible because the weights were shared.

Table 3. Dataset distribution.

Class and name	Number of pairs	Training	Test
1. CP-03-06	101	90	11
2. Oro Azteca	207	186	21
3. Oro San Juan	204	179	25
4. Cardenal	229	209	20
5. Colegio	205	190	15
6. Robin	319	285	34
Total	1265	1139	126

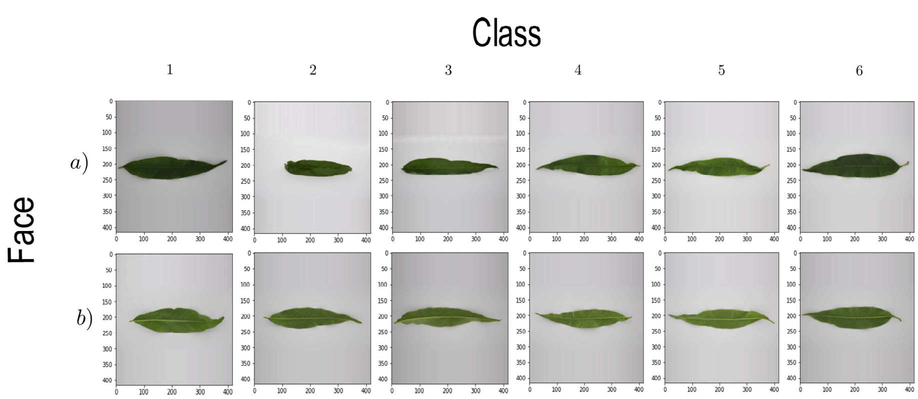

Fig. 2. Representation of the six different kinds of peach leaves. a) represents the front face of the leaf and b) its back face.

4.2 Results

As mentioned in the beginning of Sect. 4, the metrics used to test the model were accuracy, ROC, f-measure, recall and precision, here we are going to give a brief explanation about the metrics used. The accuracy is the number of correctly predicted data out of all the data points, that measure is obtained dividing the number of true positive and true negatives by the number of true positives, true negatives, false positives and false negatives. The F-measure is the harmonic mean of precision and sensitivity, and it is obtained dividing 2 times the true positives by two times the true positives plus the false positives and the false negatives. The sensitivity is the true positive rate, and it is obtained dividing the true positives by the true positives and false negatives. The specificity is the true negative rate, and it is obtained dividing the true negative by the true negatives and false positives. The AUC-ROC curve is obtained plotting the sensitivity against the specificity and obtaining its area. For all the metrics, the closest to 1 they are the better the model performance is. Given that the number of elements on each class is not homogenous, every metric was obtained with a weighted average, that is, first each metric was obtained per class, that means it was a class versus all, after that, the weighted average was obtained as in Eq. 6.

$$weighted - avg = \frac{\sum_{n=1}^{c} M_n * C_n}{\sum_{n=1}^{c} C_n} \tag{6}$$

Where M_n is the metric obtained in class n, and C_n is the number of elements in class n. The results obtained were compared with the ones obtained in previous papers [6, 7] (Table 4, 5), in which 6 different kinds of peach were classified.

Table 4. Classification results with proposed approach and comparison with the previous work.

		Accuracy	ROC	F-measure	TP rate	FP rate
Proposed models	Model (224 × 224)	92.0635%	0.990	0.920	0.920	0.978
	Model (416 × 416)	92.8571%	0.992	0.927	0.928	0.983
Complex leaves classification with features extractor [7]	Naïve Bayes	76.165%	0.922	0.761	0.762	0.050
	SVM	83.937%	0.961	0.839	0.839	0.032
	Logistic R.	73.057%	0.942	0.728	0.731	0.055
	Decision tree	72.020%	0.852	0.718	0.720	0.058
Complex identification of plants from leaves [6]	Bayes	75.64%	–	–	–	–
	BP	76.02%	–	–	–	–
	RL	73.057%	–	–	–	–
	SVM_G	83.97%	–	–	–	–

In the previous work [7], two basic CNN's were created, in order to find out how they handle the task. The models did not generalize well the data, and it was mentioned that more sophisticated models needed to be created to target this task. In Table 5, a comparison of our proposed model and the models in the previous work are shown.

Table 5. Comparison of models proposed with the models proposed in the previous work.

		Accuracy
Proposed models	Model (224 × 224)	92.0635
	Model (416 × 416)	92.8571
Previous work [7]	Convolutional model 1	62%
	Convolutional model 2	55%

For [7], the dataset size was of 498 samples, smaller than the one used for this paper that sum up 1265 pairs, and their picture size used was 300 × 400 pixels for the feature extractor method, and 416 × 416 pixels for the one stream CNN. In the case of our hybrid-CNN approach the size used in [7] was like our second model (Model 416 × 416). In [6], the dataset was even smaller, consisting of 193 elements, being this one reason why the results were poor, the resolution of the images is not specified and only accuracy was considered.

