Check for updates

Weed Identification in Plant Seedlings Using Convolutional Neural Networks

Samuel Damilare[1], Chika Yinka-Banjo[1](✉), and Olasupo Ajayi[2]

[1] Department of Computer Science, University of Lagos, Lagos, Nigeria
cyinkabanjo@unilag.edu.ng

[2] Department of Computer Science, University of the Western Cape, Cape Town, South Africa

Abstract. Agriculture is essential for the continuous survival of man, however, the adverse effect of weeds in agronomy cannot be ignored. These weeds compete with crops for nutrients and sunlight, hence resulting in low crop yield. It is therefore necessary to identify and remove them at an early growth stage for effective weed control and maximum farm produce. This study focuses on distinguishing between crops and weeds at their infancy, using images processing. To achieve this, three convolutional neural networks (CNNs) architectures, ResNet, MobileNet and InceptionV3, were evaluated using a transfer learning technique on a dataset of 5,339 RGB plant images containing 12 different species of plants. Comparing their performances from experiments carried out, the results revealed the Inception V3 model as the best for crop identification with an accuracy of 82.4%, while ResNet and Mobilenet both achieved average accuracies of 71.1% and 75.4% respectively. ResNet however gave the best performance in terms of identifying weeds. Overall, Inception v3 was the best as other performance metrics including the recall, precision, and F1-score also corroborated the superiority of Inception v3 in distinguishing between crops and weed.

Keywords: Precision Agriculture (PA) · Deep Learning · Convolutional Neural Network (CNN) · Weed Control · Plant Seedling Classification · Transfer Learning

1 Introduction

Food is a necessity for human survival. The continuous increase in world population and corresponding increase in demand for food, places immense strain on farmers. There is the need to meet up with the ever-increasing demand of food while maximizing profits. According to reports from Food & Agriculture Organization of the United Nations (FAO) [1], the population of the world is estimated to be 7.7 billion and this is expected to grow to almost 10 billion by the year 2050. To feed this population, the FAO estimates that global food production must increase by at least 70%, moreover, food production in developing countries must be at least twice that of the developed countries [1]. Since the challenges before farmers goes beyond meeting the demands of the increasing human populace to maximizing their farm produce, there is an important need to eliminate or

© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2023
Published by Springer Nature Switzerland AG 2023. All Rights Reserved
M. Masinde and A. Bagula (Eds.): AFRICATEK 2022, LNICST 503, pp. 206–224, 2023.
https://doi.org/10.1007/978-3-031-35883-8_14

reduce weeds in farmlands, as they can cause up to 100% loss of farm produce. Weeds are simply undesired plants that compete with crops for land, sunlight, water, and other resources. According to [2], weeds are said to be any unwanted biological vegetation or plant which interferes with man's activities and goals. Weeds are one of the most cogent and significant factors affecting agricultural production, as they affect crop productivity and sometimes harm livestock [3].

Smart farming is key to the future of agriculture. It can be referred to as the use of modern technologies to increase the quantitative and qualitative output of agricultural products. In recent years, Artificial Intelligence (AI), Machine Learning (ML) and technologies of the fourth industrial revolution (4IR) have been applied in agriculture to improve crop quality and yield. The impact of the 4IR on the agricultural sector has led to the birth of terminologies such as Precision Agriculture or Precision Farming or Smart Agriculture. Precision agriculture (PA) enabled farmers to capture and analyze data related to their farms, using technologies such as Global Positioning System (GPS), sensors, weather tracking etc. [1, 6]. The goal of PA is to ensure sustainability, profitability, and protection of the farming environment. PA leverages on several technologies including software applications (mobile and web), robots, drones, and cloud computing. With the help of PA, farmers can automate irrigation and harvesting processes, easily identify and control weeds, as well as get powerful insights about farmlands and produce that can help them make informed decisions.

A standard agricultural practice is the use of chemicals to improve farm output and crop yield. The use of herbicides constitutes about two-third of the chemical applications to agricultural lands globally. However, despite the use of modern techniques and changes in the composition of these chemicals, there are growing concerns, both biologically and environmentally, of the use of these chemicals on crops. Recent studies have shown that glyphosate, which is a prevalent herbicide in use since the early 1970s, contains harmful carcinogenic toxicities that are dangerous to man [4]. There is therefore a need for a better and more balanced approach in the use of herbicide on crop fields. Coupled with this challenge of using chemicals in farmlands, farmers encounter the problem of distinguishing between weeds and native plant species. Traditionally, the differentiation is done by farm personnel while walking or driving around the farm. This process is difficult and strenuous due to the resemblance and/or similarities between the weeds and some crops, particularly at the early stages of growth. Recent studies have applied PA to solve this challenge. One of such study is [4], where PA was deployed for weed identification using field mapping. Field mapping assumes that the crops are planted in rows, hence uses line detection to classify crops by assuming that plants outside the seedling lines are weeds. Similarly, in [5], three models (4 convolution layers, 6 convolution layers, 8 convolution layers, and 13 convolution layers) were built to identify the weeds that grow alongside crops. The network with 8 convolution layers resulted in the highest accuracy of 97.83% for training and 96.53% as test accuracy.

It is important to know that the successful cultivation of plants in large numbers is directly proportional to the weed control efficacy, especially as weeds compete with the crops for space, nutrients, and water during the first eight weeks of seeding, i.e., the critical period. Weed identification is therefore pertinent for ample crop yield. Image analysis or "computer vision" can be impactful in this regard, as it can precisely identify weeds with minimal negative impacts to the crops or the environment. This paper aims to investigate the efficacy of image analysis models, specifically convolutional neural network (CNN), in distinguishing weeds from crop seedlings. The contributions of this paper are to explore the application of CNN in distinguishing between weeds and crop seedlings based on their images, and to compare the performance of three CNN models i.e., ResNet, MobileNet v2 and Inception v3 for the above task as recommended.

