Weed Identification in Crop Field Using CNN

*Saniya Zahoor[1], Shabir A.Sofi[2]
[1]Post Graduate Department of Computer Science, University of Kashmir, Hazratbal
[2]Department of Information Technology, National Institute of Technology Srinagar
[1]saniyazahoor.ku@gmail.com, [2]shabir@nitsri.ac.in

Abstract: Since the ages agriculture has remained as the backbone of economies especially developing countries like ours, where population is growing rapidly being second most populated country in the world, food demands are increasing so, farmers need to maximize their productivity. Weed is one of the enemies to farmer's crop which competes with the crop for nutrients and sometimes hinders the growth of crop. Weed can cause loss of production ranging from 10 to 100%. There has been research on the use of many CNN models for weed identification. This paper presents a classification model to distinguish between weed and crop images and it classifies 12 species of weeds and crops. The proposed model achieves 96.45% of accuracy during training and of 90.08% during validation and testing.

Keywords: Machine Learning, Densenet Models, CNN, Regression.

1. Introduction

The growth of the population and global warming are creating new challenges to the farmers. There is a need to increase the farming production and maximize profits. According to the research done by the UN Food and Agriculture Organization the world is going to require to produce 70% more food in 2050 than it did in 2006 in order to suffice the need of food for rising population and to fulfil this demand farmers and agricultural fields are switching to IoT for enhancement of productivity, global market, less human intervention, minimum time and cost etc [2]. India's financial resources are mainly dependent upon agriculture and the important hindrance that arises in traditional farming is shifting of climatic conditions from one to another which encompasses heavy or less rainfall, intensified storms, heat waves etc and these conditions affect the yield or productivity drastically. So in order to increase the productivity and to reduce the obstacles in agriculture there is a need to use ingenious technology. Precision agriculture employs new technologies to analyze various hazardous factors that affect the crop growth.

One of the main issues in crop fields is growth of weeds. It affects the overall production of crop because it competes for nutrients, space, water, light and hinders the growth of crops in the field. Therefore, identification of weed in such crop fields plays a vital role in precision farming [2]. Many traditional methods are employed which includes spraying natural plucking or herbicides but it becomes tedious task and increases the labour cost. Hence, there is a need to have an automated system for detection of weeds in such fields. This paper presents CNN models for weed identification in crop fields. The organization of paper is as follows: section 2 discusses literature survey, section 3 proposed methodology, section 4 presents the results and section 5 gives conclusions.

2. Literature Survey

There has been a lot of work done to classify crops and weeds. Authors in [3] identified three classes: soil, soybean and broadleaf (weeds) by using histogram based on color indices and tested with two methods viz. Support Vector Machine (SVM) and Back-propagation Neural Network (BPNN) with an accuracy of 95% and 96% respectively. Several other models such as GoogLeNet, VGGNet, DetectNet have been tested as well and these showed high accuracy with high f1 score values over 95% for detecting weeds in Bermuda grass turfgrasse [4]. Further, research has been carried out in implementing CNNs with unsupervised training dataset collection for weeds detection from images captured with drones [5]. An Area-Under-Curve (AUC) over 80% was achieved while performing tests with bean and spinach.

There has been research in broad leaf weed detection in pasture [6]. The work uses Quadric- CNN and Support Vector Machine models for conducting analysis with accuracies of 89.4% and 96.88% respectively. Authors in [7] present SVM models to classify 11 plant species including weeds with a precision of 93%. Similar work has been proposed in [8] for prediction of growth stage in species of weed with an accuracy of 70%. For detecting broadleaf and grass weeds in relation to soil and soybean, the authors in [9] investigated the use of CNNs and achieved above 98% accuracy with an accuracy average between all images above 99%. Traditional machine learning algorithms and deep learning models were compared for seedling classification in the paper [14]. A good accuracy of 92.6% has been obtained by performing background segmentation. The authors in [10] demonstrated the great performance of CNNs to learn useful features representations for 44 different plant species with high precision. There has been a lot of recent research aiming to develop automated analysis of plant images [11, 12, 13, 14].

