DETECTION OF DAMAGED LEAF USING CONVOLUTIONAL NEURAL NETWORK

M. Senthamil Selvi¹, K. Deepa², Mrs. S. Jansirani Sankar³

¹Professor & Head, Department of Information Technology, Sri Ramakrishna Engineering College, Coimbatore-22
²Professor, Department of Information Technology, Sri Ramakrishna Engineering College, Coimbatore-22
³Assistant Professor, Department of Information Technology, Sri Ramakrishna Engineering College, Coimbatore-22

¹senthamilselvi@srec.ac.in, ²deepa.senthil@srec.ac.in, ³jansi.sankar@srec.ac.in

Abstract

In recent years, Deep Learning technologies are more popular and used in many fields like agriculture, healthcare, manufacturing etc. One of the areas in deep learning is image classification and the results are useful, successful with more accuracy. Deep learning algorithm for image classification is CNN (Convolutional Neural Network). This paper uses the leaf image dataset like Good leaf images, leaf with worms and leaf with insect images. It is very important to classify the leaf in the agriculture field to spray the pesticide or insecticides. Sometimes, some leaves are good in particular areas; those areas need only water for growth. This paper deals with deep learning techniques such CNN, used to classify leaf images using MATLAB. The objectives of the work is to classify leaves as Good, Worms, Insects for better understanding and spray of Pesticides, Insecticides, this helps farm owners for better yield and it indirectly increases the economic growth of the country.

Keywords: CNN, Alexnet, Pesticides, Insecticides, MATLAB

I. Introduction

Automation in every field, agriculture also automated with the help of AI, Machine Learning and deep learning techniques and methods. In India, Agriculture is influencing factor for economic growth. More population directly or indirectly involving in agriculture. For better yield and growth, farmers are using Pesticides, Insecticides and Herbicides spray [III]. It is very important to choose spray based on leaf with insect or worms (in market separate sprays available for insects and worms). Worms and insects in the leaf is unavoidable, this proposed framework can be used to detect good, worm and insect leaf. The work based on novel deep learning...
CNN algorithm. Alexnet and Googlenet used for the leaf dataset classification. In this Paper, all the implementation done by MATLAB online framework.

Dataset: Leaf dataset (images collected from google) consists of 30 samples includes 10 good leaf images, 10 leaf with worms and 10 leaf with insect shown in the figure 1.1,1.2,1.3 .The images in the dataset are different in size and pixels

II. Convolutional Neural Network

One of the mainclass of machine learning is Deep Learning [V]. It has sequential of multi-layer to learn the features of the given input. Feature learning can be done automatically based on the algorithm and the input given. There are many types of learning such as unsupervised, supervised or semi-supervised. LeCun et al. defines the deep learning methods as a representation learning method [VI]. Representation learning used to understand the representations or features from the input and the representations must be appropriate. It is based on the input and the algorithm [II]. During the training process, the deep learning network extract the features from the input. Deep learning used in many applications such as healthcare, agriculture, manufacturing etc. Deep learning main research areas such as image processing, image restoration, speech recognition, natural language processing and bioinformatics uses the model. In this paper, for the study, CNN is chosen as a deep learning method. CNN also multi-layered structure, it identifies and classify the images with less pre-processing and it extract the features from the input image for classification. Four main layers in CNN are: convolutional layer, pooling layer, activation function layer and fully connected layer. Fig:2.1 Typical CNN Architecture (Figure taken from Wikipedia)
II.i. Alexnet

Alexnet is one of the type of Concolutional Neural Network [I] and the same method is used in this paper. The accuracy and the speed of the Alexnet deep learning is better than others. Alexnet architecture consists of many layers includes eight learned layers, five convolutional layers and three fully connected layers. Alexnet have 1000 pre defined classes. The last layer in Alexnet is fully-connected layer, it connects to 1000 predefined classes and the remaining in the network is considered as a feature extractor.

