Journal of Computer Science

Original Research Paper

Dataset of Selected Medicinal Plant Species of the Genus *Brachylaena*: A Comparative Application of Deep Learning Models for Plant Leaf Recognition

[1]Avuya Deyi, [2]Arnaud Nguembang Fadja, [3]Eleonora Deborah Goosen, [4]Xavier Siwe Noundou and [1]Marcellin Atemkeng

[1]*Department of Mathematics, Rhodes University, 6139 Makhanda, South Africa*
[2]*Department of Engineering, University of Ferrara, Via Saragat 1, 44122 Ferrara, Italy*
[3]*Department of Pharmacy, Faculty of Pharmacy, Rhodes University, Makhanda 6139, South Africa*
[4]*Department Pharmaceutical Sciences, School of Pharmacy, Sefako Makgatho Health Sciences University, Pretotia, 0204, South Africa*

Article history
Received: 27-02-2023
Revised: 24-04-2023
Accepted: 06-07-2023

Corresponding Authors:
Marcellin Atemkeng
Department of Mathematics, Rhodes University, 6139 Makhanda, South Africa
Email: m.atemkeng@gmail.com

Eleonora Deborah Goosen
Department of Pharmacy, Faculty of Pharmacy, Rhodes University, Makhanda 6139, South Africa
Email: l.goosen@ru.ac.za

Abstract: Since several active pharmaceutical ingredients are sourced from medicinal plants, identifying and classifying these plants are generally a valuable and essential task during the drug manufacturing process. For many years, identifying and classifying those plants have been exclusively done by experts in the domain, such as botanists and herbarium curators. Recently, powerful computer vision technologies, using deep learning or deep artificial neural networks, have been developed for classifying or identifying objects using images. A convolutional neural network is a deep learning architecture that outperforms previous state-of-the-art approaches in image classification and object detection based on its efficient feature extraction of images. This study investigated several pre-trained convolutional neural networks for identifying and classifying leaves of three species of the genus *Brachylaena*. The three species considered were *Brachylaena discolor*, *Brachylaena ilicifolia*, and *Brachylaena elliptica*. All three species are used medicinally by people in South Africa. We trained and evaluated different deep convolutional neural networks from 1259 labeled images of those plant species (at least 400 for each species) split into training, evaluation, and test sets. The best model provided a 98.26% accuracy using cross-validation with a confidence interval of ±2.16%.

Keywords: Deep Learning, Medicinal Plants Classification, *Brachylaena Discolor*, *Brachylaena Ilicifolia*, *Brachylaena*, Transfer Learning

Introduction

Organizations protecting endangered species, forestry services, pharmaceutical laboratories, physicians, botanists, and traditional healers benefit significantly from their knowledge of identifying plant species (Beentje *et al*., 2000). Identification of medicinal plants is critical and needs attention as misidentification can affect the consumers' health (Dileep and Pournami, 2019). In this instance, experts like botanists are consulted for plant species identification. It is a serious issue for non-botanist researchers who do not have the skills and expertise in plant taxonomy. It may also delay or compromise the results of their research work. Therefore, there is a need to find or implement new methodologies of plant species identification (Lukas *et al*., 2021). The subset of Machine Learning (ML) (Michie *et al*., 1994) known as Deep Learning (DL) (LeCun *et al*., 2015) has proven to be efficient in image classification and identification. There has been rapid growth in computer vision since the breakthrough of applying DL to computer vision in the ImageNet competition (Esteva *et al*., 2021). Computer vision is increasingly becoming the most reliable solution to plant classification and identification as different powerful technologies are built for image capturing (Wäldchen and Mäder, 2018). In the study titled "ImageNet Classification with Deep Convolutional Neural Networks", NIPS 2012 outperformed the traditional models of ML by a margin of 10% (Krizhevsky *et al*., 2012).

Artificial Intelligence (AI) (Selvaraj *et al*., 2019) algorithms, especially Convolution Neural Networks

(CNN), have made computer vision more powerful than ever. Several publications also describe disease identification based on computer vision of plant leaves. Ghosal and Sarkar (2020) modeled a CNN architecture based on VGG-16 and obtained 92.46% accuracy. InceptionV3, InceptionResNetV2, MobileNetV2, and EfficientNetB0 were built for 38 classes of diseases found in 14 different plant species (Haassan *et al*., 2021). The models returned acceptable results ranging from 97.02 to 99.56%. A nine-layered CNN model was trained to classify 39 different classes of plant diseases and it returned an accuracy of 96.46% (Geetharamani and Pandian, 2019). We refer the reader to (Albawi *et al*., 2017) for an in-depth discussion of CNNs.

However, there are still challenges impacting the capabilities of DL in image classification. Intra-class variation and inter-class similarities are some of the challenges. Intra-class variation applies to the variation between images of the same class. And inter-class similarities apply to the similarity between images of different classes. It is a complex task to handle since the variation and similarity of these images can be, for example, a simple scale, point of view, occlusion, or background. In this study, we tackled an inter-class similarity and an identification problem. Three species of medicinal plants belonging to the genus *Brachylaena* are identified using CNNs. The ML pipeline is depicted in (Fig. 1). The process involves collecting images of plants, labeling them, and dividing them into training, validation, and testing sets. The training set is used to train and fine-tune AlexNet, VGG-16 net, and RestNet models. The resulting models are then validated using the validation set. After tuning the hyper-parameters using the validation set, the models are finally tested on the test set. The best model achieved an accuracy of 98.26%.

