PLANT LEAF DISEASE IDENTIFICATION USING CONVOLUTIONAL NEURAL NETWORK

Data Mining and Machine Learning II

Pedro Acosta
School of Computing
National College of Ireland
Dublin, Ireland
x21138745@student.ncirl.ie

Brendan O'Dwyer
School of Computing
National College of Ireland
Dublin, Ireland
x21145172@student.ncirl.ie

Femi Adeboye
School of Computing
National College of Ireland
Dublin, Ireland
x21137684@student.ncirl.ie

Antonio Milian y Albacar
School of Computing
National College of Ireland
Dublin, Ireland
x19172125@student.ncirl.ie

***Abstract*—Detecting plant diseases at an early stage allows the implementation of plans to control their propagation, the current research offers a tangible solution that could be used by all types of farmers or gardeners outdoors regardless of the size of the business, harvest and natural outdoor conditions including surroundings. It is technically flexible as it could be implemented either in traditional mobile devices with cameras such as smartphones and tablets as well as in bigger and complex systems such as an embedded module that could be integrated with modern pesticide systems. The current investigation evaluates if using CNN could help to create reliable and robust classification models in indoor, outdoor and mixed conditions. Particularly, outdoor conditions introduce naturally more complexity in image classification, besides, the lack of datasets available with outdoor images labelled adds an important limitation to the research. Those challenges were addressed by applying data augmentation techniques and transfer learning, and pre-trained model layers using convolutional layers were evaluated and compared. MobileNet v.2 was selected for having better accuracy and faster findings in terms of accuracy, the comparison between all datasets suggests that it is not worth training a model in a mixed dataset.**

***Index Terms*—Convolutional Neural Networks, Plant Disease Detection, Adam, Hyperparameters Optimization**

I. Introduction

Complex problems in sensible areas such as agriculture demand the implementation of early detection actions that enable corrective plans to prevent consequences that impact not only finances but the life quality of people. Nowadays geopolitical conflicts urge all stakeholders in society to reduce errors and waste of resources, especially at any stage of the food value chain, the current project aims to contribute and make possible early disease detection of plants through image classification.

This research aims to contribute not only to make available the results and knowledge obtained from the techniques implemented but also to suggest its integration or embedment as part of either a simple or wider solution that includes not only indoor but also outdoor data. The project addressed the natural and technical challenges of the datasets through experiments and results comparison.

Since one of the core objectives of the project is to provide a reliable model that identifies plants diseases, the images taken outdoor include an additional level of complexity due to dynamic environmental factors such as noise, illumination, background, overlapping leaves and natural disease signs, all of them add noise to the image taken, this challenge demands the right application of the technique including modelling, hyperparameters optimization and tunning among some other methods required to work out any particular characteristics and limitations of each dataset.

The research question is orientated to compare the performance of four classification models applying Convolutional Neural Network (CNN), the model with the highest performance in terms of accuracy was selected. The first model was trained using the PlantVillage dataset with indoors and colorful images, the second model was trained with the Plan Doc dataset including outdoor colorful images, the third dataset was trained with a combined dataset merging both previous datasets and finally a fourth model was trained using the PlantVillage dataset with segmented images. Due to data limitation and imbalanced data, transfer learning was applied to all the models, data augmentation was applied specifically to the Plant Doc dataset to balance some classes where the number of images was not enough to train the model.

The current project is presented in 4 sections, the related work included a review of 21 papers, it focused mainly on the application of CNN on image processing applied both with limited and sufficient data, the methodology is followed by the process model CRISP-DM and it discloses how images were pre-processed and standardized to ensure compatibility with the CNN used for transfer learning, how the techniques and methods were applied to generate each model, this included the comparison between two well-known convolutional pre-trained models used for transfer learning, it also includes results with visual content analyzing each

situation, finally, conclusions suggests relevant future works related to this research, natural limitations of the investigation and the implementation of additional techniques.

II. Related work

The author used the dataset 'ImageClef2013' Plant Identification to train the CNN model for image recognition, transfer learning and parameter fine-tuning were implemented, and results outperformed traditional Machine Learning (ML) methods [1]. Using Transfer Learning techniques to apply an image classifier to other seismic data, the author optimized and trained CNNs on synthetic seismic data to form a base model, the results of Transfer Learning for the Netherlands offshore F3 block showed the technique is suitable for practical use, it can be tuned and trained easily on a CPU in a few minutes [2]. Plant disease detection included challenges such as the position of a pest due to complex background also, errors in labeling and limited number of samples, this paper proposes a model called Selective Kernel MobileNet (SK-MobileNet) that significantly reduces server computing achieving an accuracy of 99.28% [3]. A transfer learning technique in deep learning is used to keep track of expiry dates in products, the CNN - Inception ResNet V2 was trained on synthetic data containing images of near-realistic expiration dates, Inception ResNet V2 was accurate to 0.9964, and a noisy image to 0.9612 [4]. A research to identify Autism Spectrum Disorder (ASD) using a transfer learning-based approach and fMRI images was deployed, according to the author, the performance can be improved by using pre-train 'ImageNet' models, when trained with ImageNet weights, Inception v3 classified epi images 98% accurately [5].

An image classifier built using an existing model VGG - 16 with Deep CNN is used in this research to classify images providing 72.40% validation accuracy. After fine-tunning, an accuracy of 79.20 % was achieved with image augmentation. The author employed VGG-16 which has been trained on a large dataset of images and fine-tuned using image augmentation to gain accuracy of 95.40 % [6]. The next research proposed a method capable of estimating the age of an individual based on facial images. Audience benchmark results were significantly outperformed by the proposed work by approximately 12% [7].