The remainder of this paper is organised as follows, related works in literature are reviewed in Sect. 2, while Sect. 3 presents the research methodology. Section 4 discusses the experimental setup and obtained results, while Sect. 5 concludes the paper and provides suggestions for future work.

2 Related Works

Over the years, there have been several applications of computer vision techniques to solve classification issues at various levels. In [6], the authors reviewed the application of drones in agriculture, including their use in crop monitoring, irrigation, and weed control through identification and spot spraying. A unique advantage of drones in spot spraying is their ability to cover a large area in a short amount of time. Once images are captured using cameras mounted on drones or from satellite images, image detection algorithms are then used to distinguish crops from weeds. For instance, in [7] the authors used histograms based on color indices to distinguish between soybean, soil, and broadleaf (weeds). The representation of the features was tested with Support Vector Machine (SVM) and Back-propagation Neural Network (BPNN) and obtained total accuracy of 95% and 96% respectively. The authors in [12] aimed at ascertaining the growth levels of paddy crop while applying CNN to the paddy dataset which was gotten from FAO repository. According to them, paddy should be well monitored to help farmers know when to water, when to harvest and know the growth level of the paddy. They obtained an 82% accuracy with their used metrics which was better than SVM.

Though supervised ML algorithms such as SVM and Random Forest (RF) have been successful in the past, most recent research work favour deep learning and neural networks. [11] presented a deep learning algorithm that could perform image segmentation and classification. They used a convolutional network to separate maize plants from the weeds in real time. The network performance was then analyzed with various models such as LeNet, AlexNet, cNet and sNet using metrics such as processing time and accuracy. cNet performed the best and had huge potential for autonomous weed control in real world systems. Likewise, authors presented in [9] displayed a system that autonomously detected milkweed plants by placing cameras on top of vehicles. The authors used faster region-based CNN and aggregated channel features (ACF). While the latter was used on embedded systems with central processing units for running the detectors, the former was used with ResNet and graphics processing units for optimized processing. These detections were mainly used to estimate the milkweed plant densities in geo-referenced areas, which were dependent on the GPS point that corresponds to the recorded images. In [10] a pipeline based on deep learning was developed that localized and found the total number of agricultural pests in various images. A combination of Zeiler & Fergus model, region proposal network, and non-maximum suppression were used to handle overlapping detections.

The researchers in [8] focused on an agricultural robotics system that addressed weeding problem by means of selective spraying or mechanical removal of the detected weeds. They described a deep learning-based method that enabled a robot to perform an accurate weed/crop classification using a sequence of two Convolutional Neural Networks (CNNs) applied to RGB images. The first network was based on an encoder-decoder segmentation model and performed a pixel-wise, plant-type agnostic segmentation to distinguish crops from soil. The second network was then used to classify crops from weeds. Similarly, in [13] the authors investigated the influence of input image resolution on the classification performance and proposed a patch augmentation strategy. Radhika et al. proposed the classification of paddy and weed using colour features in [14], while [15] studied different tools and techniques which are necessary for the assessment of weeds development using four major procedures, i.e., pre-processing, segmentation, feature extraction and classification. Ref. [16] implemented an Encoder-Decoder based architecture for weed classification.

In [17], a field weed and crop classification algorithm based on CNN and SVM was proposed. The model used VGG16 + SVM for classification and achieved an accuracy of 96.4%. Ref. [18] also used a CNN model to differentiate between weeds and crops on a farm field. The work then suggested the best type of herbicide to apply based on the classification result. Ref. [3] used a combination of CNN and Long-Short-Term Memory (LSTM) for the identification and classification of weed plants. The CNN had a unique structure to get discriminative features for the input images, while the LSTM allowed to jointly optimize the classification. To validate the proposed model, nine species of weeds were classified using the proposed method including vine weeds, three-leaf weeds, spiky weeds, and invasive creeping weeds. After several extensive experiments, they achieved an average classification accuracy of 99.36%.

From the research papers surveyed, it was observed that there are different approaches to achieving plant image recognition and weeds/crops classification. While some researchers tend to segment the images before classification, others-built hybrid models from ground up. Moreover, most authors used datasets containing RGB images of plants obtained from online repositories or captured from drone mounted cameras or mobile phones.

This paper aims to investigate and implement a ResNet model similar to that proposed in [1], then compare it with two models using plants image dataset.

3 Materials and Methods

In this paper, the power of CNN is adopted and applied to classify weeds and crops on a plant-seedling image dataset. This work builds on the work done in [1] by comparing the proposed ResNet model with two CNN models i.e., MobileNet and Inceptionv3. In this section, a brief description of the architectures used is done before an overview of the methodology is given.

3.1 System Architecture

a. ResNet Overview. The Residual Network as presented in [19] by some researchers at Microsoft allows for successful training of networks with hundreds of layers. This was very difficult hitherto the novel architecture came to existence as previous models suffered from the problem of vanishing gradients. Vanishing gradient can simply be summarized as a problem that occur when the gradients of an artificial neural network (ANN) become smaller as they are being applied to the previous layers during back propagation. This will in turn influence the performance as well as the accuracy of the network. A neural network's depth is paramount to performing visual recognition tasks. Unlike a traditional neural network (see Fig. 1), ResNet works by using "skip connections" to connect components in different layers of the network to an identity mapping as illustrated in Fig. 2. This essentially feeds the original input to the output, thus allowing easy flow of information from previous layers to later layers in the network. The flow of information can also be sent through alternate paths if the need arises.