Related Work

Van Hieu and Hien (2020) conducted a study to classify 12000 plant species from Vietnam. They used the MobileNetV2, VGG16, ResnetV2, and Inception Resnet V DL models. MobileNetV2 registered a relatively best performance accuracy of 83.9% with a Support Vector Machines (SVM) classifier in the four models evaluated. A recent study by Huixian (2020) used four DL models to extract plant leaf features and identify species based on image analysis in a dataset of 7 different species. KNN-based neighborhood classification, a self-organizing feature mapping algorithm called Kohene network, a back-propagation neural network, and SVM were trained and compared on a database with 200 images. The back-propagation network returned the highest recognition accuracy of 92.48%, followed by the Kohonen classifier with an accuracy of 86.78%. A 91.78% accuracy was obtained by (Sun *et al*., 2017) a study that sought to identify and classify plants from the BJFU100 dataset (Shaikh *et al*., 2018), with 10000 images. The study employed the basic structure of ResNet with 26 layers and it obtained an acceptable accuracy in a dataset that contained 100 classes (Sun *et al*., 2017).

Kho *et al.* (2017) built a model to classify three different types of Ficus intra-class species. They had a relatively small dataset of 54 images. A neural network and an SVM were used to train the dataset, resulting in 83.3% accuracy with both models. Since neural networks are data-consuming models (Kho *et al*., 2017), it is expected to improve significantly with more data. Jeon and Rhee (2017) used the GoogleNet model for plant leaf classification. It achieved just above 94% accuracy. They describe GoogleNet as a CNN model that extracts and learns feature points. GoogleNet is famous for winning the 2014 competition where it was trained with a database of 1.2 million images (Szegedy *et al*., 2015), returning the smallest loss of 6%.

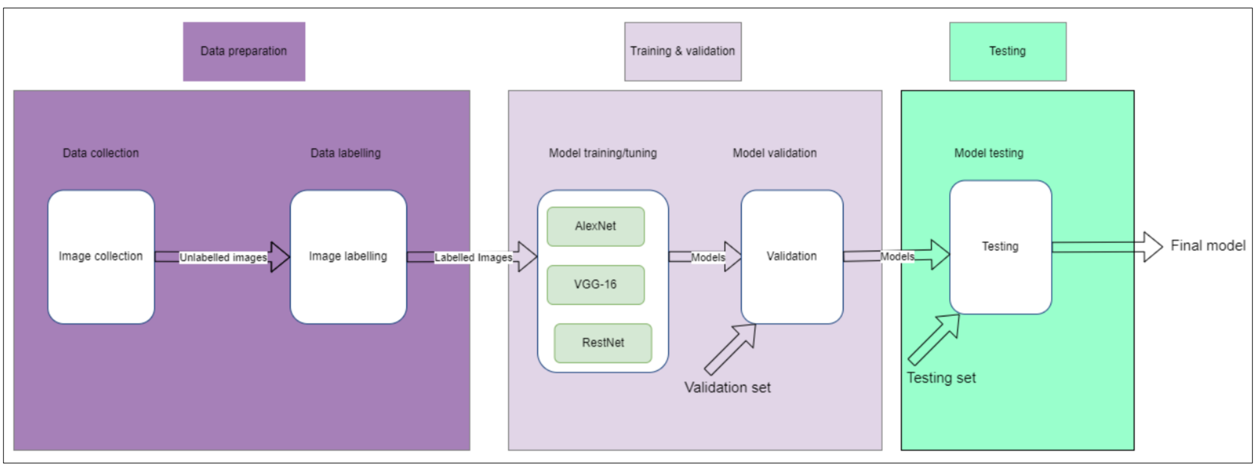

Fig. 1: ML pipeline: From data collection, labeling, training and fine-tuning to testing

DOI: 10.3844/jcssp.2023.1387.1397

Recently, a nine-layer CNN model was developed to detect and classify infected plant leaves (Geetharamani and Pandian, 2019). The CNN model competed with SVM, logistic regression, decision tree, and KNN. It achieved the highest performance of 96.46%. This study supports that CNN models excel with more extensive datasets. More than 556 million images were used to train the model and it was tested on 1950 images. Li *et al*. (2020) proposed a combination of shallow CNN and SVM classic ML classification algorithms as a good attempt for plant disease identification. They suggest that shallow CNN models must be considered before attempting deep CNN models. These shallow models provide more superficial structures and have lower computational costs. However, they struggle with complex datasets. Pawara *et al*. (2017) used the two famous CNN models; AlexNet and GoogleNet, for plant classification. AlexNet is an eight-layered neural network, where five layers are convolutional, three layers are pooling, and two fully-connected layers. GoogleNet architecture is deeper than AlexNet architecture. It introduces a new module known as inception (Shin *et al*., 2016). It also comprises two convolutional layers, two pooling layers, and nine inception layers. The inception module concatenates filters with different sizes and dimensions in a single filter (Shin *et al*., 2016). Contrary to the CNN models, KNN is a non-parametric model. It has a simple structure but is still effective in many cases (Guo *et al*., 2003). KNN is a supervised versatile ML algorithm. The algorithm considers K-nearest neighbors to predict the class or continuous value for the new input (Guo *et al*., 2003).

Dataset: Selected Medicinal Plants

The genus *Brachylaena* consists of eight species. Three, namely *Brachylaena discolor* DC. (*B. discolor*), *Brachylaena elliptica* (Thunb.) Less (*B. elliptica*) and *Brachylaena ilicifolia* (Lam.) Phillips and Schweick (*B. ilicifolia*) grow in the Makanda district municipality of the Eastern Cape Province of South Africa (Cilliers, 1993). Below we discuss the three types of *Brachylaena* species identified or classified with CNN in this study.