The author's review included some approaches using Deep learning methods to colorize grayscale images. InceptionResNet-v2 pre-trained model and a deep CNN were combined in this model, the author used a subset of 'ImageNet'; the algorithm efficiency to recognize unseen images depends on their specific content. In conclusion, a larger training dataset should be used to train the network and overcome this challenge [8]. The study shows that captioning can yield plausible results using deep CNNs and a well-chosen objective function. It is still necessary to improve the ability to form and connect sentences properly. There are often partial errors in images caused by not addressing details, (e.g., correcting a picture of a dog walking through the grass as a picture of a dog walking in an enclosure because bushes are in the background). This study explored the effect of emitted words on hidden states in the LSTM, which is a novel contribution to this field [9]. Using CNN, the author develops a model for detecting bridge cracks end-to-end. A genetic algorithm-based K-means clustering method (GKA) performs accurate segmentation of a target area and improved detection speeds. Accuracy, recall, F-measure, and frames per second, were 99.03, 99.79, and 196, respectively [10].

This paper proposes the use of face-based age recognition technology to improve the efficiency of ticket inspections. A simple CNN was replaced in this paper to fine-tune Inception ResnetV2 to recognize facial features. Age recognition accuracy improved significantly according to the experiment results [11]. The differences in some parts of the brain can be used to detect ASD, InceptionResNetV2 was proposed to be used with the augmented dataset to utilize transfer learning. Training, validation, and testing results remained at 70.22%, 57.75%, and 57.6%, respectively, after freezing layer by layer, up to 2.6% better performance was obtained with the transfer learning approach than with CNN [12].

In conjunction with fine-tuning trained CNN such as Xception, Inception-Resnet, VGNets, Mobilenet, and Densenet, this study proposes a novel crop/weed identification system. The approach was evaluated by generating unrestricted access datasets of two crops, tomato and cotton as well as other species using RGB cameras under natural variable lighting. Based on the results, a fine-tuned Densenet and Support Vector Machine combination achieved a micro F1 score of 99.29% and over 95% F1 score [13]. The next study attempted to classify, new CNN models were trained and tested using the Adience data set, 88.5% of gender classifications were accurate, this was the best accuracy value in comparison with ML methods and other CNN models [14]. This paper evaluates four different deep learning models with ML algorithms, the hybrid CNN+LSTM model performed better with 97.16 % accuracy, the MLP deep learning model provides the worst performance on the dataset, all three deep learning methods perform better than ML algorithms, except for MLP, which gets more than 95.00% accuracy [15]. Deep Learning Neural Network,(DLNN) were introduced as a model to assess landslide susceptibility and their predictive performance compared to existing ML methods. The DLNN model was compared with MLP-NN, SVM, C4.5, and RF models, it outperformed all of them. It was found that deep learning approaches can be considered as satisfactory alternatives for landslide susceptibility mapping, although it has been rarely used in landslide susceptibility assessments [16].

Automatic disease detection that segmented and annotated images can be used instead of full images. The performance of CNN models on independent data increases from 42.3% to 98.6% when trained on segmented images (S-CNN) rather than full images (F-CNN). It is important to pre-process images before CNN [17]. An architecture for detecting plant leaf diseases using CNNs achieves a disease classification accuracy of 95.81% and various observations have been made with different CNN hyperparameters [18]. Using computer vision

to detect tomato diseases by analysing leaf images captured. Three types of tomato diseases have been detected with MobileNet V2. A total of 4,671 images from the PlantVillage dataset were used and more than 90% of the disease can be detected with MobileNet V2 [19]. Identifying cash crop diseases by combining an Automatic Image Segmentation Algorithm (AISA) with deep learning. AISA removes the background information from images, public dataset were added to PlantVillage to increase MobileNet's performance, the author estimates it correctly recognizes 27 diseases in 6 crops with a correct recognition rate of more than 80% [20]. A two-step comparative evaluation of plant disease has been proposed based on deep learning, using the Adam optimizer, the Xception architecture was able to achieve 99.81% validation accuracy and 0.9978 F1 score [7].

III. Methodology

The CRoss Industry Standard Process for Data Mining (CRISP-DM) data mining methodology has been followed in this study. The data mining lifecycle is described by the five phases of this methodology. It is a flexible and iterative process. Phases can be revisited as the project progresses. This study will involve training different models and as a result phases will be revisited repeatedly. Business understanding is the first phase in CRISP-DM. The data mining goals are determined in this phase and an understanding of the objectives and requirements of the project developed. In the introduction the objectives of the study have been outlined. The business understanding phase is addressed through these objectives as well as through the understanding developed from the related work publications. In the remainder of this report a description of the other CRISP-DM phases is provided.

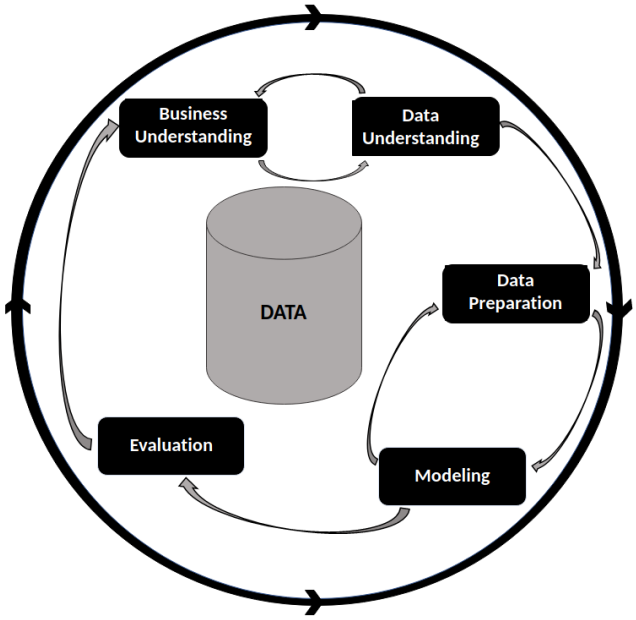

Fig. 1. The phases of the CRISP-DM cycle. Business understanding, data understanding, data preparation, modelling and evaluation in this case there is no deployment phase as the evaluation is the output.

