MULTIDISCIPLINARY SCIENCE JOURNAL

TECHNICAL NOTE
Published Online: May 31, 2023
https://doi.org/10.31893/multiscience.2023039

Crop image classification using convolutional neural network

Mhalsakant M. Sardeshmukh[a] | Midhun Chakkaravarthy[a] | Sagar Shinde[b] | Divya Chakkaravarthy[a]

[a]Lincoln University Malaysia, Malaysia.
[b]Dr. D. Y. Patil Institute of Technology, Pimapri, Pune, India.

Abstract A crop image classification using convolutional neural network is proposed in the paper. Classification of crop images is important and required in many applications such as yield prediction, decease detection etc. The main challenges are availability of the large dataset and extraction of meaningful features to describe a class of image. We have proposed a convolutional neural network and the pre-trained models like VGG 16 and Resnet 50 for crop image classification. The pre-trained models trained on millions of images for a very large class size. The results shows that VGG 16 can be best used for our application as gives the accuracy of more than 98 %. The CNN training accuracy is 93 % but testing accuracy is only 42%. This is due to the lack of training data available. The accuracy of the CNN can be improved using large dataset. The Resnet 50 fails for crop image classification.

Keywords: deep neural network, transfer learning, classifier

1. Introduction

Agriculture is a vital part of any economy. A staggering eighty percentage of India's population relies on this industry in some or other capacity. Numerous difficulties plague this industry in India, including erratic weather, uneven irrigation, overproduction of some crops, a lack of automation, and so on. It is clear that the use of deep learning (a type of artificial intelligence) can be of great benefit to farmers in many ways, including yield prediction, disease detection and diagnosis, etc. (Barman et al 2020).

Identification of the crop is a first and difficult step in yield prediction or any vision-based application (Chen and Li 2019). Here, we propose a straightforward application of deep learning to the problem of multiclass classification. Pre-trained models can also be used for this purpose, and these models are trained with millions of images across thousands of classes. Crop image classification is another application of these pre-trained models in our work, and we present the results for each of the relevant parameters to help other researchers to choose the best model for their own purposes (He et al 2016).

The objective of the proposed work is to check the possibility of use of transfer learning approach in crop image classification. The work also aims to compare the performance of pre-trained models with the one of emerging model Convolutional Neural Network (CNN) (Doraiswamy et al 2007).

2. Methodology

Here we have built three different models for crop image classification. The input is the image containing the crop and the model correctly identifies the crop in image (Sardeshmukh et al 2013; Sardeshmukh et al 2020). Following three models built and tested against the test dataset.

1. VGG16
2. Resnet50
3. Convolutional Neural Network (CNN)

2.1. VGG 16

The structure of VGG 16 is shown in Figure 1. For the network's input, we use a dimensional image (224, 224, 3). A 64-channel, 3x3-embedded filter size and the same padding are used for the first two layers. Following a max pool layer with stride = 2, two convolution layers with 128 and 32-bit filters are used (3, 3). The next layer is also a max-pooling layer, with the same stride (2, 2). This is followed by two 256-filter convolution layers with a size of (3, 3). A max pool layer and two sets of three convolution layers follow. There are 512 filters of size (3, 3) and the same padding in each one. Then, two convolution layers are stacked on top of this image. Convolution and max-pooling layers are implemented using 3*3-sized filters as opposed to AlexNet's 11*11 and ZF-7*7 Net's sizes. For layer-specific channel manipulation, 1x1 pixels are also employed in some stages. In order to preserve the spatial feature of the image, a 1-pixel padding is performed after each convolution layer.

Received: March 3, 2023 | Accepted: May 9, 2023

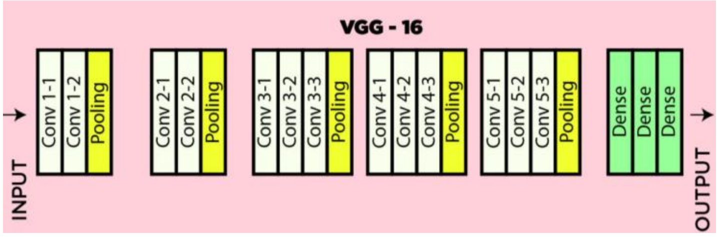


Figure 1 VGG 16 Model.