***B. discolor*:** Alternative names for this plant are wild silver oak (English) (Cilliers, 1993), wildevaalbos (Afrikaans) (Cilliers, 1993), umgqeba (isiXhosa) (Cocks and Dold, 2006), umpahla (isiXhosa (Dold and Cocks, 1999) and isiZulu (Burrows and Edwards, 1993), Skead; umpatha (isiXhosa) (Dold and Cocks, 1999) and isiduli (isiXhosa) (Dold and Cocks, 1999), Bantu Cancer Registry Herbarium BCRH 1112 (Dold and Cocks, 1999) and mphahla (Northern Sotho) (Burrows and Edwards, 1993). It is used in South Africa to treat diabetes and renal conditions (Watt *et al*., 1962). Dutch settlers in the region used the ashes to make soap. The AmaZulu use it to treat intestinal parasites such as roundworms. The timber is used for wagon building, boat timber, fencing posts and pick handles (Watt *et al*., 1962). It is found as an evergreen shrub or a small tree, between 4 and 8 m high on the margins of evergreen forests and in coastal woodland or bushes. The leaves are lanceolate to elliptic between 3.5 and 11.5 cm long. They are dark green on the top and pale white/grey with dense hairs at the bottom. The margins are irregularly or obscurely jaggedly toothed (Cilliers, 1993). *B. discolor* leaves were collected at the Sunnyside Garden Centre in Cromwell Street, (-33.3170600°, 26.5353751°), Makhanda on 2 March 2020 and 15 October 2021. A voucher specimen, EDG29112021, has been deposited at the Selmar Schonland Herbarium, Albany Museum, Somerset Street Makhanda, Eastern Cape, South Africa (Fig. 2).

***B. elliptica*:** Alternative names for this plant are bitter leaf (English) (Cilliers, 1993), bitterblaar (Afrikaans) (Cilliers, 1993) isiduli (isiXhosa) (Dold and Cocks, 1999) Skead; isagqeba, (isiXhosa) (Dold and Cocks, 1999). This plant is used medicinally by the amaZulu, amaXhosa, and people of European descent. It is known to treat diabetes successfully, but now clinical proof has been found by controlled observations (Dold and Cocks, 1999). The bitterness has been ascribed to the presence of glucosides (Dold and Cocks, 1999). The AmaZulu uses an infusion of decorticated roots to treat patients with breathing difficulties or as an emetic for side pain. An enema of a leaf infusion is used to treat backache and biliousness. Wild animals often eat the leaves (Watt *et al*., 1962). It is a small shrub or tree that grows up to 4 m in height. It grows at the margins of evergreen forests, in semi-arid areas, or in coastal shrubs (Cilliers, 1993). The leaves are evergreen, elliptic to ovate, and lanceolate between 2-11 cm long and 0.5-3 cm wide. They are dark green above with sparse hairs at times. White felted hairs are present at the bottom of the leaves. The margins are usually irregularly toothed and they are often, but not always, two lobes near the apex of the leaf, creating the appearance of three lobes, (Fig. 3). The leaves from the same plant had two lobes near the apex and others had no lobes near the apex. *B. elliptica*l leaves were collected in Gowie's kloof (-33,293142°, 26, 512950°), Makhanda, Eastern Cape, South Africa, on 15 October 2021. Voucher specimen number EDG15102021 has been deposited at the Selmar Schonland Herbarium, Albany Museum, Somerset Street Makhanda, Eastern Cape, South Africa.

B. ilicifolia: Alternative names for this plant are small bitter leaf (English) (Cilliers, 1993), fynbitterblaar (Afrikaans) (Cilliers, 1993) umgqeba (isiXhosa) (Cocks and Dold, 2006) and isiduli (isiXhosa) (Dold and Cocks, 1999).

Fig. 2: *Brachylaena discolor* (collected at the Sunnyside Garden Centre in Cromwell Street, (-33.3170600°, 26.5353751°), Makhanda, Eastern Cape, South Africa, on 15 October 2021)

Fig. 3: *Brachylaena elliptica*l (collected in Gowie's kloof (-33, 293142°, 26, 512950°), Makhanda, Eastern Cape, South Africa, on 15 October 2021)

Fig. 4: *Brachylaena ilicifolia* (collected on the Committee's Drift Road (-33, 2306°, 26, 2409°), Makhanda, Eastern Cape, South Africa on 15 October 2021)

B. ilicifolia is one of the 60 most traded medicinal plants (amaze) in the Eastern Cape (Cocks and Dold, 2006). Its use in treating diabetes was not reported by Watt *et al*. (1962). However, publications confirmed using a leaf infusion to treat diabetes (Cilliers, 1993). Coughs, sore throat, and asthma are treated by oral administration of a leaf infusion, and pimples of the mouth are treated by gargling with the infusion. Similarly, sheep with paratyphoid are treated with an infusion (Cocks and Dold, 2006). It is a small shrub or tree that grows up to 4 m in height (Burrows and Edwards, 1993). It grows in the bush, on rocky hillsides, and in scrub forests (Cilliers, 1993). The leaves are small, narrow, and oblong. They are lanceolate to ovate 1-4.5 cm long and 0.2-1 cm wide. They are green on top without hairs and have pale green-white leaves at the bottom. The entire margin has small teeth *B. ilicifolia* (Cilliers, 1993) leaves were collected on the Committee's Drift Road (-33, 2306°, 26, 2409°) in Makhanda Municipal District on 2 March 2020 and 15 October 2021. Voucher specimen numbers Carli Weyers Col no 1 and 2 at the Selmar Schonland Herbarium, Albany Museum, Somerset Street Makhanda, Eastern Cape, South Africa (Fig. 4).

When comparing the descriptions of the leaves and the photos above, it might not always be clear to which species a leaf might belong. The main aim of this project is to use AI to distinguish between the different species.

Materials

The leaves of the three *Brachylaena* species were collected and placed in plastic bags labeled with the name of each species, namely *B. discolor*, *B. ilicifolia*, and *B. elliptica*. Photos were taken of the plants and the co-ordinates were recorded to submit with the dried and pressed voucher specimens at the Selmar Schonland Herbarium at the Albany Museum in Makhanda, Eastern Cape, South Africa. Photos of each leaf were taken using a Canon PowerShot SX610 HS Point and Shoot camera. The camera captured the images at a resolution of 5152×2896 pixels. Each leaf's image was taken against a uniform dark grey background. A total of 1259 images were taken, 401 images belonged to *B. discolor*, 437 to *B. elliptica*, and 421 to *B. ilicifolia*. Roughly half of the images were of the tops of the leaves. The other half was of the bottoms of the leaves. Deep learning techniques were subsequently used for the classification of the three species. The models were trained on a computer equipped with an Intel Core i7 processor. The local setup involved Anaconda and Jupyter Notebook, with some models operating on Python version 3.10.12, while others were trained using Google Colab.