A. Data Understanding

Exploration and collection of data is carried out in the CRISP-DM data understanding phase. An understanding of the data is obtained as well as an understanding of how the objectives can be addressed through the data. Two datasets are used in this study. The first, PlantVillage [21], contains labelled images with diseased and healthy plant leaves. The images are taken indoors in laboratory conditions. The data are made available through the PlantVillage online platform. In total there are over 50,000 images of crop plant leaves that are infected or healthy. There are 14 different types of plant species with 38 classes in total. Each class corresponds to a plant-disease pair.

The second dataset, Plant Doc [22], is for detecting plant disease from images. Unlike the first dataset the images are taken outdoors rather than in laboratory conditions. There are 2,598 images in total with 13 species of plants in total with 17 disease classes. The data were obtained by scraping images from webpages and annotating them.

The PlantVillage dataset contains coloured, segmented and greyscale versions of each image. The coloured and segmented versions of the images were used. In the case of Plant Doc only colour images are provided. This study used 13 classes with 18,317 images from both datasets, PlantVillage and Plant Doc.

B. Data Preparation

This phase of CRISP-DM involves manipulating data acquired in the data understanding phase to prepare it for the modelling phase. The datasets were explored using Python. The feature selection is based on the commonalities between the previously described datasets.

Three plant species present in both the PlantVillage and Plant Doc datasets were selected apple, pepper and tomato. In total there are 13 disease classes for these plants common to both datasets. These classes are apple scab leaf, apple healthy leaf, apple rust leaf, bell pepper leaf healthy, bell pepper leaf spot, tomato early blight leaf, tomato septoria leaf spot, tomato leaf healthy, tomato leaf bacterial spot, tomato leaf late blight, tomato leaf mosaic virus, tomato leaf yellow virus and tomato mold leaf, more details for the classes in Table I.

Two test datasets were created. The first uses only images from the dataset of indoor laboratory data. The second consists of images taken outdoors from the Plant Doc dataset. For each of the PlantVillage and Plant Doc datasets the remaining data are assigned randomly to training datasets, validation datasets and datasets used for tuning hyperparameters. 72% of the data are assigned to the training datasets, 20% are assigned to the validation datasets and 8% are assigned to the datasets for tuning hyperparameters.

All images were adjusted to have the dimensions 240 pixels by 240 pixels for InceptionResNetv.2 and to 224 x224 when using MobileNetv.2. The PlantVillage dataset has more images for each class than the Plant Doc dataset. As a result of this data augmentation techniques were applied to the Plant Doc dataset. These techniques included rotation, shearing,

TABLE I
DATA SETS USED

Classes Ojbective	Data sets			
	Indoor	*Outdoor*	*O.Augmented*	*Combined*
apple scab	630	87	1307	1937
apple healthy	275	89	1392	1667
apple rust	1645	91	1325	2970
bell pepper healthy	997	71	1355	2352
bell pepper spot	1478	61	1581	3059
tomato early blight	2127	107	1256	3383
tomato septoria	1591	62	1392	2983
tomato healthy	1000	83	1379	2379
tomato bacterial	1771	148	1272	3043
tomato late blight	1909	111	1306	3215
tomato mosaic	373	54	1131	1504
tomato yellow virus	5357	75	1172	6529
tomato mold	952	91	1444	2396

[a]Original Data set classes and values.

zooming, flipping horizontally and adjusting the brightness range. The data was pre-processed. The pixel values are rescaled.

C. Modelling

CNN was the model typology used to analyse the data and take advantage of the properties of the models in image recognition, Modelling techniques are used to build and assess models in the CRISP-DM modelling phase. A batch size of 32 was used. As convolutional layers two types are taken into consideration, Inception ResNet, in this case the pre-trained convolutional neural network and Inception-ResNet-v2 was used for transfer learning. Weights trained on the ImageNet dataset, which has one thousand classes and 1.4 million images, were used. On top of this model, a new classifier is added. The layers of the base convolutional model are frozen. The second type is MobileNet. In addition to Inception-ResNet-v2 the pretrained convolutional neural network MobileNet V2 was also used for transfer learning. The data were again preprocessed in preparation for use with the MobileNet V2 model. Weights trained on the ImageNet dataset were used. As before a new classifier was added on top of this model and the layers of the base convolutional model were frozen.

D. Hyperparameter Optimisation

Adding to the convolutional layers described the optimizers Adam and Adagrad were also tested. Combining the different values of convolutional layers and optimizers several models are proposed as potential candidates. The models are passed to a build model function in which several parameters are proposed for the rate of drop on the and the learning rate. A few values are provided for both variables. The optimization is centered in finding the best values for drop rate and from the learning rate. The model with the Convolution layer is frozen and a range of values proposed for both variables. The values for learning rates ranging from 0.01 to 0.00001, and the drop rates used range from 0.1 to 0.2. From the algorithm point of view, the library keras tuner has been used to ascertain the optimal values. From keras tuner, the Hyperband tuning algorithm is chosen. The algorithm has the championship bracket approach, running the models and choosing only the half best performers to be carried onto the next epoch. This approach is faster and guarantees better results than a simple Grid optimization approach. The keras tuner implementation has been run for 6 epochs in each trial, obtaining the optimal learning rate for each model as well as the optimal number of epochs. Accuracy over the validation is the metric used to find the best parameter values. The goal is to decide which model's architecture render the best validation accuracy and continue with that model through the fine-tuning process.

Same methodology has been used for the three different implementations; the values gathered have subsequently used in the tested models. The three models were trained using the three different data sets, Indoors, Outdoor and Combined. The outdoor model is using the Outdoor data set, with the augmentation images for training.