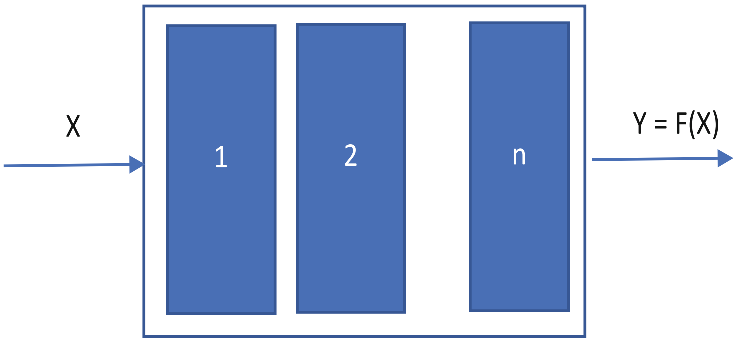

Fig. 1. Classic Neural Network Architecture

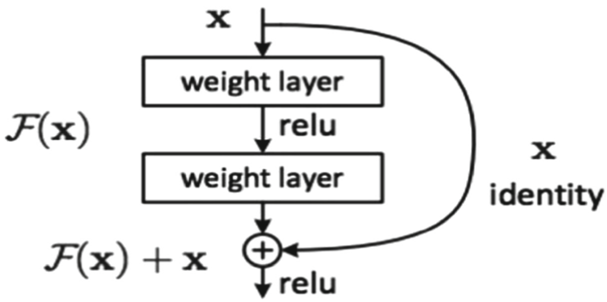

Fig. 2. Basic building blocks of a residual module

The residual learning building block in Fig. 2 is defined in Eq. 1

$$y = F(x_i\{W_i\} + x) \tag{1}$$

where input (x) and output (y) are vectors of the considered layer, while the function $F(x, \{W_i\})$ depicts the residual mapping that the model needs to learn.

b. MobileNet Overview. The MobileNet V2 was developed by Howard *et al.* [20] and was pre-trained using the ImageNet dataset (with about 1.4 million images and

1000 categories of web images). They were built as a unit of mobile-first models for computer vision for Tensorflow [21]. MobileNet aims for effectiveness and maximum accuracy while being frugal on resources, as it is meant to run on embedded or mobile devices. MobileNet can be used for classification, segmentation, and detection tasks. The architecture works based on a depth-wise separable convolutions. Unlike traditional CNNs which adds filters and the input into the next class of outputs in one stride, MobileNet's convolution is divided into two layers i.e., a 3x3 depth-wise convolution and a 1x1 pointwise convolution. Equation 2 shows a depth-wise convolution which has a single filter per channel:

$$\boldsymbol{G}K, i, n = \sum\nolimits_{ij,m} \boldsymbol{K}i, j, m.\boldsymbol{F}k + i - 1, l + j - 1, m \tag{2}$$

where K represents the depth-wise kernel with magnitude Dk x Dk x M and the m^{th} filter in the kernel is used on the m^{th} channel of **F**, which gives an output of m^{th} channel with a filtered feature map **G**. The cost of computation is given in Eq. 3 as:

$$Dk.Dk.M.DF.DF \tag{3}$$

The depth-wise convolution does not combine the input channels but rather filters them. For it to bring new attributes, an extra layer which computes progressively the output of the pointwise (1×1) and depth-wise convolution is done. The combination of these two convolutions is known as Depth-wise Separable Convolution and is highly efficient compared to the standard convolution. The computation is shown in Eq. 4 defined as:

$$Dk.Dk.M.DF.DF + M.N.DF.DF \tag{4}$$

It is paramount to note that the Rectilinear Linear Activation Function (ReLu) and batch normalization (BN) are applied after each convolution. While there are other activation functions, these two were selected as they can solve the vanishing gradient problem. The ReLu is simply a non-linear function that returns a value as input directly, or the value 0.0 if the given input is 0.0 or negative. It is given as:

$$f(x) = \max(0, x) \tag{5}$$

Figure 3 shows a standard convolutional layer using BN and ReLU on the left, and a MobileNet convolutions with the depth-wise layers and pointwise layers respectively followed by BN and ReLU.

c. Inception Overview. Also known as GoogleNet was developed 2014 during ImageNet visual recognition challenge. It uses a 1×1 convolution technique in the middle of the architecture and global average pooling to create deeper neural networks. The idea behind the inception model is to set up a deep NN while reducing the outputs. In a typical deep learning network, certain operation(s) need to be performed on each layer, such as addition of a pooling layer, a convolution operation, or filter size adjustment. The inception model allows all three to be performed in parallel. Though this would normally lead to an extreme large output, the introduction of the 1 x 1 convolution in

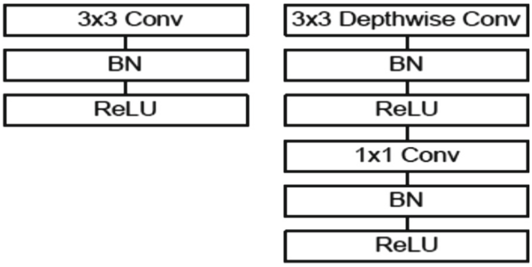

Fig. 3. Illustration of neural net layers using ReLu and Batch normalization.

the layer before the 3 × 3 and 5 × 5 layers solves this problem. Thus, providing a form of dimension reduction in the numbers of output to be passed to the next layer [22].

The architecture is built progressively in steps of factorized and the smaller convolutions. The factorized convolutions help to decrease the inefficiency of computations, as it reduces the size of parameters involved in each network. It also serves as a guard on the efficiency of the network. Furthermore, by replacing bigger convolutions with smaller convolutions, there is an increment in the speed of training. For instance, a 5 × 5 filter consist of 25 parameters, but if two smaller 3 × 3 filters are used instead of the 5 × 5 filter, the parameters are reduced from 25 to 18 (3*3 + 3*3) (Fig. 4).

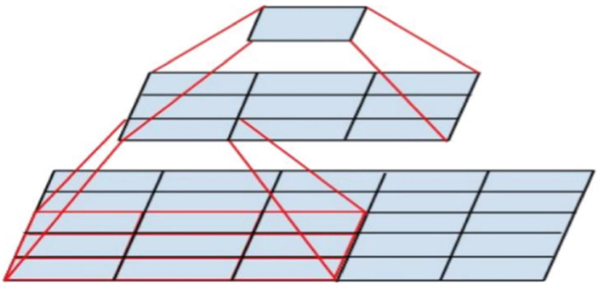

Fig. 4. Mini-network replacing the 5 × 5 convolutions

3.2 System Methodology

This subsection discussed the methodology applied in this work, including data collection, processing, transfer learning and tools used, as illustrated in Fig. 5.

i. Data Collection. The need for a large data cannot be overestimated when performing deep learning tasks. This is because it aids the neural network (NN) to better learn the relationships and patterns in each dataset. The dataset used in this paper contains images created from the department of Engineering and Signal processing at the University of Southern and Aarhus University. It contains a total of 5,539 images of crops and weeds seedlings. The dataset is grouped into 12 classes of plant species common in Denmark.