Methods

Multiclass Classification

The dataset used in this study is multiclass. A multiclass identification is a classification problem with three or more classes. Supervised multiclass classification algorithms assign a class label for each input data sample (Tanha *et al.*, 2020). The goal of multiclassifiers is to build a rule to predict a classy given a data sample x (Voloshynovskiy *et al.*, 2009).

Assuming that D is the dataset; $D = \{x,y\}$, where $x = \{x_1, x_2, \cdots, x_n\}$ is the input data and $y = \{y_1, y_2, \cdots, y_n\}$ is the vector of class labels associated with the input data and n is the number of data samples. We can write, $D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$ A multiclass algorithm is built from the basis of a binary classifier, by naturally extending the binary decision boundary, $h(x)$ defined as:

$$h(x) = \begin{cases} 1, if \ x is True \\ 0, otherwise \end{cases}$$

where, there are only two possible classes; 0 or 1. There are two common algorithms to build a multiclass classifier from a binary classifier; the One-vs-All and the One-vs-One algorithms.

For an m-class instances dataset, the One-vs-All generates m-binary classifiers. Since each class is represented by one and only one classifier, it is possible to learn about the class by inspecting its corresponding classifier. It is the most commonly used strategy and is a suitable default choice. This technique assigns class 1 to a given class $k \in c$ and treats all the c-k classes as 0. It is run for all $k \in c$, where c is a set of classes. If the size of c is m, then the method would build m binary classifiers. These binary classifiers return the probability that the input belongs to class 0 or 1. In other words, the classifier returns the probability that the input belongs to class k for all classes in c, and the class with a probability score bigger than a given threshold is chosen. The class with a bigger score will be selected when the threshold is not defined. The One-vs-One classifier constructs one classifier per pair of classes. During prediction, it selects the class that receives the most of the votes. Suppose two classes received equal votes. In that case, it selects the class with the highest aggregate classification confidence by summing over the pair-wise classification confidence levels computed by the underlying binary classifiers. This technique generates $m(m\text{-}1)/2$ classifiers for m classes. For an in-depth discussion of the One-vs-All algorithm and One-vs-One classifier, we refer the reader to (Yang *et al.*, 2013; Eichelberger and Sheng, 2013).

AlexNet Architecture

Many CNN architectures can be used when dealing with image classification. We employed the well-known AlexNet architecture as the basis for this project. It made a breakthrough performance of a CNN model in computer vision in 2012 in an imageNet competition (Krizhevsky *et al.*, 2012). It had an unprecedented model performance and produced a margin of approximately 10% compared to the model in the second position. It is an eight-layered network, five of which are Convolution layers (Conv) and two are Fully Connected (FC), followed by one softmax layer (output). The AlexNet model looks very similar to the LeNet model. LeNet is the earliest convolutional network structure proposed in (Zhang *et al.*, 2019). The authors used back-propagation and feed-forward neural networks to classify handwritten digits (MNIST). In this study, we fine-tuned the AlexNet model and trained several classifiers. The fine-tuning allowed us to extract higher order. The input data size of the original AlexNet model is 27×27×3, with three convolution layers followed by max-pooling (Krizhevsky *et al.*, 2012) and a ReLU activation, the Adam (Adaptive Moment Estimation) optimizer is used and a dropout rate of 0.05.

VGG-16 Architecture

VGG-16 is one of the models of DL that has produced excellent and acceptable performance in image identification and classification. Recent studies (Brima *et al.*, 2021; Sitaula and Hossain, 2021) use VGG-16 architecture to build a successful model for COVID-19 chest X-ray image classification. Lately, several researchers have shown interest in using VGG-16 for image classification, see (Zhang *et al.*, 2020; Liu and Deng, 2015; Islam *et al.*, 2019; Chitic *et al.*, 2020; Tamuly *et al.*, 2020; Fountsop *et al.*, 2020). VGG-16 is two pairs and three trios of convolutional layers, followed by a trio of dense layers, which are all separated by one pooling layer. VGG-16 is an improvement of AlexNet since it replaces the large kernel-sized filters 11×11 and 5×5 with 2×2 and 3×3 kernel sizes, respectively (Qassim *et al.*, 2018). Guan *et al.* (2019) claim that VGG-16 was built to be very deep CNN because it had to classify 1000 categories in a dataset of more than six million.

ResNet

ResNet is an abbreviation of residual neural network which was first developed in 2015 by researchers at Microsoft. It is the first working very deep feedforward artificial neural network with hundreds of layers. It is much deeper than preceding CNNs. The ResNet architecture adds an intermediate input to the output of a sequence of convolutional blocks. Deep CNNs often suffer from vanishing or exploding gradients, and ResNet has a way to avoid these obstacles. It uses skip connections to solve the vanishing gradient problem.

Cross-Validation

Selection bias and over-fitting are common challenges of ML and DL models. The holdout technique splits the dataset into two independent parts, training and testing data. This technique is likely to suffer from selection bias sinace only one part of the data is used in training. Selection bias occurs when the data used for training or testing does not represent most of the dataset's features of the population intended to be analyzed. It can lead to models performing poorly on large and unseen datasets.

The k-fold cross-validation method trains the model by splitting the dataset into k differently approximately equal-sized parts. Each of the k-parts is used for testing while the other k-1 parts are combined to train the model. It then generates k models from an architecture of one model, which produces k possible results. The recommended choices of k are 5 and 10 (Moreno-Torres *et al*., 2012). The bigger the value of k, the more unbiased the result. The most unbiased result is likely to be generated when k is the number of the data samples, which is computationally expensive.