II.ii. Transfer Learning

Transfer learning[IV] used to learn new task. By the transfer learning, the pre-defined network is fine tuned to learn a new task and identify the new classes,This papers, fine tune the Alexnet to learn the leaf images as new classes. Training the new network as layer by layer is very difficult process, but the transfer learning eliminates the difficulty. It produces the better result and it is easy and fast

The transfer learning can be done with the following sequence to learn the network for new samples:

- Identify the pretrained network model (here it is Alexnet) and import the same
- Replace the layers with fresh layers, the new layer must adopted to the new data set
- Indicate the number classes in last layer , here it is 3
- Set learning rates to learn faster in the new layers than in the transferred layers.
- Export the network for training at the command line.

III. Implementation

Step 1: Upload the dataset into the MATLAB Folder

![Fig 3.1: Leaf Image Dataset](image-url)
Fig 3.1 shows the upload dataset into MATLAB environment.

**III.i. Using Pre-Trained Alexnet:**

1. First read the image from the folder
   
   ```
   img1=imread('Goodleaf01.jpg')
   ```

   ![](image1.png)

   ![](image2.png)

   B = `imread(filename, fmt)` reads a greyscale or color image from the specified file by the string filename, where the string fmt specifies the format of the file.

2. Use the pretrained network by using the command
   
   ```
   net=alexnet
   ```

   The input size for the alexnet is 227x227. So convert the image size to 227 high and 227 width using `augmentedImageDatastore([227,227],image name)`. Then the image is resized with 227x227.

3. Predict the label of the image using `pretrainedalexnet` (alexnet have 1000 pre-classified labels).

   By using the pre trained alexnet, the images mapped to one of the 1000 classified labels based on the score value of the labels. The scores can be plotted as bar chart. Fig 3.1.2 show the some of the Alexnet categories.

   ```
   >> categorynames = net.Layers(end).ClassNames
   categorynames =
   1000x1 cell array
   {'tench'    }
   {'goldfish' }
   {'great white shark'}
   ```

   ![](image3.png)
The image given as a input is shown in the Fig 3.1.3.

![Good Leaf Image](image)

**Fig 3.1.3:** Good Leaf Image

By using the pretrained Alexnet, the image is classified as sulphur butterfly.

**Fig: 3.1.4:** Barchart based on the score value

Based on the above Fig 3.1.4, PretrainedAlexnet predict the given image is sulphur butterfly but the image is good leaf. So, network has to train on the new dataset by dividing the samples into training and testing. This paper uses the known leaf images and it is defined in excel sheet and stored it as .csv file format. Further section explains transfer learning in Alexnet. This example dataset uses transfer learning to train a deep network that can classify images of leaf as either good or worm or insect.

Create csv file:

|    |        |    |
|----|--------|----|
| 1  | File   | A  |
| 2  | Goodleaf01.jpg | Good |
| 3  | Goodleaf02.jpg | Good |
| 4  | Goodleaf03.jpg | Good |
| 5  | Goodleaf04.jpg | Good |
| 6  | Goodleaf05.jpg | Good |
| 7  | Goodleaf06.jpg | Good |
| 8  | Goodleaf07.jpg | Good |
| 9  | Goodleaf08.jpg | Good |
| 10 | Goodleaf09.jpg | Good |
| 11 | Goodleaf10.jpg | Good |
| 12 | Insect01.jpg  | Insect|
| 13 | Insect02.jpg  | Insect|
| 14 | Insect03.jpg  | Insect|
| 15 | Insect04.jpg  | Insect|
| 16 | Insect05.jpg  | Insect|
| 17 | Insect06.jpg  | Insect|
| 18 | Insect07.jpg  | Insect|

**Fig. 3.1.5:** csv File

Fig 3.1.5 .csv file created with file name and known status of the leaf and the same uploaded into matlab for further process.

*Copyright reserved © J. Mech. Cont. & Math. Sci.  M. Senthamil Selvi et al.*
III.ii. Fine Tuning Alexnet

Are the Leaf Good or Worm or Insect?

1. Get the training images and classes
Create a datastore to the images.

```matlab
imd = imageDatastore('LeafImages');
```

`imd` consists of 30 leaf images

2. Get the known classifications from a file and use these as the image labels.

```matlab
>> imds.Labels = categorical(groundtruth.Status)

imd =

struct with fields:

Labels: [30x1 categorical]
```

![Fig 3.2.1: Image Labels](image)

Fig 3.2.1 show some label of the image data set.