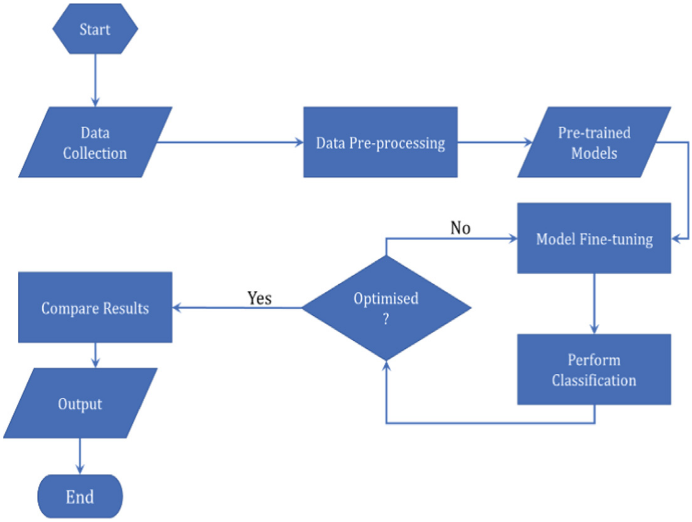

Fig. 5. Flow diagram of the proposed approach

The images are in PNG formats, and are pictures of plants of different sizes, and growth stages (Fig. 6).

ii. Data Pre-processing. The first step was to remove the non-segmented label class found in the dataset. This ensured that the total number of classes were 12, as summarized on Table 1. Other data pre-processing techniques such as transforms, resize, converting the images to float tensor were done using the PyTorch [23]. Using PyTorch's random split method, the dataset was split into 4,539 training samples and 1000 test samples.

iii. Pre-trained Model. The PyTorch framework provides out of the box deep learning pre-trained models which was utilized. A pre-trained model is a model which has been trained already on a large dataset such as ImageNet which is a very large dataset with many parameters and weights with more than 1 million labels and 1000 different categories [24]. The pre-trained models neural network models from PyTorch were utilized for the purpose of this research [25].

iv. Transfer Learning. The idea behind this is that we can use pre-trained model or an architecture which was trained differently on a particular dataset and task, then make it suitable for our own task. By doing this, we avoid having to build our NN from scratch and spending hours (or days) training it. The technique of feature extraction was used in the implementation of the project. This simply involves the reduction in numbers of resources to describe a large data. In PyTorch, the pre-trained models contain the fully connected layers. In applying it to our work, we froze the early convolutional layers of the models and trained only the last few layers which make the actual predictions or classification. We also reshaped the final layers to output twelve classes, corresponding to the number of classes of our dataset. In essence, though a pre-trained model was used, not all the layers were trained (Fig. 7).

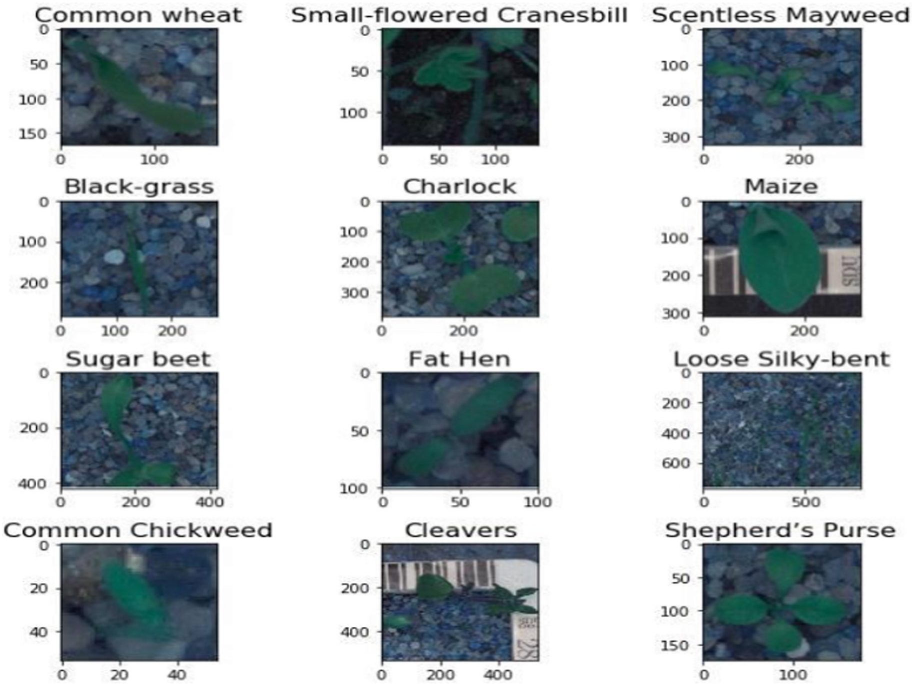

Fig. 6. Snapshot of dataset showing each plant type.