ROC and AUC

The Receiver Operating Characteristic (ROC) curve is a probability curve used to graph the True Positives Rate (TPR) against False Positives (FPR). The area under the ROC curve, AUC, tells the model's capability to distinguish between classes. AUC measures the degree of separability between the categories. The bigger the value of AUC, the better the model. The following equations define how to compute the TPR and the FPR:

$$TPR = \frac{TP}{TP + FN}$$

where, TP is the number of true positives and it shows the number of actual positive examples classified as negative. TPR measures the sensitivity of the model to a particular class:

$$FPR = \frac{FP}{TN + FP}$$

where, FP is the number of false positives, it returns the number of actual negative examples classified as positive. TN is the number of true negatives and it shows the number of negative examples classified accurately. There is also a measure of specificity:

$$Specificity = \frac{TN}{TN + FP}$$

The Specificity is inversely proportional to sensitivity; it increases with the decrease in sensitivity and vice versa. ROC and AUC curves are very important matrices when assessing a classification model. For this study, we consider a multi-classification problem and a one-vs-all approach is employed.

Results and Discussion

We trained and tested different configurations of the AlexNet, ResNet, and VGG-16 architecture on a dataset of 1259 images. The dataset was split as follows: 66% was used for training, 14% for validation, and 20% for testing the trained models. The original AlexNet architecture was first trained and led to an accuracy of approximately 44%. Different hyper-parameters were investigated to improve the accuracy. Initially, the number of filters in each layer was modified as follows: The filters in the first layer were kept the same, i.e., 48. In the second layer, the number of filters was changed from 128-20. The successive layers whose initial filters were 192, were modified to 20 and 30, respectively. In layer seven, the number increased from 30-128. The modified architecture was trained using Adam and Nadam Optimisers with early-stopping. Note that a model trained for a long time could over-fit. Moreover, training a model for a short time does not guarantee a return of a well-skilled model as it might under-fit. As a result, the early-stopping strategy was used to monitor when to stop the training. The early-stopping strategy monitored the validation loss throughout the training process. In our experiment, the training process ended when the validation accuracy did not improve over four epochs. We set the maximum number of epochs at 20 and the training stopped at epoch 10. Other hyper-parameters such as batch size and learning rate were also investigated. For the ResNet and VGG-16 model, a pre-trained model. The grid search suggested using 6 epochs for training the configurations. Twelve models; referred to as Model i, $i = 1,\cdots,36$ were trained. The number of batch sizes explored are 32, 64, and 128, while the learning rates used are 0.001 and 0.0001. Tables 1-3 show the results of the models trained using the Adam and Nadam optimizers. The models trained with ResNet returned relatively poor results compared to models returned by VGG and AlexNet. The accuracy of ResNet models ranges between 77 and 89%, Table 1. The other two models returned an accuracy above 90%, Tables 2-3. For these reasons, we have decided to exclusively examine the results produced by VGG and AlexNet models. The testing accuracy and loss were used to assess the trained models. The ROC curve is another method used to show the ability of the best models to classify each species. To validate and support the results, we used the k-fold cross-validation.

The results show that all the models trained with Adam and Nadam optimizers are able to extract relevant features necessary to classify any plants belonging to the three categories.

AlexNet and VGG-16, both trained with the Adam optimizer, exhibited higher performance than RestNet, achieving test accuracies ranging between 95.10 and 97.14% and between 95.10 and 97.59%, respectively. The AlexNet models exhibited the highest accuracy with Model 2, which had a batch size of 64 and a learning rate of 1×10^{-3}, achieving an accuracy of 97.14%. On the other hand, among the VGG-16 models, Model 16 achieved the highest accuracy of 97.59% with a batch size of 32 and a learning rate of 1×10^{-4}.

Table 1: AlexNet models trained using the Adam and Nadam optimizers with the number of epochs 10

	Optimiser = Adam		
	Model 1	Model 2	Model 3
Batch size	32	64	128
Learning rate	0.001	0.001	0.001
Test loss	11.47%	6.29%	10.60%
Test accuracy	95.92%	97.14%	95.51%
	Model 4	Model 5	Model 6
Batch size	32	64	128
Learning rate	0.001	0.001	0.001
Test loss	10.18%	11.70%	18.59%
Test accuracy	95.18%	95.92%	95.10%
	Optimiser = Nadam		
	Model 7	Model 8	Model 9
Batch size	32	64	128
Learning rate	0.001	0.001	0.001
Test loss	13.62%	9.70%	9.31%
Test accuracy	93.88%	96.73%	98.36%
	Model 10	Model 11	Model 12
Batch size	32	64	128
Learning rate	0.0001	0.0001	0.0001
Test loss	14.48%	11.14%	17.57%
Test accuracy	93.87%	96.73%	94.69%

Table 2: VGG-16 models trained using the Adam and Nadam optimizers with the number of epochs 6

	Optimiser = Adam		
	Model 13	Model 14	Model 15
Batch size	32	64	128
Learning rate	0.001	0.001	0.001
Test loss	11.47%	6.29%	28.46%
Test accuracy	95.92%	97.14%	97.55%
	Model 16	Model 17	Model 18
Batch size	32	64	128
Learning rate	0.0001	0.0001	0.0001
Test loss	10.18%	27.05%	18.59%
Test accuracy	97.59%	95.92%	95.10%
	Optimiser = Nadam		
	Model 19	Model 20	Model 21
Batch size	32	64	128
Learning rate	0.001	0.001	0.001
Test loss	70.4%	20.18%	30.3%
Test accuracy	97.0%	97.59%	97.14%
	Model 22	Model 23	Model 24
Batch size	32	64	128
Learning rate	0.0001	0.0001	0.0001
Test loss	54.14%	13.94%	19.85%
Test accuracy	96.73%	97.55%	97.96%

When the Nadam optimizer is used with the same hyperparameters as the Adam, the test accuracy for the AlexNet models ranged between 93.87 and 98.36%, and Model 9 achieved the highest accuracy of 98.36%. With the Nadam, the test accuracy for the VGG-16 models ranged between 96.73 and 97.96% and Model 24 achieved the highest accuracy of 97.96%. Tables 1-2 show that, on average, VGG-16 models trained with Nadam optimizer outperform those of AlexNet. The two highest-performing models from both architectures were found using the Nadam optimizer: Model 9 with an accuracy of 98.36% and Model 24 with an accuracy of 97.96%.