Our final feature map size (7, 7, 512) was the result of a convolution layer followed by a max-pooling layer stack. The output is a (1, 25088) feature vector after being flattened. The third fully connected layer is used to implement the softmax function required for classifying the 1000 classes in the ILSVRC challenge. The first fully connected layer takes input from the final feature vector and outputs a (1, 4096) vector, while the second fully connected layer also outputs a vector of size (1, 4096). The activation function of all the hidden layers is ReLU. Since ReLU leads to faster learning and reduces the likelihood of vanishing gradient problems, it is more computationally efficient.

Model Summary:

```
Model: "model"
_________________________________________________________________
Layer (type)              Output Shape              Param #
=================================================================
input_1 (InputLayer)      [(None, 256, 256, 3)]     0
_________________________________________________________________
block1_conv1 (Conv2D)     (None, 256, 256, 64)      1792
_________________________________________________________________
block1_conv2 (Conv2D)     (None, 256, 256, 64)      36928
_________________________________________________________________
block1_pool (MaxPooling2D)  (None, 128, 128, 64)    0
_________________________________________________________________
block2_conv1 (Conv2D)     (None, 128, 128, 128)     73856
_________________________________________________________________
block2_conv2 (Conv2D)     (None, 128, 128, 128)     147584
_________________________________________________________________
block2_pool (MaxPooling2D)  (None, 64, 64, 128)     0
_________________________________________________________________
block3_conv1 (Conv2D)     (None, 64, 64, 256)       295168
_________________________________________________________________
block3_conv2 (Conv2D)     (None, 64, 64, 256)       590080
_________________________________________________________________
block3_conv3 (Conv2D)     (None, 64, 64, 256)       590080
_________________________________________________________________
block3_pool (MaxPooling2D)  (None, 32, 32, 256)     0
_________________________________________________________________
block4_conv1 (Conv2D)     (None, 32, 32, 512)       1180160
_________________________________________________________________
block4_conv2 (Conv2D)     (None, 32, 32, 512)       2359808
_________________________________________________________________
block4_conv3 (Conv2D)     (None, 32, 32, 512)       2359808
_________________________________________________________________
block4_pool (MaxPooling2D)  (None, 16, 16, 512)     0
_________________________________________________________________
block5_conv1 (Conv2D)     (None, 16, 16, 512)       2359808
_________________________________________________________________
block5_conv2 (Conv2D)     (None, 16, 16, 512)       2359808
_________________________________________________________________
block5_conv3 (Conv2D)     (None, 16, 16, 512)       2359808
_________________________________________________________________
block5_pool (MaxPooling2D)  (None, 8, 8, 512)       0
_________________________________________________________________
flatten (Flatten)         (None, 32768)             0
_________________________________________________________________
dense (Dense)             (None, 5)                 163845
=================================================================
Total params: 14,878,533
Trainable params: 163,845
Non-trainable params: 14,714,688
```

2.2. RESNET 50

The resnet 50 architecture contains the following element Convolution with a 7x7 kernel and 64 separate kernels, each with a stride of 2, yielding a single layer. We then move on to maximum pooling, which, in this case, also uses a stride size of 2. We then perform a convolution with a 1 * 1,64 kernel, a 3 * 3,64 kernel, and a 1 * 1,256 kernel, with each of these three layers being repeated three times for a total of nine layers. There is then a kernel of size 1 * 1,128, followed by a kernel of size 3 * 3,128, and finally a kernel of size 1 * 1,512. This sequence was repeated four times, for a total of 12 layers. Thereafter, we have a kernal of 1 * 1,256 followed by a kernal of 3 * 3,256 and a kernal of 1 * 1,1024 for a grand total of 18 layers (this is repeated 6 times). After that, we added three more layers, each consisting of a 1 * 1,512 kernel, followed by a 3 * 3,512 kernel, and finally a 1 * 1,2048 kernel. When we're done, we'll have 1 layer thanks to an average pool followed by a fully connected layer with 1000 nodes and a softmax function at the end.

2.3. Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) are a class of neural networks that are particularly well-suited for image classification tasks. CNNs are designed to automatically and adaptively learn spatial hierarchies of features from raw input data, without the need for manual feature extraction. At a high level, a CNN consists of a series of layers, each of which performs a specific transformation on its input data. The first layer typically consists of a set of filters that are convolved with the input image to produce a set of feature maps, each of which highlights a particular aspect of the image. Subsequent layers may perform additional convolutional, pooling, and non-linear transformations to further extract higher-level features from the input. The final layer of the network typically consists of one or more fully connected layers that use the extracted features to make a prediction about the input data. During training, the parameters of the network are learned through backpropagation, a process by which the errors between the predicted output and the true output are propagated backwards through the network to update the weights of each layer. CNNs have achieved state-of-the-art results on a wide range of computer vision tasks, including image classification, object detection, and segmentation.