Table 1. Partitioning of plant species

Specie	Number of elements	Plant Type
Black grass	309	Weed
Charlock	452	Weed
Cleavers	335	Weed
Common chickweed	713	Weed
Common wheat	253	Crop
Fat hen	538	Weed
Loose silky-bent	762	Weed
Maize	257	Crop
Scentless mayweed	607	Weed
Shepherd's purse	274	Weed
Small-flowered cranesbill	576	Weed
Sugar beet	463	Crop
TOTAL	5539	

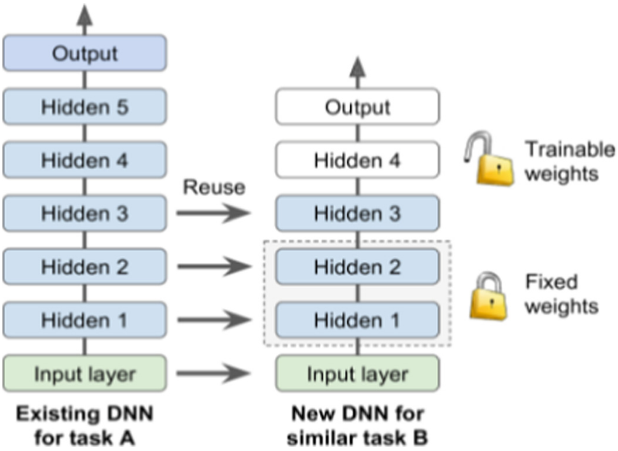

Fig. 7. An illustration of Transfer learning

3.3 Performance Metrics

Evaluating a CNN model is a major part of deep learning tasks. For the implementation done in this paper, performances were evaluated using the average accuracy, average f1-score, average recall, average precision, and the confusion matrix, and its corresponding TN (True Negative), FP (False Positive), FN (False Negative), and TP (True Positive). These metrics are described as follows:

i. Accuracy. This is the number of accurate predictions made by the model divided by the total predictions made. It is a good measure when the target variables or the target classes are nearly balanced.

$$Accuracy = \frac{(TN + TP)}{(TN + TP + FP + FN)} \tag{6}$$

ii Precision. This is a measure that tells if the number of predicted classes are correct. As an example, it determines if the proportion of plants which the model classifies as weeds are actually weeds.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

iii. Recall (or sensitivity): In the example, recall is a measure which tells the proportion of actual weeds versus those the model predicted as being weeds.

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

iv. F1_Score (or F-measure): It is the harmonic mean between the recall and the precision. It is used to get a balance between the precision and the recall.

$$F1\_Score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \tag{9}$$

4 Results and Discussion

This section presents the results, analysis, and evaluation of the models compared. As stated, a deep learning framework in PyTorch was used, while the implementation was done using Python programming language and Jupyter notebook served as the integrated development editor (IDE).

4.1 Results

Extensive experiments were carried out on the models using the PyTorch framework. The following figures shows the performance plots of the models.

ResNet Result. The results of the ResNet are as follows.

i. Accuracy and Loss: The performance plot of the ResNet model is shown in Fig. 8.

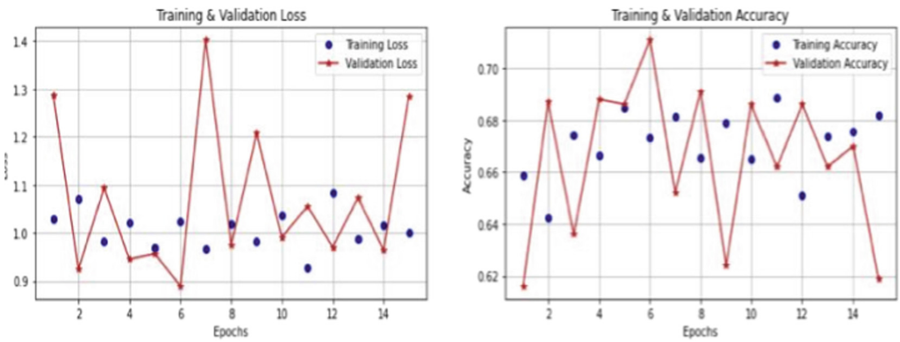

Fig. 8. ResNet performance plot

From the above plot, it can be observed that the model obtained an accuracy of 71.1% and 68.9% on the testing and training samples after 15 epochs, representing an increase of 2.2%. Similarly, the loss is shown to dropped from 1.02% in the training sample to 0.88% in the testing sample.

ii. Confusion matrix: The confusion matrix that shows the predicted classes versus the true classes is shown in Fig. 9.

The confusion matrix in Fig. 9 depicts twelve rows and columns of the actual and predicted classes respectively. The diagonal record tells the result of the true positives i.e., the correctly predicted classes.

iii. Classification Report: The classification result below shows the F1-score, recall and the precision of the model for the twelve classes of the plants.

Table 2 shows the classification report of the ResNet model using other performance metrics. Of all the 12 plants, Cranesbill, Charlock, and Maize had the highest F1-Scores at 83%, 79% and 85% respectively. The model struggled slightly with identifying weeds,

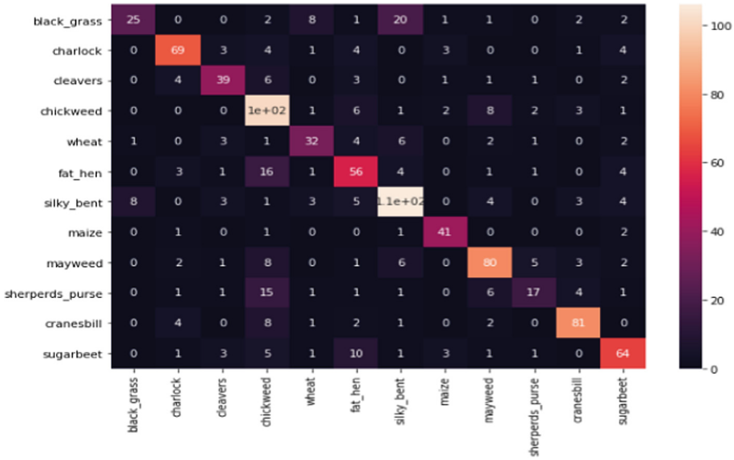

Fig. 9. Confusion matrix for ResNet mode

Table 2. Classification report for ResNet model

Class	Species/Metrics	Precision (%)	Recall (%)	F1-Score (%)
1	Black grass	74	40	52
2	Charlock	81	78	79
3	Cleavers	72	68	70
4	Chickweed	60	81	69
5	Wheat	65	62	63
6	Fat-hen	60	64	62
7	Silky-bent	72	77	75
8	Maize	80	89	85
9	Mayweed	75	74	75
10	Shepherd's purse	61	35	45
11	Cranesbill	84	82	83
12	Sugar beet	73	71	72
Average Values		71.4%	68.4%	69.2%

with the F1-Scores of Chickweed and Mayweed being 69% and 75%. There is therefore a very high probability that the model would mistake these weeds for crops.