Our proposed hybrid models obtained a high accuracy of 92.0635% for Model 224 × 224, and a 92.8571% accuracy for Model 416 × 416, overpassing the SVM approach from [6] and [7], where an accuracy of 83.97% and 83.937% was obtained. The difference is notable in the case of the one stream CNN approach [7], where the higher accuracy was of 62%. The performance of both hybrid models was slightly similar, being the model trained with higher resolution the one with better accuracy, with just a difference of 0.7936. The use of higher resolution pictures did not bring out better results in our approach.

5 Conclusions

Every species of plants has different kinds of varieties, and these varieties behave different under certain environments, given to the continuing development of better varieties. That is why it is important to identify the different varieties of plants a species has; this kind of leaves are called complex leaves. As seen in Fig. 2, the leaves by simple sight have the same color, texture, and morphology, this makes these kinds of leaves hard to classify. To target this task, two models were proposed to classify 6 different kinds of peach's leaves and a new dataset was created using leaves from the Fruit Department of the Postgraduate College, Texcoco. These pictures were taken under a controlled environment with white background. To train the proposed models, the dataset was divided into two parts, the front face of the leave and its back face, this pair of samples represent one element of the dataset, for each model two datasets were used, one model trained with 224 × 224 pictures and the other with 416 × 416. The proposed models significantly overpassed the models in [6] and [7]. The results demonstrated that it is important to consider both sides of a leaf as one. The Hybrid-CNN models gained more insights about how to classify a plant from a complete leaf, instead than learning from one single side.

Nowadays there are datasets available with thousands and millions of pictures available for training a model, but unfortunately, this is not always the case. Sometimes the data is not always available, and it is of common knowledge that the more data the model train with the best its performance will be. But there is not an universal rule to state what amount of data is enough for the model to not overfit. In this work, not all classes were balanced, in Table 3, the number of elements of class 1 with which the model trained represents less than 10% of all the training set, and despite this drawback, we obtained a high accuracy. On the other hand, the Bayesian Neural Networks has demonstrated that they can handle small datasets, and avoid the overfitting, handling it as uncertainty [28, 29]. Being this our next step, by applying Bayesian CNN in combination of our proposed method, to increase the accuracy rate and see how this new approach can handle the complex leaves classification.

References

1. Zhang, Z., Sejdić, E.: Radiological images and machine learning: trends, perspectives, and prospects. Comput. Biol. Med. **108**, 354–370 (2019)
2. Amador, J.D.J., Espejel Cabrera, J., Cervantes, J., Jalili, L.D., Ruiz Castilla, J.S.: Automatic calculation of body mass index using digital image processing. In: Figueroa-García, J.C., Villegas, J.G., Orozco-Arroyave, J.R., Maya Duque, P.A. (eds.) WEA 2018. CCIS, vol. 916, pp. 309–319. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00353-1_28
3. Wang, Z., Sun, X., Zhang, Y., Ying, Z., Ma, Y.: Leaf recognition based on PCNN. Neural Comput. Appl. **27**(4), 899–908 (2015). https://doi.org/10.1007/s00521-015-1904-1
4. Alishahi, A., Farahmand, H., Prieto, N., Cozzolino, D.: Identification of transgenic foods using NIR spectroscopy: a review. Spectrochimica Acta Part A Mol. Biomol. Spectrosc. **75** (1), 1–7 (2010)
5. FAO: Save and Grow (2011)
6. Cervantes, J., Garcia Lamont, F., Rodriguez Mazahua, L., Zarco Hidalgo, A., Ruiz Castilla, J.S.: Complex identification of plants from leaves. In: Huang, D.-S., Gromiha, M.M., Han, K., Hussain, A. (eds.) ICIC 2018. LNCS (LNAI), vol. 10956, pp. 376–387. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95957-3_41
7. Ayala Niño, D., Ruíz Castilla, J.S., Arévalo Zenteno, M.D., Jalili, L.D.: Complex leaves classification with features extractor. In: Huang, D.-S., Jo, K.-H., Huang, Z.-K. (eds.) ICIC 2019. LNCS, vol. 11644, pp. 758–769. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26969-2_72
8. Garcia, F., Cervantes, J., Lopez, A., Alvarado, M.: Fruit classification by extracting color chromaticity, shape and texture features: towards an application for supermarkets. IEEE Latin Am. Trans. **14**(7), 3434–3443 (2016)
9. Brahimi, M., Boukhalfa, K., Moussaoui, A.: Deep learning for tomato diseases: classification and symptoms viausalization. Appl. Artif. Intell. **31**, 299–315 (2017)
10. Caglayan, A., Guclu, O., Can, A.B.: A plant recognition approach using shape and color features in leaf images. In: Petrosino, A. (ed.) ICIAP 2013. LNCS, vol. 8157, pp. 161–170. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41184-7_17
11. Grinblat, G.L., Uzal, L.C., Larese, M.G., Granito, P.M.: Deep learning for plant identification using vein morphological patterns. Comput. Electron. Agric. **127**, 418–424 (2016)
12. Perez, L., Wang, J.: The effectiveness of data augmentation in image classification using deep learning. arXiv:1712.04021 (2017)
13. AaKif, A., Khan, M.F.: Automatic classification of plants based on their leaves. Biosyst. Eng. **139**, 66–75 (2015)
14. 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)