Applying the Hyperparameter optimization over the datasets the below results are obtained,

E. Model using the Indoor images

Applying the Hyperparameter tuning to the indoor data set, the model that offers the best results has a convolutional layer based on MobileNet v.2 with the Adam optimizer, which gives not only better accuracy at 96.9% but is also 4 times faster than the ResNet v.2 (94.5%) with the same optimizer. Trying Adagrad as optimizer for MobileNet v.2 the accuracy obtained 95%, so the model chosen for the indoor data set is MobileNet v.2 with Adam optimizer with the parameters obtained from the hyperparameter tuner.

The images are transformed into 224 x 224 pixels, to match with MobileNet requirements. The drop rate is of 0.2 and the learning rate 0.0001 as provided by the Keras tuner algorithm. The dense layer has 1280 nodes and 16,653 trainable parameters. The model is then trained for ten epochs

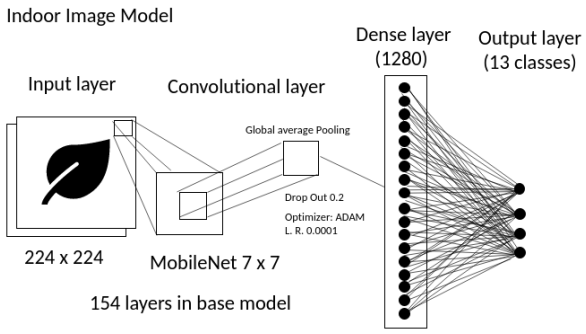

Fig. 2. MobileNet Indoor model schema.

before fine tuning. The accuracy obtained was 91.91% for the training subset and 92.82% for the validation subset as show in figure 3. The model was trained for another ten epochs for fine tuning. For fine tuning as series of modifications were

introduced, the learning rate was reduced by a factor of 10 and the layers from 100 upwards were used in fine-tuning the model. For this model the total number of parameters was again 2,274,637, but in this case 1,878,093 of these were trainable. e fine tuning epochs a big increase on the accuracy is observed, where on the epoch 17 the accuracy of the validation reaches its maximum and the one from the training subset are similar at 98.55%. For reference, in [20] for classification of crop disease the authors carry out a comparison of deep learning architectures and optimisers.

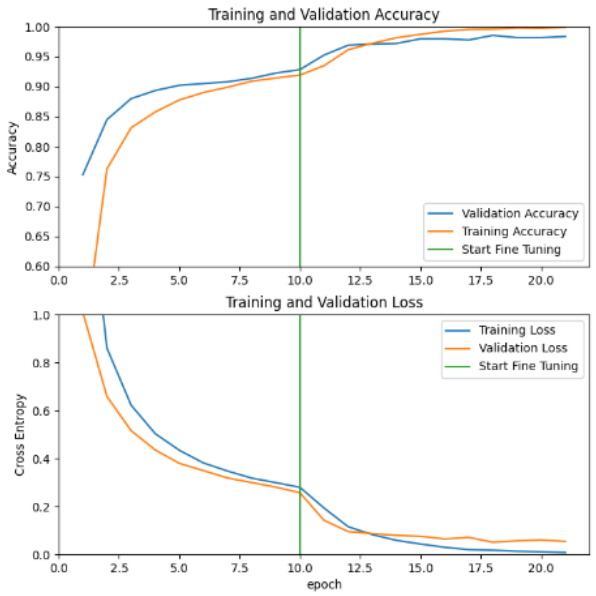

Fig. 3. Model MobileNet v.2 with Adam Optimizer Indoor dataset for Accuracy and Cross Entropy.

Similar behaviour can be observed using the metrics Precision and Recall, when around the 16 or 17 epoch the validation and training subsets provide the best outcome values.

For this model the total number of parameters was again 2,274,637, but in this case 1,878,093 of these were trainable. e fine tuning epochs a big increase on the accuracy is observed, where on the epoch 12 the accuracy of the validation and the one from the training subset are similar at 97.4%. For reference, in [20] for classification of crop disease the authors carry out a comparison of deep learning architectures and optimisers. The best classification accuracy for the validation set was 99.81% obtained using the Adam optimiser to train the Xception architecture. Similar behaviour can be observed using the metrics Precision and Recall, when around the 11 or 12 epoch the validation and training subsets provide similar values.

F. Model using the Outdoor images

Applying the Hyperparameter tuning to the outdoor data set, the model that offers the best results uses the pretrained convolutional neural network MobileNet v.2 with the Adam optimiser, which gives not only better accuracy at 89.7

The images are transformed into 224 x 224 pixels, to match with MobileNet requirements. The drop rate is of 0.1 and the learning rate is 10-5 as provided by the Keras tuner algorithm. The dense layer has 1280 nodes and 16,653 trainable parameters.

The model is then trained for ten epochs before fine tuning. The accuracy obtained was 40.9% for the training subset and 45.2% for the validation subset as shown in Figure x. The model was trained for another ten epochs for fine tuning. For fine tuning a series of modifications were introduced. The learning rate was reduced by a factor of 10. As the model we are training is much larger we reduce the learning rate. Our model may overfit quickly if at this stage we don't adjust the learning rate. The layers from 100 upwards were used in fine-tuning the model. For this model the total number of parameters was again 2,274,637, but in this case 1,878,093 of these were trainable. In the fine-tuning epochs, a big increase on the accuracy is observed, where on the epoch 13 the accuracy of the validation and the one from the training subset are similar. The best validation accuracy obtained was 97.9