3. Proposed Methodology

Though various solutions have been proposed for this problem and some of them are very accurate, the common problem we found is they are very complex, researcher have to manually deal with problems like vanishing gradient etc. We are going to propose a solution in which we will be using data augmentation [15] to reduce chances of over-fitting and Convolutional Neural Network Architecture DenseNet-BC[16] for image classification. The reason for choosing this architecture and technology is described in subsequent sections. It is obvious to think of using Machine Learning for solving classification problem like this one, but there are various Machine Learning algorithms that fit in for this problem. So we began with basic and easy like Logistic Regression, Convolutional Neural Network and finally Dense Convolutional Neural Network (DenseNet). In the subsequent sections we will explain in detail how our observation in each one of them was. But before that let's have a look at Machine Learning.

For conducting our experiments, we are going to use a dataset containing approximately 3486 plant images constituted by Aarhus University Signal Processing group in collaboration with University of Southern Denmark [17].

Table 1.Repartition of Species

Species	**No. of Elements**	**Type**
Black-grass	279	Weed
Charlock	299	Weed
Cleavers	289	Weed
Common Chickweed	347	Weed
Common Wheat	255	Crop
Fat Hen	305	Weed
Loose Silky-bent	291	Weed
Maize	258	Crop
Scentless Mayweed	300	Weed
Shepherds Purse	275	Weed
Small-flowered Cranesbill	300	Weed
Sugar beet	300	Crop
Total	3486	

This dataset presents 12 plant species at several growth stages. It comprises annotated RGB images with a physical resolution of roughly 10 pixels per mm. The following table (table I) shows the repartition of the dataset. All the images of the dataset are in png format. The dataset will be divided into a training set and test set with the test size equal to 0.05 (See table II). Now our training set comprises of 3311 images and testing set comprises of 175 images.

Table 2. Training and Test Set

Species	Training	Test
Black-grass	265	14
Charlock	284	15
Cleavers	275	14
Common Chickweed	330	17
Common Wheat	243	12
Fat Hen	290	15
Loose Silky-bent	276	15
Maize	245	13
Scentless Mayweed	285	15
Shepherds Purse	261	14
Small-flowered Cranesbill	285	15
Sugar beet	285	15

Data augmentation techniques will be performed to diversify our training set and prevent over-fitting[16]..By applying a variety of image transforms such as flip, rotate, whitening, Zoom. It is worth noting that 5% of the training set will be considered as a validation set. This is permitted to measure the performance of the model during the training. So, the exact number of images dedicated for training is 3311 images and for validation was 175. After applying data augmentation on training set the now contains images given below:

$$\textit{No of Images} = (\textit{No of actual training samples}) * (\textit{No of epochs})$$

Figure 1: Images of each class

So, in our training set we have 3311 images and we trained our model with 30 epochs, therefore total no of images samples now are 99330. As already specified we are using Densely Connected Neural Network (DenseNet) to perform training. Images in the dataset are not uniform and there are some images of size less than 71x71 so training set images were resized to 64x64. And we are using 3 channels image i.e RGB colored images. Dataset was splitted into train and test in the ratio of 95%, though 75% is most prefered ratio but as our dataset is very small so chose the given ratio.
Before applying DenseNet for classification, Logistic Regression and Simple Convolutional Neural Network was used for training. The details of each training process are given below. Choice of Logistic Regression was made because of unavailability of sufficient dataset for training and unavailability of fast computational resources. Model details are given in Table 3.