15. Naresh, Y.G., Nagendraswamy, H.S.: Classification of medicinal plants: an approach using modified LBP with symbolic representation. Neurocomputing **173**(3), 1789–1797 (2016)
16. Lee, S.H., Chan, C.S., Mayo, S.J., Remagnino, P.: How deep learning extracts and learns leaf features for plant classification. Pattern Recogn. **71**, 1–12 (2017)
17. 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)
18. Neto, J.C., Meyer, G.E., Jones, D.D., Samal, A.K.: Plant species identification using Elliptic Fourier leaf shape analysis. Comput. Electron. Agric. **50**(2), 121–134 (2006)
19. Backes, A.R., Bruno, O.M.: Plant leaf identification using multi-scale fractal dimension. In: Foggia, P., Sansone, C., Vento, M. (eds.) ICIAP 2009. LNCS, vol. 5716, pp. 143–150. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04146-4_17
20. Elnemr, H.A.: Feature selection for texture-based plant leaves classification. In: 2017 Intl Conf on Advanced Control Circuits Systems (ACCS) Systems & 2017 Intl Conf on New Paradigms in Electronics & Information Technology (PEIT), Alexandria, pp. 91–97 (2017)
21. Larese, M.G., Bayá, A.E., Craviotto, R.M., Arango, M.R., Gallo, C., Granitto, P.M.: Multiscale recognition of legume varieties based on leaf venation images. Experts Syst. Appl. **41**, 4638–4647 (2014)
22. Lee, S.H., Chan, C.S., Wilkin, P., Remagnino, P.: Deep-plant: plant identification with convolutional neural networks. In: 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, pp. 452–456 (2015)
23. Paco Ramos, M.M., Paco Ramos, V.M., Fabian, A.L., Osco Mamani, E.F.: A feature extraction method based on convolutional autoencoder for plant leaves classification. In: Orjuela-Cañón, A., Figueroa-García, J., Arias-Londoño, J. (eds.) ColCACI 2019. Communications in Computer and Information Science, vol. 1096. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-36211-9_12
24. Wang, B., Wang, D.: Plant leaves classification: a few-shot learning method based on siamese network. IEEE Access **7**, 151754–151763 (2019)
25. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. arXiv:1502.01852 (2015)
26. Hayou, S., Doucet, A., Rousseau, J.: On the selection of initialization and activation function for deep neural networks. arXiv:1805.08266 (2018)
27. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, vol. 9, pp. 249–256 (2010). arXiv:1506.02158
28. Gal, Y., Ghahramani, Z.: Bayesian convolutional neural networks with Bernoulli approximate variational inference. http://proceedings.mlr.press/v9/glorot10a.html (2015)
29. Kendall, A., Gal, Y.: What uncertainties do we need in Bayesian deep learning for computer vision? arXiv:1703.04977 (2017)
30. Jalili, L.D., Morales, A., Cervantes, J., Ruiz-Castilla, J.S.: Improving the performance of leaves identification by features selection with genetic algorithms. In: Figueroa-García, J.C., López-Santana, E.R., Ferro-Escobar, R. (eds.) WEA 2016. CCIS, vol. 657, pp. 103–114. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-50880-1_10
31. Cervantes, J., García-Lamont, F., Rodríguez-Mazahua, L., Rendon, A.Y., Chau, A.L.: Recognition of Mexican sign language from frames in video sequences. In: Huang, D.-S., Jo, K.-H. (eds.) ICIC 2016. LNCS, vol. 9772, pp. 353–362. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42294-7_31
32. Ip, R.H.L., Ang, L.-M., Seng, K.P., Broster, J.C., Pratley, J.E.: Big data and machine learning for crop protection. Comput. Electron. Agric. **151**, 376–383 (2018)