MobileNet Result. The results of the ResNet are as follows.

i. Accuracy and Loss: The performance plot for the MobileNet is shown below (Fig. 10).

The above plot shows the performance of both training and validation samples. It is seen that the testing accuracy of the model is 75.4% improving on the 68.4% accuracy it

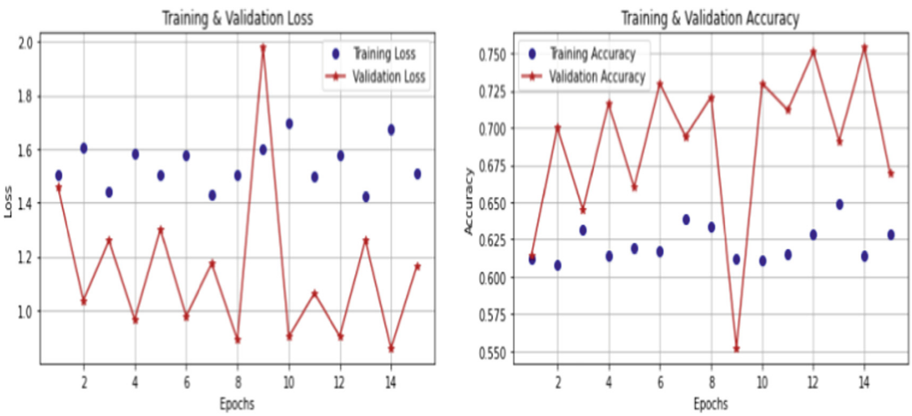

Fig. 10. Performance plot for MobileNet

achieved during training. This is a 7% increase on the training sample without overfitting. The loss accuracy also showed a progress from 1.67 to 0.61.

ii. Confusion matrix: The confusion matrix of predicted versus true values is depicted in Fig. 11.

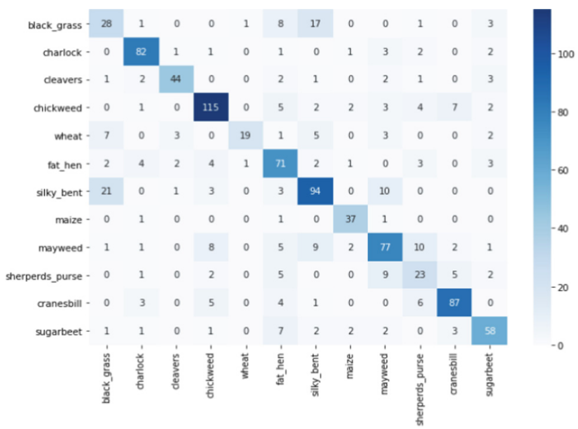

Fig. 11. Confusion matrix for MobileNet model

The confusion matrix in Fig. 11 depicts twelve rows and columns of the actual and predicted classes respectively. The diagonal record tells the result of the true positives i.e., the classes that the model correctly predicted.

iii. Classification Report: The classification result below shows the F1-score, recall and the precision of the model for the twelve classes of the plants for the model.

Table 3 shows the classification report of the MobileNet model using the performance metrics. For F1-Scores, in addition to Charlock (86%), Maize (87%) and Cranesbill

Table 3. Classification report for MobileNet model

Class	Species/Metrics	Precision (%)	Recall (%)	F1-Score (%)
1	Black grass	46	47	47
2	Charlock	85	88	86
3	Cleavers	86	79	82
4	Chickweed	83	82	82
5	Wheat	90	47	62
6	Fat-hen	63	76	69
7	Silky-bent	71	71	71
8	Maize	82	93	87
9	Mayweed	46	49	47
10	Shepherd's purse	90	96	93
11	Cranesbill	84	82	83
12	Sugar beet	76	75	76
Average values:		75.2%	73.8%	73.8%

(83%) for which ResNet performed well, MobileNet was also able to achieve good F1-scores of 93% for Shephard's purse and 82% for Cleavers. For the weeds, MobileNet performed well for Chickweed (82%), but very poorly with Mayweed at just 47%.

Inception Result. The results of the Inception model are as follows.

i. Accuracy and Loss: The performance plot for the inception is shown below.

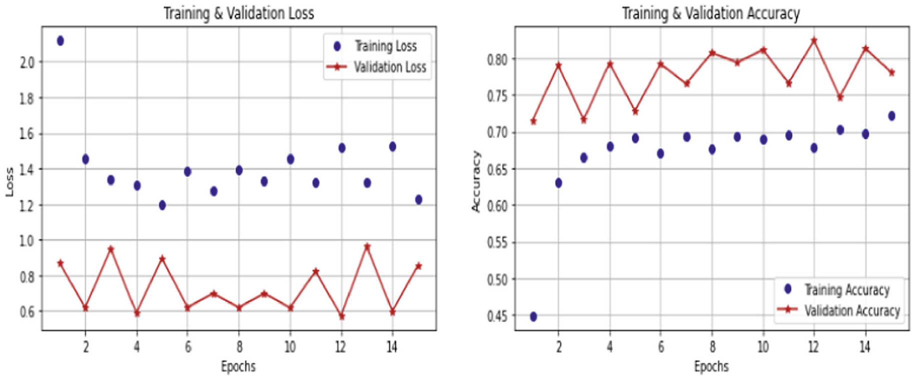

Fig. 12. Inception performance plot

Figure 12 shows testing and training accuracies of 82.4% and 67.9% respectively, while the training and testing losses were 1.5% and 0.7% respectively. This shows a significant improvement in the accuracy between the training and test sample sets. With a testing accuracy of 82.4%, Inception produced the best result of the three models.

ii. Confusion matrix: The confusion matrix showing the number of true positives in the diagonal is shown Fig. 13.