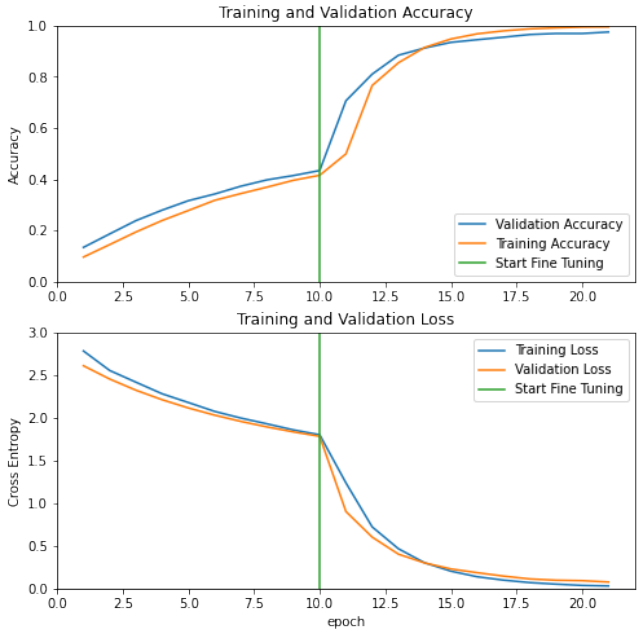

Fig. 4. For model trained on the augmented Plant Doc outdoor data with transfer learning using the pretrained MobileNet V2 CNN with fine tuning and Adam optimiser training and validation accuracy are plotted against epoch number in the upper panel. In the lower panel the training and validation loss are plotted against epoch number.

Similar behaviour can be observed using the metrics precision and recall, when around epoch 13 the validation and training subsets provide similar values.

G. Combined data model

The same methodology is applied to a dataset that combines the pictures from the outdoor and indoor data sets. The outdoor data includes the augmented added items in the training subset. From the data set thus conformed, a training data set of 33394

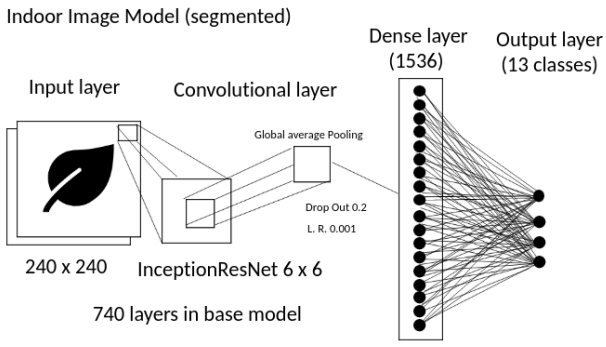

Fig. 5. InceptionResNetv.2 Indoor Segmented model schema.

images is extracted and further divided into 30052 for the training side and 3339 for the validation. Using these images to compare the different models proposed, InceptionResNetV2 with Adam as optimizer, renders the best accuracy at 6 epochs for the optimized values of 0.1 drop and 0.00001 learning rate of 85.56%. MobileNet with Adagrad for the same number of epochs and optimized parameters, Learning Rate, 0.01 and drop, 0.1 renders better accuracy at 87.24%. The best accuracy is obtained using the MobileNet with Adam, with the optimised values of 0.0001 for the learning rate and 0.2 for the drop at 90.35%. It is also worth to acknowledge that the MobileNet instances were 4 times faster than the InceptionResNetV2, which would make them more compelling even with similar results.

Using the best model from the hyperparameter tunning phase, it is trained for 10 epochs in a first phase where the layers are frozen for training, therefore only the parameters after the dense layer are trainable, 16,653 in total. The accuracy obtained is 79.19% for the training set and 80.20% for the validation set.

.

After the first training phase is over, some of the layers of the convolutional part are made trainable, in this case 100 of them are kept frozen and 56 are deemed trainable, bringing the number of trainable parameters up to 1,878,093. During the fine-tuning phase, a great improvement can be observed in all metrics analyzed, up to the 14 epochs when the results of the validation subset go under the training ones, but the validation accuracy keeps growing up to 19th epoch when a validation accuracy of 98.16% occurs.

H. Indoor Segmented dataset

The performance of models trained with segmented images and the original unsegmented version of the same images was compared. For the PlantVillage indoor laboratory dataset a model was trained on the original unsegmented version of the images using transfer learning with Inception-ResNet-v2. The model was trained for ten epochs before fine tuning. The test accuracy obtained was 94.38%. The model was trained for another ten epochs with fine tuning. The test accuracy obtained was 98.37%. Another model was trained on the

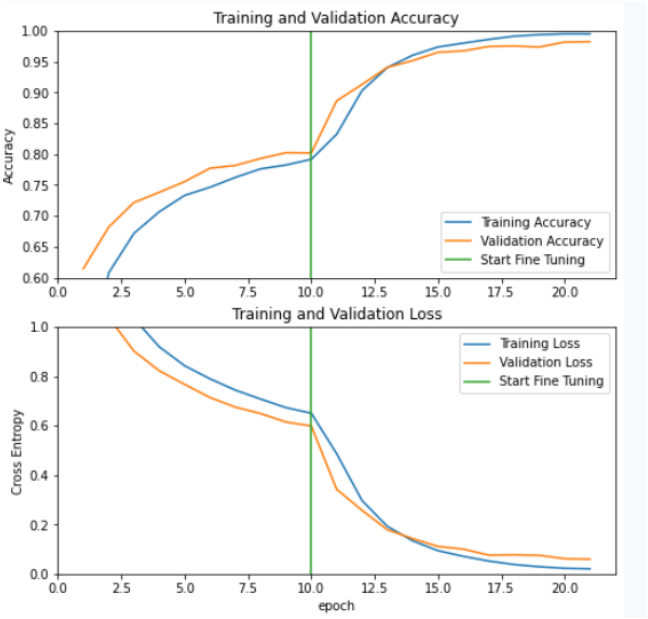

Fig. 6. Model MobileNet v.2 with Adam Optimizer Combined dataset for Accuracy and Cross Entropy.

segmented version of the same images using transfer learning with Inception-ResNet-v2. The model was trained for ten epochs before fine tuning. The test accuracy obtained was 92.50%. The model was trained for another ten epochs with fine tuning. The test accuracy obtained was 96.88%. The model trained on the original unsegmented data gave better test accuracy. In [23] for the PlantVillage dataset experiments are run on the segmented leaves. Segmentation removes any background information. In their experiments models trained with segmented images are found to give worse performance than those trained with the original unsegmented images. While we are using a pretrained model for transfer learning which is different from the pretrained models in their work our results are consistent with models trained on the original unsegmented images giving better performance. It is possible that some inherent bias is introduced into the dataset due to the background information.