Network Architecture is presented in Figure 2:

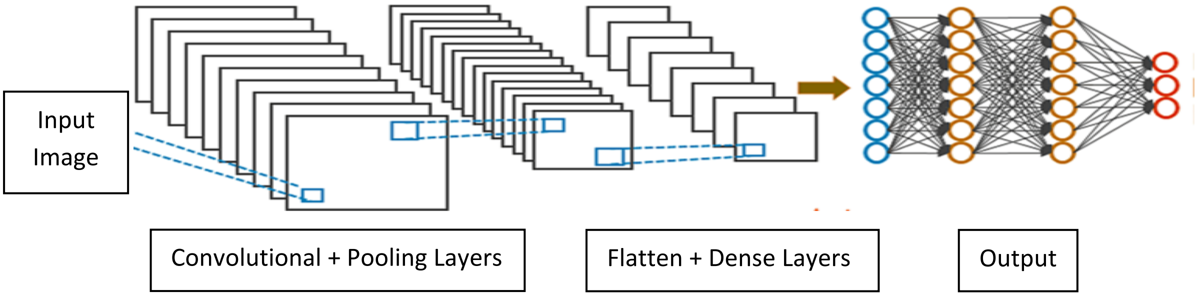


Figure 2 Training Loss of models.

Model Description

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)              Output Shape                    Param #
=================================================================
conv2d_3 (Conv2D)          (None, 254, 254, 16)           448
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 127, 127, 16)       0
_________________________________________________________________
conv2d_4 (Conv2D)          (None, 125, 125, 32)          4640
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 62, 62, 32)          0
_________________________________________________________________
conv2d_5 (Conv2D)          (None, 60, 60, 64)           18496
_________________________________________________________________
flatten_1 (Flatten)        (None, 230400)                   0
_________________________________________________________________
dense_1 (Dense)            (None, 5)                    5990426
=================================================================
Total params: 6,014,010
Trainable params: 6,014,010
Non-trainable params: 0
```

The structure of our proposed CNN model is given in Figure 2. In our CNN we have used 3 convolution layers with 3*3 filter, 2 Max pooling layers, Flatten layer and the Dense layer. The convolution layers are used for feature extraction. Features like edges are extracted from the first convolution layer, whereas the detailed features like objects and shapes are obtained in the subsequent layers. Max pooling 2D layer is a common layer used in convolutional neural networks for image classification tasks. It operates on a 2D input tensor (usually a feature map produced by a convolutional layer) and reduces its spatial dimensions by taking the maximum value in each non-overlapping rectangular region of the input. The primary use of max pooling 2D layer is to down sample the input tensor and reduce its spatial dimensions, while preserving the most important features of the input. This can help to make the network more computationally efficient by reducing the number of parameters and computations required in subsequent layers. Another benefit of max pooling is that it can provide some degree of translation invariance to the input, meaning that small variations in the position or orientation of an object in the input can be tolerated to some extent without affecting the output of the network. This is because the pooling operation takes the maximum value within a given region, regardless of the precise location of the feature Overall, max pooling 2D layer is an important building block in many convolutional neural networks, as it allows the network to effectively reduce the spatial dimensions of the input while preserving the most important features, thereby improving its performance and computational efficiency. The flatten layer reshape (rearrange) input in vector of size 1*n. Final layer that is output layer is Dense layer consisting of 5 neurons as we have to classify the crop image in to one of the five classes available in the training and testing dataset. The ReLu is used as activation function in convolution layer and Sigmoid is used in the Dense/output layer.

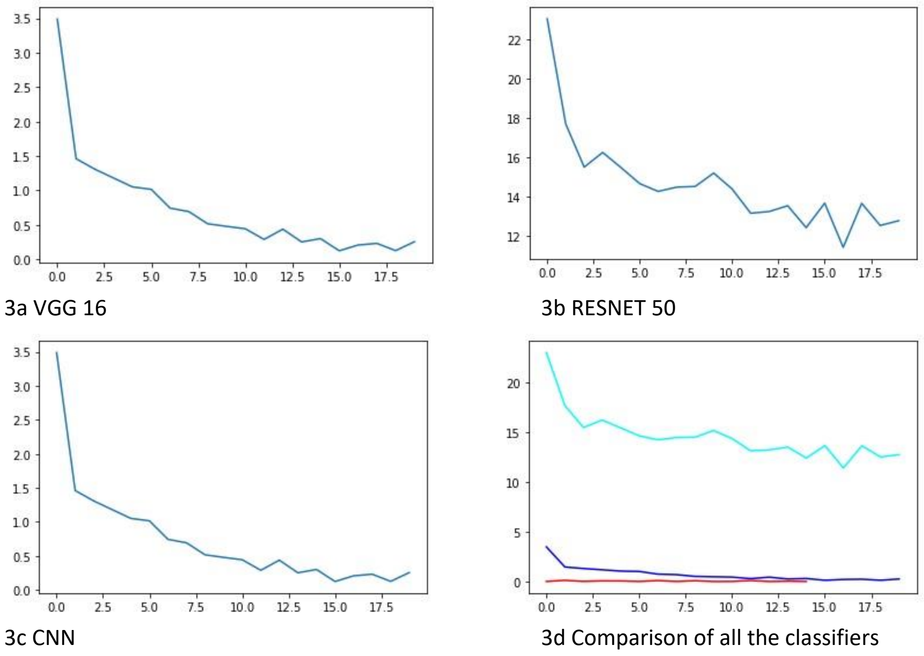
3. Results and Discussion

3.1. Dataset