Table 4 displays the confusion matrices for Model 9 (AlexNet) and Model 24 (VGG-16), while Table 5 summarizes the performance of the models by presenting their accuracy, F1-score, recall, precision, and specificity.

Table 3: ResNet models trained using the Adam and Nadam optimiserx with the number of epochs 10

	Optimiser = Adam		
	Model 25	Model 26	Model 27
Batch size	32	64	128
Learning rate	0.001	0.001	0.001
Test loss	36.43%	55.54%	67%
Test accuracy	86.07%	84.55%	83.33%
	Model 28	Model 29	Model 30
Batch size	32	64	128
Learning rate	0.0001	0.0001	0.0001
Test loss	44.67%	37.85%	59.4%
Test accuracy	87.34%	88.2%	87.39%
	Optimiser = Adam		
	Model 31	Model 32	Model 33
Batch size	32	64	128
Learning rate	0.001	0.001	0.001
Test loss 51	50%	%	%
Test accuracy	85.31%	82.44%	77.50%
	Model 34	Model 35	Model 36
Batch size	32	64	128
Learning rate	0.0001	0.0001	0.0001
	54.44%	42%	
Test accuracy	77.76%	85.32%	80.03%

Table 4: Confusion matrix

	Confusion matrix of the two models					
	VGG-16			AlexNet		
Metrics	Dis	Ell	Ili	Dis	Ell	Ili
TP	80	67	68	72	57	64
FP	16	4	10	36	5	11
TN	162	165	163	165	165	163
FN	3	13	14	11	23	18

Table 5: Confusion matrix summary

	Confusion matrix of the two models				
Classifier	Accuracy %	F1-score %	Recall %	Precision %	Specificity %
AlexNet (Model 9)	79	79	79	81	90.46
VGG-16 (Model 24)	88	88	88	88	94.23

Figures 5-6 display the ROC curves for Model 9 and Model 24, respectively. Both models are capable of accurately identifying and classifying all three species, with *discolor* appearing to be the species that the models classify most accurately compared to the other species.

More than half of the trained models provided (Fig. 5). ROC curves for Model 9 accuracy above 95%. The reliability and efficiency of this performance were validated by performing a cross-validation method. The cross-validation method also reduced the selection bias. We randomly split the data into five equal folds. All the hyperparameters shared by Model 24 were kept constant for cross-validation. As per the experiments in the above sections, batch sizes did not significantly impact the performance of the models. Models trained with the Nadam optimizer were, on average, more accurate than those trained with the Adam optimizer. Hence, we use the Nadam optimizer for cross-validation.

Considering that VGG-16 shows promising accuracy on average, we selected Model 24 of the VGG-16 to undergo cross-validation. The data was split into five approximately balanced equal folds for both testing and training and five models were trained, with results displayed in Table 6. The cross-validation process yielded an average accuracy of 98.26%, which is only 0.3% bigger than Model 24's accuracy. Additionally, the average loss of the cross-validation was 14.99%, which is relatively close to the loss of Model 24, which is 19.85%. These results, as shown in the aforementioned table, indicate that Model 24 is generic enough to accurately classify all three species of *Brachylaena*.

Table 6: cross-validation results

Model	Test accuracy %	Test loss %
Model-CV_0	95.28	31.28
Model-CV_1	98.81	20.12
Model-CV_2	99.60	1.15
Model-CV_3	99.60	1.55
Model-CV_4	98.03	20.86
Average	98.26	14.99

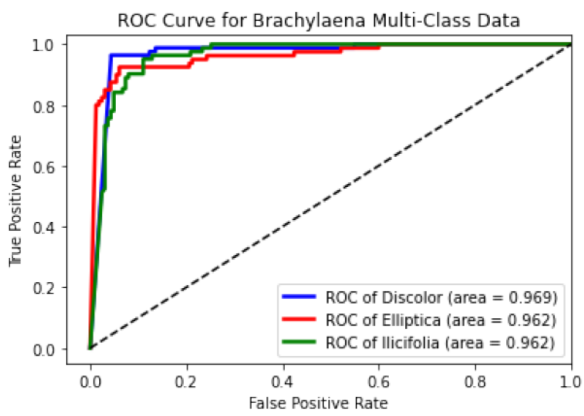

Fig. 5: ROC curves for Model 9

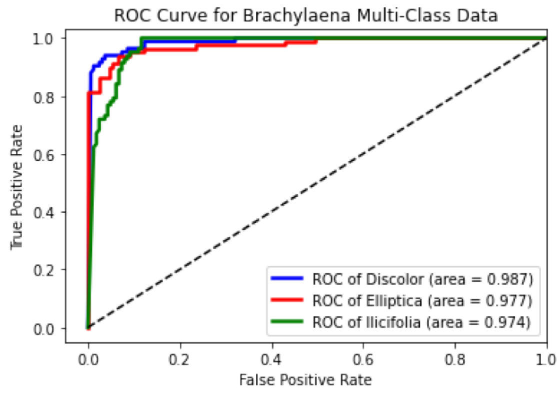

Fig. 6: ROC curves for Model 24

Conclusion

Our review of existing studies led us to conclude that CNN models are effective for image classification Fig. 6. ROC curves for Model 24 problems, a finding that this study confirms. In this study, we evaluated the performance of several CNN models, including AlexNet, pre-trained ResNet, and pre-trained VGG-16, all of which performed well in the classification of the three species in the genus *Brachylaena*, with the exception of ResNet. The VGG-16 had a classification cross-validation accuracy of 98.26% with a confidence interval of ±2.16%. The outstanding performance of AlexNet and VGG-16 highlights their efficiency in this task, with Model 24 from VGG-16 performing exceptionally well by achieving a test accuracy of 97.96%. It is worth noting that the choice of optimizer and learning rate are essential hyper-parameters that significantly affect the performance of these models.