IV. Results and Evaluation

A. Metrics used

To evaluate the performance of the proposed models the metrics to be used are accuracy, loss, precision, recall and area under the receiver operating characteristic curve (AUC). Tensorflow provide a series of metrics tthat can be extracted from the model,

- Accuracy is defined as the number of samples that the model labels correctly divided by the total number of samples, it gives general idea of the total performance of the model, values closer to 1 are expected for models performing well.

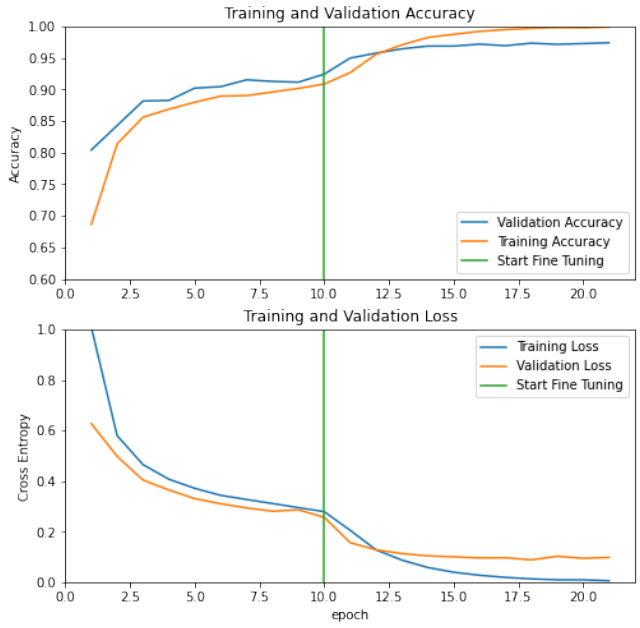

Fig. 7. For model trained on the PlantVillage indoor laboratory data segmented images with transfer learning using the pretrained Inception-ResNet-v2 CNN with fine tuning training and validation accuracy are plotted against epoch number in the upper panel. In the lower panel the training and validation loss are plotted against epoch number.

- Loss, the loss function used on the models is categorical cross entropy. It is defined by the below equation,

$$Loss = -\sum_{i=1}^{n} y_i - \log \hat{y}_i \tag{1}$$

because $\hat{y}_i$ is one when the event is predicted multiplication provides a 0 value therefore lower values are expected from better models, where the predictions are right.
- Precision calculates the ratio of the true positives by all positives across all classes as this is not a binary problem, it is defined as follows,

$$Precision = \frac{\sum_{c=1}^{Classes} TP_c}{\sum_{c=1}^{Classes} (TP_c + FP_c)} \tag{2}$$

it is also a value that ranges from 0 to 1 and the strongest model should provide a value closer to 1.
- Recall is the ratio of the true positives in each class by the true positives and false negatives, defined below,

$$Recall = \frac{\sum_{c=1}^{Classes} TP_c}{\sum_{c=1}^{Classes} (TP_c + FN_c)} \tag{3}$$

A stronger model should show a value closer to 0.
- AUC, the area under receiver operating characteristic (ROC) curve, the ROC plots the True positives rate (in y axis) with the False Positive Rate (in the x axis), intuitively, if the ratios are the same the ROC is the bisectrix of the first quadrant, and the AUC is 0.5, in the best case the True Positive ratios are much better than the false ratios, so the ROC curve bends towards the y axis given a AUC closer to 1. Bette performing models are expected to have an AUC value close to 1.

The results shown in table III come from the model previously defined in methodology section Model Indoor dataset. The model has been trained for 17 epochs in total, 10 where only the dense layer was trainable and 7 more epochs with 54 layers more trainable and a learning rate reduced by a factor of ten. The model performs satisfactory over the Indoor test subset producing accuracy over 98.1%, almost as good as the one produced by the validation subset in the model selection phase (98.5%).

TABLE II
INDOOR MODEL

Metrics Used	Indoor Model	
	Indoor	***Outdoor***
loss	0.058	9.788
accuracy	0.981	0.075
precision	0.982	0.083
recall	0.978	0.075
auc	0.999	0.537

[a]Over Indoor test and Outdoor test subsets.

The model has been used over test Outdoor subset for purely comparative purposes and there is not any expectation in terms of performance in this instance.

Included in Table III for the model trained on the augmented outdoor data with MobileNet and the Adam optimiser for 14 epochs are evaluation metrics obtained by applying the model to the outdoor test set and the indoor test set. The training accuracy and validation accuracy values for epoch 14 were 91.46% and 91.28% respectively. After epoch 14 the model starts to overfit the training data. From Table x it can be seen that the accuracy obtained on the indoor test set is 23.27%, higher than the accuracy obtained on the outdoor test set 9.17%. The precision, recall and AUC values for the indoor test set at 0.2779, 0.1945 and 0.6798 respectively are also higher than the precision, recall and AUC values for the outdoor test set at 0.1087, 0.0833 and 0.5952 respectively.