Dataset is a collection of crop images. Data plays a crucial role to build any effective model using machine learning or Deep Learning. Because the model is trained based on the provided image data so the data must be insightful and meaningful to solve our task. Training dataset contains 804 augmented images of 5 different crops. Augmentation contains Horizontal flip, rotation, horizontal shift, vertical shift. The crop images in the dataset captured at different locations, different views and in different background. Some images are aerial view some contains other objects also. The test dataset contains total 201 images of five classes. The larger the data used in training improvement can be seen in the results. More image used in training helps the CNN model to understand the patterns in the respective class more precisely.

3.2. Results and Discussion

Figure 3 gives the training loss and the training accuracy of all the models after the 20 epochs for all.



3a VGG 16

3b RESNET 50

3c CNN

3d Comparison of all the classifiers

Figure 3 Training Loss of models.

In Figure 2, the confusion matrix for all the classes is given. Confusion matrix is used to understand the quality of the classifier. In addition to the accuracy we can find out the misclassification of class if any.

A confusion matrix is often used to evaluate the performance of machine learning model, particularly for classification problems, by comparing the actual and predicted labels of a dataset. The matrix displays the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) for each class in the classification model. The confusion matrix for Vgg16, Resnet 50, proposed CNN model and accuracy for all these is shown in Figure 4.

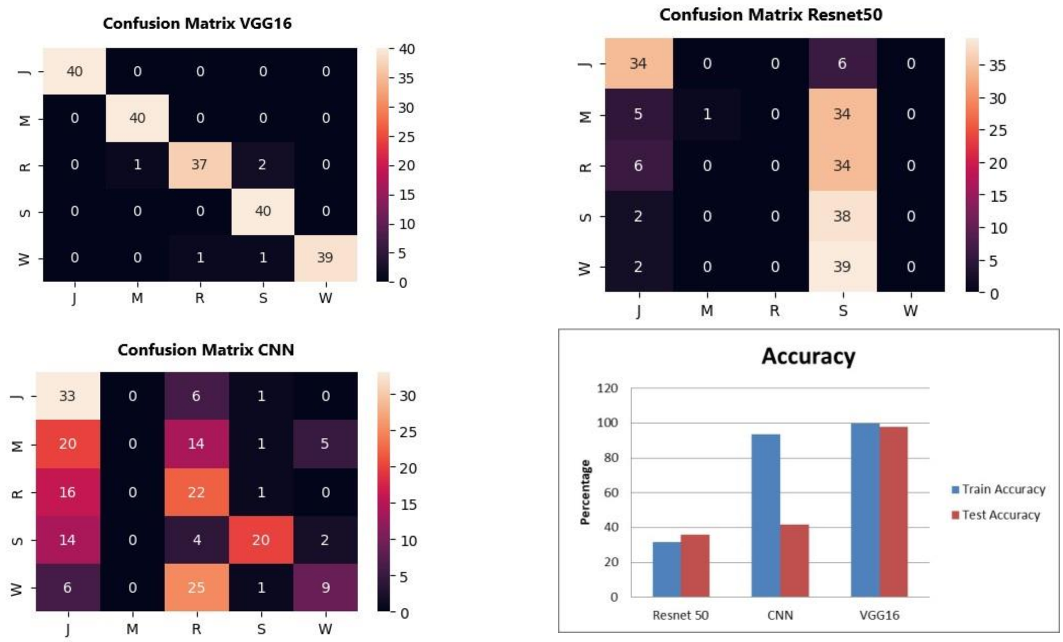


Figure 4 Confusion Matrix and Accuracy of Classifier.

From the results it is observed that the accuracy of the VGG16 model is 98% whereas the Resnet 50 and CNN gives the accuracy of 35.82 % and 41.79 percent respectively. The training accuracy in case of CNN is 93% and even it can be increased further by increasing the number of epoch but in testing it fails miserably in case of Maize and Sugarcane images. This is due to the less number of images used in training. If training images increased, then the CNN also having capacity to improve in accuracy. In case of Resnet 50 model only Jute and Sugarcane images features are extracted in case of other classes it gets misclassified by Sugarcane. This may be because of the training images used in training. From all these observations it is concluded that VGG16 model is best suited for this kind of application. Further this crop image classification can be used for yield prediction of a particular area. VGG 16 and Resnet 50 both are pretrained models but still the performance of VGG 16 model is superior to Resnet 50. The ResNet50 model is a deep convolutional neural network architecture that has achieved state-of-the-art performance on various computer vision tasks, including image classification. However, if the ResNet50 model is not performing well on cropped image classification tasks, there could be several possible reasons for this. One possible reason is that the cropped images may not be large enough to provide sufficient information to the model. ResNet50 was originally designed to work with images that are at least 224x224 pixels, and cropping these images may result in a loss of information that the model requires to make accurate