In future work, we aim to collect more data and train the VGG-16 model, as it showed better accuracy in our current study. We also plan to use model compression techniques such as Pruning, Quantization, and Weight Sharing to create a high-speed and less computationally expensive model that could be deployed in a mobile device. This will enable people, such as pharmacists or other interested individuals, to classify these plants using their smartphones.

Acknowledgment

We sincerely thank all individuals and institutions who have contributed to the successful completion of this research project. Your support, guidance, and valuable insights have been instrumental in our endeavor.

Author's Contributions

Mr Avuya Deyi: Conceptualization, data collection, data curation, formal analysis, investigation, methodology, visualized and written original drafted.

Arnaud Nguembang Fadja: Conceptualization, formal analysis, visualization, written reviewed and edited supervision and validation.

Eleonora Goosen: Data collection, data curation, visualization, written reviewed and edited and validation.

Xavier Siwe Noundou: Conceptualization, data collection, visualization, written reviewed and edited, validation and supervision.

Marcellin Atemkeng: Conceptualization, formal analysis, visualization, written reviewed and edited, supervision, validation and funding acquisition.

Funding Information

The authors acknowledge the financial support provided by the Rhodes University Research Committee (EDG and MA).

Ethics

This article adheres to ethical guidelines and does not involve any studies on animals or human participants. Furthermore, no data gathered from social media platforms has been utilized in this research.

References

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. In 2017 *International Conference on Engineering and Technology (ICET)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICEngTechnol.2017.8308186

Beentje, H. J. (2000). The genus *Brachylaena* (Compositae: Mutisieae). *Kew Bulletin*, 1-41. https://doi.org/10.2307/4117759

Brima, Y., Atemkeng, M., Djiokap, S. T., Ebiele, J., & Tchakounte, F., (2021). Transfer learning for the detection and diagnosis of types of pneumonia including pneumonia induced by covid-19 from chest x-ray images. *Diagnostics*, *11*(8):1480. https://doi.org/10.3390/diagnostics11081480

Burrows, J. E., & Edwards, T. J. (1993). Nomenclatural changes and additions to the genus Ophioglossum in Africa (Ophioglossaceae: Pteridophyta). *Bothalia*, *23*(2), 185-190. https://doi.org/10.4102/abc.v23i2.801

Chitic, R., Bernard, N., & Leprévost, F. (2020, March). A proof of concept to deceive humans and machines at image classification with evolutionary algorithms. In *Asian Conference on Intelligent Information and Database Systems* (pp. 467-480). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-42058-1_39

Cilliers, S. S. (1993). Synopsis of the genus *Brachylaena* (Asteraceae) in southern Africa. *Bothalia*, *23*(2), 175-184. https://doi.org/10.4102/abc.v23i2.800

Cocks, M. L., & Dold, A. P. (2006). Cultural significance of biodiversity: the role of medicinal plants in urban African cultural practices in the Eastern Cape, South Africa. *Journal of Ethnobiology*, *26*(1), 60-81. https://doi.org/10.2993/0278-0771_2006_26_60_csobtr_2.0.co_2

Dileep, M. R., & Pournami, P. N. (2019, October). AyurLeaf: a deep learning approach for classification of medicinal plants. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)* (pp. 321-325). IEEE. https://doi.org/10.1109/TENCON.2019.8929394

Dold, A. P., & Cocks, M. L. (1999). Preliminary list of Xhosa plant names from Eastern Cape, South Africa. *Bothalia*, *29*(2), 267-292. https://doi.org/10.4102/abc.v29i2.601

Eichelberger, R., & Sheng, V. (2013, June). Does one-against-all or one-against-one improve the performance of multiclass classifications?. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. *27*, No. 1, pp. 1609-1610). https://doi.org/10.1609/aaai.v27i1.8522

Esteva, A., Chou, K., Yeung, S., Naik, N., Madani, A., Mottaghi, A., ... & Socher, R. (2021). Deep learning-enabled medical computer vision. *NPJ Digital Medicine*, *4*(1), 5. https://doi.org/10.1038/s41746-020-00376-2

Fountsop, A. N., Ebongue Kedieng Fendji, J. L., & Atemkeng, M. (2020). Deep learning models compression for agricultural plants. *Applied Sciences*, *10*(19), 6866. https://doi.org/10.3390/app10196866

Geetharamani, G., & Pandian, A. (2019). Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Computers & Electrical Engineering*, *76*, 323-338. https://doi.org/10.1016/j.compeleceng.2019.04.011

Ghosal, S., & Sarkar, K. (2020, February). Rice leaf diseases classification using CNN with transfer learning. In 2020 *IEEE Calcutta Conference (CALCON)* (pp. 230-236). IEEE. https://doi.org/10.1109/CALCON49167.2020.9106423

Guan, Q., Wang, Y., Ping, B., Li, D., Du, J., Qin, Y., ... & Xiang, J. (2019). Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: A pilot study. *Journal of Cancer*, *10*(20), 4876. https://doi.org/10.7150/jca.28769

Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN model-based approach in classification. In *on the Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings* (pp. 986-996). https://doi.org/10.1007/978-3-540-39964-3_62

Haassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z., & Jasińska, E. (2021). Identification of plant-leaf diseases using CNN and transfer-learning approach. *Electronics*, *10*(12), 1388. https://doi.org/10.3390/electronics10121388

Huixian, J. (2020). The analysis of plants image recognition based on deep learning and artificial neural network. *IEEE Access*, *8*, 68828-68841. https://doi.org/10.1109/ACCESS.2020.2986946

Islam, S., Khan, S. I. A., Abedin, M. M., Habibullah, K. M., & Das, A. K. (2019, July). Bird species classification from an image using VGG-16 network. In *Proceedings of the 7th International Conference on Computer and Communications Management* (pp. 38-42). https://doi.org/10.1145/3348445.3348480

Jeon, W. S., & Rhee, S. Y. (2017). Plant leaf recognition using a convolution neural network. *International Journal of Fuzzy Logic and Intelligent Systems*, *17*(1), 26-34. https://doi.org/10.5391/IJFIS.2017.17.1.26