TABLE III
OUTDOOR MODEL

Metrics Used	Outdoor Model	
	Indoor	***Outdoor***
loss	4.92	3.70
accuracy	0.0917	0.2327
precision	0.1087	0.2779
recall	0.0833	0.1945
auc	0.5952	0.6798

[a]For the model trained on the augmented outdoor data.

The model has been trained for 19 epochs, 10 with the training layer being only the dense layer and a learning rate of 0.001 and for 9 epochs on the fine-tuning phase, meaning that has 56 training layers, the ones in the dense layer plus 54 from the convolutional layers. The model produced is applied

then to both testing subsets, one from the indoor model that contains 4026 samples and another from the outdoor subset that contains 120 items. There is a superior performance when the test in conducted over the indoor test subset, at 98% virtually the same precision and a recall of 97.3%. The area under the curve is almost total at 99.9%. Comparing this outstanding performance with the performance achieved with the outdoor test subset, with accuracy and recall at 9.2%, precision 9.8%, the area under the curve at slightly over the half bring to two conclusions. First, that the performance in outdoor subset is basically what it can be achieved randomly, taking into consideration that there are 13 classes of. Second, that the performance is not improving by including more images in the training set.

TABLE IV
COMBINED MODEL

Metrics Used	Combined Model	
	Indoor	***Outdoor***
loss	0.066	9.024
accuracy	0.98	0.092
precision	0.979	0.098
recall	0.973	0.092
auc	0.999	0.542

[a]Over Indoor test and Outdoor test subsets..

There is a superior performance when the test in conducted over the indoor test subset, at 98% virtually the same precision and a recall of 97.3%. The area under the curve is almost total at 99.9%. Comparing this outstanding performance with the performance achieved with the outdoor test subset, with accuracy and recall at 9.2%, precision 9.8%, the area under the curve at slightly over the half bring to two conclusions. First, that the performance in outdoor subset is basically what it can be achieved randomly, taking into consideration that there are 13 classes of. Second, that the performance is not improving by including more images in the training set.

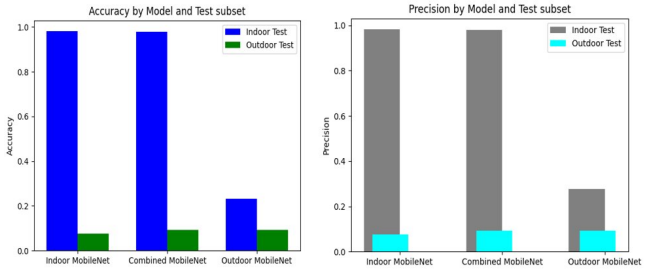

Fig. 8. Accuracy and Precision for the three models and tested on the indoors and outdoors test subsets.

Of the models trained on the indoor, outdoor and combined data for the indoor test set the best performance was obtained with the indoor model with an accuracy of 98.1%, as shown in figure. 8. The performance of the combined model gives similar accuracy of 98.0%. The performance of the outdoor model in classifying the indoor test data is poor with an accuracy of 23.3%. For the outdoor test set both the outdoor and combined models give the best classification performance with an accuracy of 9.2%.The model trained on indoor data gives an accuracy of 7.5% for the outdoor test set. From this we can see that for convolutional neural network models to be useful in detecting plant disease in real environments images of leaves taken outdoors should be used in training the models.

For all three models the classification performance on the outdoor test set is poor. Considering there are 13 classes one would expect a random classifier to give an accuracy of 7.7%. This poor performance may be due to the limited number of images in the outdoor test set which contains 120 images in total.

Fig. 9. Sample images left belongs to the Outdoor test and the right to the indoor test subset.

In Fig. 9 a sample of images from the PlantVillage dataset for the thirteen classes considered in this study are shown. All images are taken in laboratory conditions with a similar plain background.

In Fig.10 a sample of images based on images from the Plant Doc dataset for the thirteen classes considered in this study are shown. Various transformations have been applied to the original images for the purpose of augmentation. The original images were taken outdoors in real environments.

V. CONCLUSION AND FUTURE WORK

In answer to the research question, in this study it was shown that the models trained and tested using images produced in a controlled environment, in this case the laboratory, with consistent light and subject size produced satisfactory results, either the original coloured and the segmented images with different transfer functions (MobileNet and InceptionResNet). The size of the data set was large enough to guarantee good performance. On the other hand, the limitations in size of the data set in the case of the outdoor samples greatly diminished the accuracy of the models extracted from the

Fig. 10. Sample images left belongs to the Outdoor test and the right to the indoor test subset.

sample. It was observed that there was not any gain by combining the two data sets in terms of performance over the outdoor test subset. It can be argued that combining both data sets to create a model that does not perform better than the models obtained from individual data sets is not efficient, as the resources needed are significantly larger without significant improvement. It has been shown that the biggest limitation to a successfully performing model is the quality and size of the data sets

The lack of availability of images of leaves of diseased plants obtained outdoors in real world environments limited this study. Given the disparity in quality of outdoor images, pictures are taken from different distances, different exposures, light, and they are generally inconsistent. A larger dataset of outdoor images would be required to be able to produce a successful model for leaf disease classification. Also, in this study only three plant species were considered. In future work classification of plant disease from images could be extended to include additional plant species.

This study could be extended in a number of ways including through transfer learning using other pretrained CNN models to carry out image classification. In addition to the data augmentation techniques applied in this study other techniques could also be used including creating synthetic images of plant leaves using conditional generative adversarial networks. The Plant Doc outdoor images of leaves could be segmented and models, trained using these segmented images, compared against those trained using coloured versions of those images. We could extend this study from classifying the disease from the image of the leaf to grading how diseased the leaf is. In addition, we could compare the classification performance of the CNN models with that of other machine learning techniques. Our model could also be included as part of a pesticide prescription system for plant diseases. We could also compare the performance of a CNN model trained from scratch for classifying plant disease from leaf images against that of a CNN model trained by transfer learning. Our model could be used as part of a system to classify plant disease from images obtained using unmanned aerial vehicles. We could develop a lightweight model for devices with limited resources that still provides accurate performance in classifying plant disease from leaf images. The effect of varying the number of images used to train the models on the accuracy of classification could be investigated in future work.