Table 3. Logistic Regression Model Details

Solver	**L-BFGS-B**
Iterations	2000
Multiclass	ovr

Logistic Regression performance on given dataset was quite unsatisfactory, so we tried to use CNN with few conv2d layers, dropout, and max_pooling, and dense layers. The model summary is given in Figure 2 below.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 62, 62, 32)        896
_________________________________________________________________
batch_normalization_1 (Batch (None, 62, 62, 32)        128
_________________________________________________________________
dropout_1 (Dropout)          (None, 62, 62, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 60, 60, 32)        9248
_________________________________________________________________
batch_normalization_2 (Batch (None, 60, 60, 32)        128
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 20, 20, 32)        0
_________________________________________________________________
dropout_2 (Dropout)          (None, 20, 20, 32)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 12800)             0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               1638528
_________________________________________________________________
batch_normalization_3 (Batch (None, 128)               512
_________________________________________________________________
dropout_3 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 12)                1548
=================================================================
Total params: 1,650,988
Trainable params: 1,650,604
Non-trainable params: 384
```

Figure 2. CNN Model summary

Training using CNN is also very simple and it is quite fast. The hyperparameters used in training are given below in table 4.

Table 4. CNN Hyperparameter Details

Parameter	**Value**
epochs	30
Steps per epoch	100
Dropout	0.25

Performance of the CNN was also quite unsatisfactory, but better than Logistic Regression. Now DenseNet was given a try as this architecture of CNN has a great performance among the state of art. Input format of images and resize followed in this model is same as explained above. Computationally densenet is very expensive but it is transition layer which saves computation, it reduces the model size to many fold without lossing the model accuracy. The hyper parameters used in the training are given below in the table 5.

Table 5. DenseNet Hyperparameter Details

Parameter	Value
epochs	30
No of dense blocks	3
Growth rate	12
depth	16
factor	0.1
patience	2

Though training densenet took more time compared to other models but, it kept its fame of performing best among all. The results of all the experiments are described below in detail.

4. Results

Though it is very simple and fast to train, but it did not performed good on the given dataset. The test accuracy was around 11% which is quite low to be considered. The metric that was used for accuracy was 'accuracy_score'. The performance of CNN was better than the Logistic Regression. Training accuracy of 98.46% and validation accuracy of 64.19% was obtained. Though it is good to some extent with this low size dataset but still unacceptable. 'Adam' optimizer was used with 'categorical crossentropy' loss and 'accuracy' accuracy metric to evaluate the model.

```
loss: 0.0521 - accuracy: 0.9852 - val_loss: 2.0198 - val_accuracy: 0.6409
loss: 0.0435 - accuracy: 0.9871 - val_loss: 1.4169 - val_accuracy: 0.6769
loss: 0.0419 - accuracy: 0.9884 - val_loss: 2.8267 - val_accuracy: 0.5349
loss: 0.0431 - accuracy: 0.9843 - val_loss: 3.7666 - val_accuracy: 0.4751
loss: 0.0598 - accuracy: 0.9821 - val_loss: 1.2310 - val_accuracy: 0.6539
loss: 0.0439 - accuracy: 0.9846 - val_loss: 1.3748 - val_accuracy: 0.6419
```

Figure 3. Accuracy and overfitting of CNN

Early-Stopping was also used with patience of 10 to stop the training if no improvement was obtained. Figure 4 allows observing the accuracy and over-fitting of CNN model for every epoch.
DenseNet as it outperformed the state of the art architectures like ResNet etc on ImageNet and CIFAR-10, CIFAR-100, it did same here, and the accuracy of this model was highest among all. The metric used for loss was 'categorical_crossentropy' and for accuracy it was 'accuracy'. Optimizer used was Adam. Model acquired the training accuracy of 96.45% and validation accuracy of 90.08% which is much better than the rest two experiments. Following figure 5 allows to observe the accuracy and over-fitting in DenseNet model.