Kho, S. J., Manickam, S., Malek, S., Mosleh, M., & Dhillon, S. K. (2017). Automated plant identification using artificial neural network and support vector machine. *Frontiers in Life Science*, *10*(1), 98-107. https://doi.org/10.1080/21553769.2017.1412361

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *25*. https://doi.org/10.1145/3065386

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444. https://doi.org/10.1038/nature14539

Li, Y., Nie, J., & Chao, X. (2020). Do we really need deep CNN for plant diseases identification?. *Computers and Electronics in Agriculture*, *178*, 105803. https://doi.org/10.1016/j.compag.2020.105803

Liu, S., & Deng, W. (2015, November). Very deep convolutional neural network-based image classification using small training sample size. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)* (pp. 730-734). IEEE. https://doi.org/10.1109/ACPR.2015.7486599

Lukas, N., Jiang, E., Li, X., & Kerschbaum, F. (2021). Sok: How robust is image classification deep neural network watermarking? (extended version). *arXiv Preprint arXiv:2108.04974*. https://doi.org/10.48550/arXiv.2108.04974

Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). Machine learning, neural and statistical classification.

Moreno-Torres, J. G., Sáez, J. A., & Herrera, F. (2012). Study on the impact of partition-induced dataset shift on $ k $-fold cross-validation. *IEEE Transactions on Neural Networks and Learning Systems*, *23*(8), 1304-1312. https://doi.org/10.1109/TNNLS.2012.2199516

Pawara, P., Okafor, E., Schomaker, L., & Wiering, M. (2017). Data augmentation for plant classification. In *Advanced Concepts for Intelligent Vision Systems: 18th International Conference, ACIVS 2017, Antwerp, Belgium, September 18-21, 2017, Proceedings 18* (pp. 615-626). Springer International Publishing. https://doi.org/10.1007/978-3-319-70353-4_52

Qassim, H., Verma, A., & Feinzimer, D. (2018, January). Compressed residual-VGG16 CNN model for big data places image recognition. In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 169-175). IEEE. https://doi.org/10.1109/CCWC.2018.8301729

Selvaraj, M. G., Vergara, A., Ruiz, H., Safari, N., Elayabalan, S., Ocimati, W., & Blomme, G. (2019). AI-powered banana diseases and pest detection. *Plant Methods*, *15*, 1-11. https://doi.org/10.1186/s13007-019-0475-z

Shaikh, N. A., Mallah, G. A., Karaş, İ. R., & Akay, A. E. (2018). Using mobile image recognition system for nonwood species identification in the field. *Journal of Innovative Science and Engineering*, *2*(2), 40-50.

Shin, H. C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., ... & Summers, R. M. (2016). Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Transactions on Medical Imaging*, *35*(5), 1285-1298. https://doi.org/10.1109/TMI.2016.2528162

Sitaula, C., & Hossain, M. B. (2021). Attention-based VGG-16 model for COVID-19 chest X-ray image classification. *Applied Intelligence*, *51*, 2850-2863. https://doi.org/10.1007/s10489-020-02055-x

Sun, Y., Liu, Y., Wang, G., & Zhang, H. (2017). Deep learning for plant identification in natural environment. *Computational Intelligence and Neuroscience*, 2017. https://doi.org/10.1155/2017/7361042

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-9).

Tamuly, S., Jyotsna, C., & Amudha, J. (2020). Deep learning model for image classification. In *Computational Vision and Bio-Inspired Computing: ICCVBIC 2019* (pp. 312-320). Springer International Publishing. https://doi.org/10.1007/978-3-030-37218-7_36

Tanha, J., Abdi, Y., Samadi, N., Razzaghi, N., & Asadpour, M. (2020). Boosting methods for multi-class imbalanced data classification: an experimental review. *Journal of Big Data*, *7*, 1-47. https://doi.org/10.1186/s40537-020-00349-y

Van Hieu, N., & Hien, N. L. H. (2020). Automatic plant image identification of Vietnamese species using deep learning models. *arXiv preprint arXiv*:*2005*.02832. https://doi.org/10.48550/arXiv.2005.02832

Voloshynovskiy, S., Koval, O., Beekhof, F., & Holotyak, T. (2009, September). Multiclass classification based on binary classifiers: On coding matrix design, reliability and maximum number of classes. In *2009 IEEE International Workshop on Machine Learning for Signal Processing* (pp. 1-6). IEEE. https://doi.org/10.1109/MLSP.2009.5306207

Wäldchen, J., & Mäder, P. (2018). Plant species identification using computer vision techniques: A systematic literature review. *Archives of Computational Methods in Engineering*, *25*, 507-543. https://doi.org/10.1007/s11831-016-9206-z

Watt, J. M., & Breyer-Brandwijk, M. G. (1962). The Medicinal and Poisonous Plants of Southern and Eastern Africa being an Account of their Medicinal and other Uses, Chemical Composition, Pharmacological Effects and Toxicology in Man and Animal. *The Medicinal and Poisonous Plants of Southern and Eastern Africa being an Account of their Medicinal and other Uses, Chemical Composition, Pharmacological Effects and Toxicology in Man and Animal*., (2nd Edn.). https://www.cabdirect.org/cabdirect/abstract/19622902105

Yang, X., Yu, Q., He, L., & Guo, T. (2013). The one-against-all partition based binary tree support vector machine algorithms for multi-class classification. *Neurocomputing*, *113*, 1-7. https://doi.org/10.1016/j.neucom.2012.12.048

Zhang, C. W., Yang, M. Y., Zeng, H. J., & Wen, J. P. (2019). Pedestrian detection based on improved Le-Net-5 convolutional neural network. *Journal of Algorithms & Computational Technology*, *13*, 1748302619873601. https://doi.org/10.1177/1748302619873601

Zhang, Q., Bai, C., Liu, Z., Yang, L. T., Yu, H., Zhao, J., & Yuan, H. (2020). A GPU-based residual network for medical image classification in smart medicine. *Information Sciences*, *536*, 91-100. https://doi.org/10.1016/j.ins.2020.05.013