predictions. Another reason could be that the cropping process may have introduced artifacts or distortions in the images, which can affect the performance of the ResNet50 model. For example, if the cropping process results in images that are not properly aligned or contain only a partial object, the model may not be able to correctly classify them. Lastly, the ResNet50 model may not have been trained on cropped images, and therefore may not have learned the appropriate features required for this specific task. In this case, fine-tuning the model on a dataset of cropped images may be necessary to achieve better performance on this task. Overall, the performance of the ResNet50 model on cropped image classification tasks depends on various factors, including the size and quality of the cropped images, the nature of the cropping process, and the training data used to train the mode.

4. Conclusion

From the results it can be concluded that the use proper selection of pretrained models can help in developing the models to solve the real-life problems. Availability of training data is biggest hurdle in the real-life problems like agricultural issues and many more. Use of pretrained model can certainly help in this situation. It is observed from the results that all the pretrained models are not useful we have to select it with proper experimentation. Here the VGG 16 finds suitable in crop image classification.

Ethical considerations

Not applicable.

Conflict of Interest

The authors declare that they have no conflict of interest.

Funding

This research did not receive any financial support.

References

Barbedo JGA (2018) Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. Computers and Electronics in Agriculture 153:46-53. DOI: 10.1016/J.COMPAG.2018.08.013

Barman U, Choudhury RD, Sahu D, Barman GG (2020) Comparison of convolution neural networks for smartphone image based real time classification of citrus leaf disease. Computers and Electronics in Agriculture 177:105661. DOI: 10.1016/J.COMPAG.2020.105661

Chen H, Li Y (2019) Three-Stream Attention-Aware Network for RGB-D Salient Object Detection. IEEE Transactions on Image Processing 28:2825–2835. DOI: 10.1109/TIP.2019.2891104

Doraiswamy PC, Stern AJ, Akhmedov B (2007) Crop classification in the U.S. corn belt using MODIS imagery. International Geoscience and Remote Sensing Symposium (IGARSS) 809-812. DOI: 10.1109/IGARSS.2007.4422920

He K, Zhang X, Ren S, Sun J (2016). Deep residual learning for image recognition. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 770-778. DOI: 10.1109/CVPR.2016.90

Kavitha AV, Srikrishna A, Satyanarayana C (2022) Crop image classification using spherical contact distributions from remote sensing images. Journal of King Saud University - Computer and Information Sciences 34:534–545. DOI: 10.1016/J.JKSUCI.2019.02.008

Sardeshmukh MM, Kolte MT, Chaudahri DS (2013) Activity recognition using multiple features, subspaces and classifiers. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8298LNCS:617-624. DOI: 10.1007/978-3-319-03756-1_55/COVER

Sardeshmukh MM, Kolte MT, Sardeshmukh VM (2020) Inter person activity recognition using RGB-D data. Journal of Engineering Science and Technology 15:3601-3614.

Yang G, He Y, Yang Y, Xu B (2020) Fine-Grained Image Classification for Crop Disease Based on Attention Mechanism. Frontiers in Plant Science 11:1–15. DOI: 10.3389/fpls.2020.600854