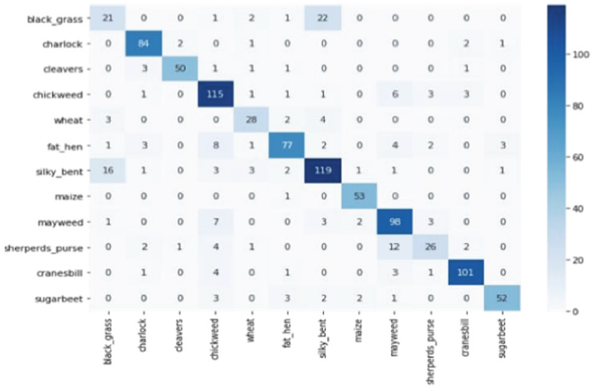

Fig. 13. Confusion matrix for the inception model

iii. Classification Report: The table below shows the classification report of the model using the metrics F1-score, precision, and recall.

Table 4. Classification report for Inception model

Class	Species/Metrics	Precision (%)	Recall (%)	F1-Score (%)
1	Black grass	50	45	47
2	Charlock	88	93	91
3	Cleavers	94	88	91
4	Chickweed	79	88	83
5	Wheat	74	76	75
6	Fat-hen	87	76	81
7	Silky-bent	78	81	79
8	Maize	91	98	95
9	Mayweed	78	86	82
10	Shepherd's purse	74	54	63
11	Cranesbill	93	91	92
12	Sugar beet	91	83	87
Average values:		81.4%	79.9%	80.5%

Table 4 shows the classification report of the Inception model using other performance metrics. Again Cranesbill, Maize and Charlock had good F1-Scores at over 90% each. Inception performed well in identifying the weeds with 83% and 82% F1-Scores for Chickweed and Mayweed respectively. Like the other models, Inception also struggled with Black grass.

4.2 Comparative Performance Evaluation

Figure 14 shows a graphical comparison of the average performance of the 3 models. It can clearly be seen that Inception performed the best across board, followed by MobileNet.

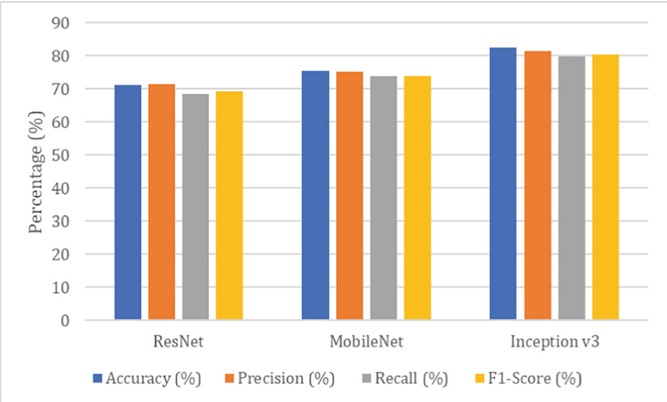

Fig. 14. Summary of Results – ResNet vs MobileNet vs Inception v3

4.3 Weed Detection

Figure 15 shows the performance of the three models in weed identification. This result is important because it shows how well the models perform in weed detection, which is vital step in weed control on farmlands. For this result we considered "Black-grass", "Chick-weed", and "Mayweed" as weeds. The graph shows the Precision, Recall and F1-Score of the three models – ResNet, MobileNet, and Inception v3.

The figure reveals that all three models performed relatively well with regards identifying Chick-Weed Inception v3, however, their performance was poor for black-grass, with only ResNet managing a precision of over 70%. A possible explanation for the poor performance w.r.t black-grass could be because of the blades / leaves of the black-grass are relatively slim compared to all other crops in the dataset. Finally, for all 3 weed samples, ResNet achieved an overall average Precision of 69.7%, followed by Inception v3 at 69% and MobileNet with a score 58.3%.

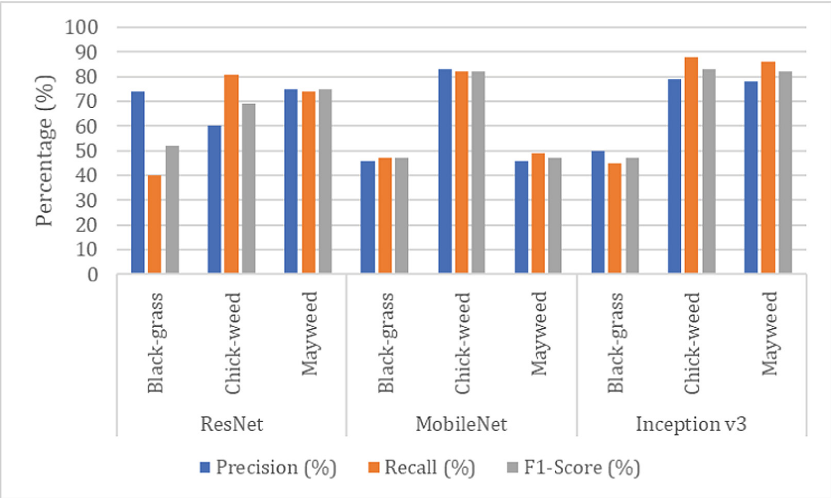

Fig. 15. Comparative Performance in Weed Detection

4.4 Discussion

In this section, inclusive arguments for the realized comparison results are summarized. From the overall results seen, it can be said that the proposed methodological system was relatively efficient and effective. Overall, ResNet seems to perform the poorest compared to the other two models while Inception is seen to have the best results of the three. The high results of the Inception could be attributed to the auxiliary classifier of the model which improve convergence during the training process. It is important to note that, though ResNet seemed to perform poorly, on closer examination, especially w.r.t to weed detection, ResNet was almost at par with Inception v3 and even outperforming it in detecting black grass.