```
loss: 1.1622 - accuracy: 0.9706 - val_loss: 1.3116 - val_accuracy: 0.8926
loss: 1.1734 - accuracy: 0.9623 - val_loss: 1.3274 - val_accuracy: 0.8843
loss: 1.1554 - accuracy: 0.9693 - val_loss: 1.2683 - val_accuracy: 0.9008
loss: 1.1614 - accuracy: 0.9663 - val_loss: 1.3032 - val_accuracy: 0.8843
loss: 1.1533 - accuracy: 0.9684 - val_loss: 1.3063 - val_accuracy: 0.8843
loss: 1.1620 - accuracy: 0.9645 - val_loss: 1.2835 - val_accuracy: 0.9008
```

Figure 4. Accuracy and Overfitting of DenseNet Model

The performance of DenseNet model could have been further boosted by following factors:

- Increasing training samples: Due to the present situation it was not possible to collect the dataset by going into some agricultural institute, otherwise we would have collected few thousand more training samples which surely was an option to increase performance.
- Use of Transfer Learning: Since DenseNet have been trained on ImageNet and weights are available as DenseNet-121, 169, 201. But as we looked into the dataset of ImageNet it do not contain the weed and crop that we have in our dataset.
- Increaseing Image dimensions: Dense has been found performing better for 512x512 and 224x224 image dimension, since our dataset do not have all the images in uniform dimension some are like 71x71. So this option was also not possible.

5. Conclusions

An accurate deep learning model for crops and weeds classification can definitely assist farmers in maximizing crop yields and consequently minimizing the losses. In this paper, we applied Densely Connected Neural Network to classify crops and weeds. The DenseNet-Bottleneck-Compression achieved incredible results with an accuracy of 96.45% on the training set and 90.08% on the test set. The dataset contains approximately 3486 images of plants belonging to 12 species at various growth stages. The proposed approach was the result of several techniques applied successively. Firstly, we resized the images to 64x64 dimensions. Then we applied data augmentation techniques such as rotation, zooming, flipping to make more balanced the 12 classes of dataset. After that, we used a DenseNet- BC model to perform features extraction. We applied also the learning rate scheduler method to determine the most appropriate range of learning rates. We concluded that the size of input image had a significant impact on the performance of our model.

This model can be extended to other species of plants in other countries as well as other online datasets. In future work, this work can be aimed to increase further accuracy and deploy the application for farmers to use for their benefit.

REFERENCES

[1] Fao how to feed the world in 2050.
[2] http://www.fao.org/fileadmin/templates/wsfs/docs/expert_paper/How_to_Feed_the_World_in_20 50.pdf.
[3] University of Pretoria. “Important weeds in maize”. https://www.up.ac.za/sahri/article/1810372/important-weeds-in- maize.
[4] Saad Abouzahir, Mohamed Sadik, and Essaid Sabir. “Enhanced approach for weeds species detection using machine vision”. In 2018 International Conference on Electronics, Control,
[5] Optimization and Computer Science (ICECOCS), pages 1-6. IEEE, 2018
[6] [Jialin Yu, Shaun M Sharpe, Arnold W Schumann, and Nathan S Boyd. “Deep learning for image-based weed detection in turfgrass”. European Journal of Agronomy, 104:78-84, 2019
[7] M Dian Bah, Adel Hafiane, and Rapha¨el Canals. “Deep learning with unsupervised data labeling for weeds detection on uav images”. arXiv preprint arXiv:1805.12395, 2018.
[8] Wenhao Zhang, Mark F Hansen, Timothy N Volonakis, Melvyn Smith, Lyndon Smith, Jim Wilson, Graham Ralston, Laurence Broadbent, and Glynn Wright. “Broad-leaf weed detection in pasture”. In 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), pages 101-105. IEEE, 2018.
[9] Mads Dyrmann and Rasmus NyholmJørgensen. “Roboweedsupport: weed recognition for reduction of herbicide consumption”. In Precision agriculture’15, pages 259-269. Wageningen Academic Publishers, 2015.
[10] NimaTeimouri, Mads Dyrmann, Per Nielsen, SolvejgMathiassen, Gayle Somerville, and Rasmus Jørgensen. “Weed growth stage estimator using deep convolutional neural networks”. Sensors, 18(5):1580, 2018.
[11] Alessandro dos Santos Ferreira, Daniel Matte Freitas, GercinaGon¸calves da Silva, HemersonPistori, and Marcelo TheophiloFolhes. “Weed detection in soybean crops using convnets”. Computers and Electronics in Agriculture, 143:314-324, 2017.
[12] Sue Han Lee, Chee Seng Chan, Simon Joseph Mayo, and Paolo Remagnino. “How deep learning extracts and learns leaf features for plant classification”. Pattern Recognition, 71:1-13,2017.
[13] Andreas Kamilaris and Francesc X Prenafeta-Boldu. “Deep learning in agriculture: A survey”. Computers and Electronics in Agriculture, 147:70-90,2018.
[14] Konstantinos Liakos, PatriziaBusato, DimitriosMoshou, Simon Pearson, and DionysisBochtis. “Machine learning in agriculture: A review”. Sensors, 18(8):2674, 2018.

[15] Thomas Mosgaard Giselsson, Mads Dyrmann, Rasmus NyholmJørgensen, Peter Kryger Jensen, and Henrik SkovMidtiby. A Public Image Database for Benchmark of Plant Seedling Classification Algorithms. arXiv preprint, 2017
[16] Gong Cheng, Zhenpeng Li, Junwei Han, Xiwen Yao, and Lei Guo. "Exploring hierarchical convolutional features for hyperspectral image classification". IEEE Transactions on Geoscience and Remote Sensing, (99):1-11,2018.
[17] JinruXue, Sigfredo Fuentes, Carlos Poblete-Echeverria, Claudia Gonzalez Viejo, Eden Tongson, Hejuan Du, and BaofengSu. "Automated Chinese medicinal plants classification based on machine learning using leaf morpho colorimetry, fractal dimension and." International Journal of Agricultural and Biological Engineering", 12(2):123-131, 2019.