3. View the first few images. The second argument to imshow scales the display based on the range of pixel values in the image. Fig 3.2.2 shows the first image of the dataset.

```matlab
>> imshow(readimage(imdb,1))
```

![Fig 3.2.2: imshow result](image)
4. Divide data into training (60%) and testing (40%) sets

Fig 3.2.3: Train Images

Fig 3.2.4: Test Images

5. Create augmented image datastores to preprocess the images.

Fig 3.2.5: Converting the images into Input size of Alexnet

Fig 3.2.5 shows the conversion of the input images both training and testing image set into 227 high and 227 wide because the required input size of the alexnet is [227,227].

6. Build a network:
Start with a pretrained network.

```matlab
>> net = alexnet

net = 
```

SeriesNetwork with properties:
```
InputNames: {'data'}
OutputNames: {'output'}
```

Fig 3.2.6: Pretrained Alexnet
7. Take the CNN layers and add new classification layers at the end.

```matlab
>> reluLayer = net.layer(5); % Fully Connected layer
>> reluLayer = net.layer(6); % Softmax layer
```

**Fig 3.2.7:** FinetuneAlexnet

Fig 3.2.7 shows FinetuneAlexnet by giving 3 fully connected layers because the final image label is Good, Insect, and Worms, and the final layer as classification layer.

8. Set some training options

```matlab
>> topts = trainingOptions('adam', 'InitialLearnRate', 0.0001)
```

The learning rate assigned as 0.0001

9. Train the network

```matlab
>> LeNet = trainNetwork(tr, net.layers, topts)
```

**Fig 3.2.8:** Result of Trained Network

Fig 3.2.8 shows the base learning rate, epoch value, iteration, mini-batch accuracy and loss.

10. Evaluate network on test data.
The fine tuned model used to make predictions on testing dataset

```
>> preds = classify(Leafnet,testds)
preds =
12x1 categorical array
   Insect
   Good
   Insect
   Insect
   Insect
   Worm
   Insect
   Insect
   Worm
```

**Fig 3.2.9: Prediction**

Fig 3.2.9 shows the predicted label of 12 testing images and the test images classified as Good, Insect and Worm.

11. Compare with reality

```
>> nnz(preds == truetest)/numel(preds)
ans =
0.5000
```

**Fig 3.2.10: Value of Prediction**

12. **View Confusion Matrix**

```
>> confusionchart(truetest,preds)
an =
ConfusionMatrixChart with properties:
   NormalizedValues: [8x3 double]
   ClassLabels: [3x1 categorical]
```

**Fig 3.2.11 Confusion Matrix**

Fig 3.2.11 Confusion matrix shows the image label of true class and predicted classes.

**IV. Conclusion**

This work aims to study the Alexnet model and apply the transfer learning for new classes in Alexnet. The paper uses Alexnet to classify the leaf images and finetuned to recognize the 3 classes as Good, Insect and Worms only with 30 leaf images as training and testing dataset. The proposed model gives better result for the static images. On the other hand, the architecture of the proposed fine tuned AlexNet model
can be improve as real-time testing may stoppage recognition. The opportunity to extend the work can be done by using same dataset for other networks and with comparing the results with other like googlenet. Also the network can be tested with the new dataset.

V. Acknowledgement

Implementation to classify the leaf as Good, Insect and Worm with fine-tunedAlexnet Network have been accomplished without the support of MathWorks document and MATLAB online.

References

I. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” Adv. Neural Inf. Process. Syst., pp. 1–9, 2012.

II. H. Durmus, E. O. Gunes, and M. Kirci, “Disease detection on the leaves of the tomato plants by using deep learning”, In Agro-Geoinformatics, IEEE 6th International Conference on, pp. 1-5, 2017.

III. http://www.llojibwe.org/drm/greenteam/pesticides_Article.pdf

IV. https://in.mathworks.com/help/deeplearning/ug/transfer-learning-with-deep-network-designer.html

V. MelikeSardogan ;AdemTuncer ; YunusOzen, “Plant Leaf Disease Detection and Classification Based on CNN with LVQ Algorithm,” 2018 3rd International Conference on Computer Science and Engineering (UBMK) on pp.382-385

VI. Y. Le Cun, Y. Bengio and G. Hinton, “Deep Learning”, Nature, vol. 521, pp. 436-444, 2015