REFERENCES

[1] M. A. Hedjazi, I. Kourbane, and Y. Genc, "On identifying leaves: A comparison of cnn with classical ml methods," in *2017 25th Signal Processing and Communications Applications Conference (SIU)*. IEEE, 2017, pp. 1–4.

[2] A. Cunha, A. Pochet, H. Lopes, and M. Gattass, "Seismic fault detection in real data using transfer learning from a convolutional neural network pre-trained with synthetic seismic data," *Computers & Geosciences*, vol. 135, p. 104344, 2020.

[3] G. Liu, J. Peng, and A. A. A. El-Latif, "Sk-mobilenet: A lightweight adaptive network based on complex deep transfer learning for plant disease recognition," *Arabian Journal for Science and Engineering*, pp. 1–15, 2022.

[4] W.-Y. Ong, C.-W. Too, and K.-C. Khor, "Transfer learning on inception resnet v2 for expiry reminder: a mobile application development," in *International Conference on Mobile Web and Intelligent Information Systems*. Springer, 2021, pp. 149–160.

[5] L. Herath, D. Meedeniya, M. Marasingha, and V. Weerasinghe, "Autism spectrum disorder diagnosis support model using inception v3," in *2021 International Research Conference on Smart Computing and Systems Engineering (SCSE)*, vol. 4. IEEE, 2021, pp. 1–7.

[6] S. Tammina, "Transfer learning using vgg-16 with deep convolutional neural network for classifying images," *International Journal of Scientific and Research Publications (IJSRP)*, vol. 9, no. 10, pp. 143–150, 2019.

[7] A. A. Mallouh, Z. Qawaqneh, and B. D. Barkana, "Utilizing cnns and transfer learning of pre-trained models for age range classification from unconstrained face images," *Image and Vision Computing*, vol. 88, pp. 41–51, 2019.

[8] F. Baldassarre, D. G. Morín, and L. Rodés-Guirao, "Deep koalarization: Image colorization using cnns and inception-resnet-v2," *arXiv preprint arXiv:1712.03400*, 2017.

[9] Y. Bhatia, A. Bajpayee, D. Raghuvanshi, and H. Mittal, "Image captioning using google's inception-resnet-v2 and recurrent neural network," in *2019 Twelfth International Conference on Contemporary Computing (IC3)*. IEEE, 2019, pp. 1–6.

[10] J. Wang, X. He, S. Faming, G. Lu, H. Cong, and Q. Jiang, "A real-time bridge crack detection method based on an improved inception-resnet-v2 structure," *IEEE Access*, vol. 9, pp. 93 209–93 223, 2021.

[11] X. Wan, F. Ren, and D. Yong, "Using inception-resnet v2 for face-based age recognition in scenic spots," in *2019 IEEE 6th International Conference on Cloud Computing and Intelligence Systems (CCIS)*. IEEE, 2019, pp. 159–163.

[12] N. Dominic, T. W. Cenggoro, A. Budiarto, B. Pardamean *et al.*, "Transfer learning using inception-resnet-v2 model to the augmented neuroimages data for autism spectrum disorder classification," *Commun. Math. Biol. Neurosci.*, vol. 2021, pp. Article–ID, 2021.

[13] B. Espejo-Garcia, N. Mylonas, L. Athanasakos, S. Fountas, and I. Vasilakoglou, "Towards weeds identification assistance through transfer learning," *Computers and Electronics in Agriculture*, vol. 171, p. 105306, 2020.

[14] Ö. İnik, K. Uyar, and E. Ülker, "Gender classification with a novel convolutional neural network (cnn) model and comparison with other machine learning and deep learning cnn models," *Journal Of Industrial Engineering Research*, vol. 4, no. 4, pp. 57–63, 2018.

[15] M. Roopak, G. Y. Tian, and J. Chambers, "Deep learning models for cyber security in iot networks," in *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)*. IEEE, 2019, pp. 0452–0457.

[16] D. T. Bui, P. Tsangaratos, V.-T. Nguyen, N. Van Liem, and P. T. Trinh, "Comparing the prediction performance of a deep learning neural network model with conventional machine learning models in landslide susceptibility assessment," *Catena*, vol. 188, p. 104426, 2020.

[17] P. Sharma, Y. P. S. Berwal, and W. Ghai, "Performance analysis of deep learning cnn models for disease detection in plants using image segmentation," *Information Processing in Agriculture*, vol. 7, no. 4, pp. 566–574, 2020.

[18] S. Z. M. Zaki, M. A. Zulkifley, M. M. Stofa, N. A. M. Kamari, and N. A. Mohamed, "Classification of tomato leaf diseases using mobilenet v2," *IAES International Journal of Artificial Intelligence*, vol. 9, no. 2, p. 290, 2020.

[19] Y. Xiong, L. Liang, L. Wang, J. She, and M. Wu, "Identification of cash crop diseases using automatic image segmentation algorithm and deep learning with expanded dataset," *Computers and Electronics in Agriculture*, vol. 177, p. 105712, 2020.

[20] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant disease classification: A comparative evaluation of convolutional neural networks and deep learning optimizers," *Plants*, vol. 9, no. 10, p. 1319, 2020.

[21] A. Ali. (2022) Plantvillage dataset in kaggle. [Online]. Available: https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset

[22] A. H. Uddin. (2022) Plant doc dataset in kaggle. [Online]. Available: https://www.kaggle.com/datasets/abdulhasibuddin/plant-doc-dataset

[23] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in plant science*, vol. 7, p. 1419, 2016.