5 Conclusions

In this paper, three CNN architectures were studied and evaluated on a plant seedlings image dataset to classify weeds from crops. The dataset which contains 9 crop plants, and 3 weed plants were in their early growth stage. It contains a total of 5539 images to which 4539 was set for training and 1000 plant images set for the test or validation samples. This dataset was divided into 12 classes of different plant species at different growth stages. Using a transfer learning technique and a popular deep learning library - PyTorch, the models which are pre-trained were evaluated on the dataset. Comparing the results of the three models, the ResNet model achieved an average classification accuracy of 71.1% on the testing sample while, the MobileNet and Inception v3 achieved accuracy scores of 75.4% and 82.4% respectively. When their performance in term of weed identification alone was considered, ResNet performed the best with a score of 69.7%, but was closely followed by Inception v3 (69%), and MobileNet (58.3%).

There is a need to develop datasets for local farm produce and crops. The curation of such datasets might be another avenue for expanding this work. Once such datasets are in place, the deployment of pre-trained models via transfer learning on localized dataset for weed control might be considered. Furthermore, the development of a web-based platform via which farmers can access the solutions might also be considered. It would be interesting to note the performance of these neural network models on embedded and mobile devices.

References

1. Binguitcha-Fare, A., Sharma, P.: Crops and weeds classification using convolutional neural networks via optimization of transfer learning parameters. Int. J. Eng. Adv. Technol. (IJEAT). **8**(5), 2249–8958 (2019)
2. Iderawumi, A., Friday, C.: Characteristics effects of weed on growth performance and yield of maize (zea mays). Biomedical J. **1**(4) (2018)
3. Wu, Z., Chen, Y., Zhao, B., Kang, X., Ding, Y.: Review of weed detection methods based on computer vision. Sensors **21**(11), 3647 (2021)
4. Asad, M., Bais, A.: Weed detection in canola fields using maximum likelihood classification and deep convolutional neural network. Inf. Process. Agric. **7**(4), 535–545 (2020)
5. Gothai, E., Natesan, P., Aishwariya, S., Aarthy T., Singh G.: Weed identification using convolutional neural network and convolutional neural network architectures. In: 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), pp. 958–965 (2020)
6. Yinka-Banjo, C., Ajayi, O.: Sky-farmers: applications of unmanned aerial vehicles (UAV) in agriculture. Auton. Veh. 107–128 (2019)
7. Petit, S., Boursault, A., Le Guilloux, M., Munier-Jolain, N., Reboud, X.: Weeds in agricultural landscapes. A review. Agron. Sustain. Dev. **31**(2), 309–317 (2011)
8. Fawakherji, M., Youssef, A., Bloisi, D., Pretto, A., Nardi, D.: Crop and weeds classification for precision agriculture using context-independent pixel-wise segmentation. In: 2019 Third IEEE International Conference on Robotic Computing (IRC), pp. 146–152. IEEE (2019)
9. Ozcan, K., Sharma, A., Bradbury, S., Schweitzer, D., Blader, T., Blodgett, S.: Milkweed (Asclepias syriaca) plant detection using mobile cameras. Ecosphere **11**(1), e02992 (2020)
10. Li, W., Chen, P., Wang, B., Xie, C.: Automatic localization and count of agricultural crop pests based on an improved deep learning pipeline. Sci. Rep. **9**(1), 1–11 (2019)
11. Dyrmann, M., Karstoft, H., Midtiby, H.: Plant species classification using deep convolutional neural network. Biosys. Eng. **151**, 72–80 (2016)
12. Neforawati, I., Herman, N., Mohd, O.: Precision agriculture classification using convolutional neural networks for paddy growth level. J. Phys. Conf. Ser. **1193**(1), 012026 (2019)
13. Brilhador, A., Gutoski, M., Hattori, L., De Souza Inácio, A., Lazzaretti, A., Lopes, H.: Classification of weeds and crops at the pixel-level using convolutional neural networks and data augmentation. In: 2019 IEEE Latin American Conference on Computational Intelligence (LA-CCI), pp. 1–6. IEEE (2019)
14. Kamath, R., Balachanra, M., Prabhu, S.: Paddy crop and weed classification using color features for computer vision-based precision agriculture. Int. J. Eng. Technol. (UAE) **7**(4), 2909–2916 (2018)
15. Mishra, A.M., Gautam, V.: Weed species identification in different crops using precision weed management: a review. In: 2021 ISIC, pp. 180–194 (2021)

16. Khan, A., Ilyas, T., Umraiz, M., Mannan, Z., Kim, H.: Ced-net: crops and weeds segmentation for smart farming using a small cascaded encoder-decoder architecture. Electronics **9**(10), 1602 (2020)
17. Haichen, J., Qingrui, C., Zheng Guang, L.: Weeds and crops classification using deep convolutional neural network. In: 2020 the 3rd International Conference on Control and Computer Vision, pp. 40–44 (2020)
18. Grace, R.: Crop and weed classification using deep learning. Turkish J. Comput. Math. Educ. (TURCOMAT) **12**(7), 935–938 (2021)
19. He, K., Zhang, X., Ren, S., Sun, J.: Microsoft Research: Deep Residual Learning for Image Recognition. arXiv preprint 2015:1512.03385 (2015)
20. Howard, A., Zhu, M., Chen, B., Kalenichenko, D., et al.: Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint 2017:1704.04861 (2017)
21. Abadi, M., Agarwal, A., Barham. P., Brevdo, E., et al.: TensorFlow: large-scale machine learning on heterogeneous systems. www.tensorflow.org. Accessed 21 June 2022
22. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: 2016 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
23. Paszke, A., Gross, S., Massa, F., Lerer, A., et al.: Pytorch: an imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
24. Deng, J., Dong, W., Socher, R., Li, L., et al.: Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
25. Cadene, et al.: pre-trained ConvNets for pytorch: NASNet, ResNeXt, ResNet, InceptionV4, InceptionResnetV2, Xception, DPN etc. (2017)