Plant Disease Detection using Image Segmentation& Convolutional Neural Network

A MAJOR PROJECT-II THESIS REPORT

SUBMITTED IN PARTIAL FULFILMENT OF REQUIREMENT
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
SOFTWARE ENGINEERING

Submitted By -
Ravi Kaushik
(Roll No. 2K17/SWE/19)

Under the supervision of
Dr. Shailender Kumar
(Associate Professor)
Delhi Technological University

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042
June, 2019

DECLARATION

I hereby declare that the Major Project-II work entitled **“Plant Disease Detection using Image Segmentation”** which is being submitted to Delhi Technological University, in partial fulfillment of requirements for the award of the degree of Master of Technology (Computer Science and Engineering) is a bona fide report of Major Project-II carried out by me. I have not submitted the matter embodied in this dissertation for the award of any other degree or diploma.

Place: Delhi

Ravi Kaushik
2K17/SWE/19

CERTIFICATE

This is to certify that Project Report entitled **“Plant Disease Detection using Image Segmentation”** submitted by **Ravi Kaushik** (2K17/SWE/19) in partial fulfilment of the requirement for the award of degree Master of Technology (Software Engineering) is a record of the original work carried out by him under my supervision.

Place: Delhi

Date:

Project Guide
Dr. Shailender Kumar
Associate Professor
Department of Computer Science & Engineering
Delhi Technological University

ACKNOWLEDGEMENT

First of all, I would like to express my deep sense of respect and gratitude to my project supervisor **Dr. Shailender Kumar** for providing the opportunity of carrying out this project and being the guiding force behind this work. I am deeply indebted to him for the support, advice and encouragement he provided without which the project could not have been a success.

Secondly, I am grateful to Dr. Rajni Jindal, HOD, Computer Science & Engineering Department, DTU for her immense support. I would also like to acknowledge Delhi Technological University library and staff for providing the right academic resources and environment for this work to be carried out.

Last but not the least I would like to express sincere gratitude to my parents and friends for constantly encouraging me during the completion of work.

Ravi Kaushik
Roll No – 2K17/SWE/19
M. Tech (Software Engineering)
Delhi Technological University

ABSTRACT

Identifying regions in an image and labeling them to class is called image segmentation. Automatic image segmentation has been one of the major research areas which is in trend nowadays. Every other day a new model is being discovered to do better image segmentation for the task of computer vision. As the better a computer is able to see, the better we can automate the tasks around our daily life. In this survey we are comparing various image segmentation techniques and on the basis of our research we are applying the best approach to an application i.e. developing a model to identify diseased plants and to give an idea to the people what kind disease is present in a plant. The detailed analysis of the methodology is done with the help of various analysis techniques, which are used in reference to the context of the work. Our focus is on the techniques which we are able to optimize and make them better than the one which are present before. This survey emphasizes on the importance of application of image segmentation techniques and to make them more useful for the common public in daily life. So that they get benefits of this technology in the monitoring of any activity occurring around that can't be done manually.

Keywords: Image segmentation, Deep Learning, Convolutional Neural Networks, Edge detection models etc.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

List of Abbreviations and Nomenclature

1. CNN- Convolutional Neural Network
2. ANN- Artificial Neural Network
3. R –CNN- Region based Convolutional Neural Network
4. GPU- Graphics Processing Unit
5. ReLu- Rectified Linear Unit
6. UAV- Unmanned Aerial Vehicle
7. CC- Connected Component
8. MRI- Magnetic Resonance images
9. MOCNN- Multiple Output Convolutional Neural Network
10. IDE- Integrated Development Environment

CHAPTER 1

INTRODUCTION

1.1 Background

In modern world, images are primary source of information sharing. We can find them in almost every area of work. But images can be read accurately only by humans as they need to be interpreted in a natural perspective way. Nowadays, as we know the technology has advanced up to an extent such that it can match the ability of a human brain. Now interpretation of images by computer is possible. Images can be read and objects in it can be identified automatically. Based on those results analysis and automatic decision making is now a possible task. Various machine learning technologies have been introduced for such tasks, but CNN (Convolutional Neural Network) proves to be the best technique to do image segmentation or object identification.

1.1.1 Machine learning introduction and technique

Machine learning is the quintessential skill of this digital age. As we dissect the process how a machine learns to classify and the inputs or the raw materials needed for learning the specifics of the desired task, features or attributes forms the basis of what we feed in the learning algorithm. In the task of image processing and object identification machine learning plays an important role. There might be many techniques available to do such task. We will discuss various methods that can be used to achieve such task.

In this world of digitization, images play a very important role in various areas of life including scientific computing and visual persuasion tasks. Technically images can be binary images, grey scale images, RGB images, hue saturation value or hue saturated lightness images etc. Each data record can be represented via a huge number of features. But all features are not necessarily significant for analysis or classification. Thus, feature selection and feature extraction are significant research areas.

1.1.2 ANN (Artificial Neural network) using an MLP (Multilayer Perceptron)

A multilayer perceptron is a kind of feed forward artificial neural network. Artificial neural network is inspired by structure and functions of biological neural network. Generally, by biological neural network we mean the structure and working of human brain. ANN tries to imitate the functioning of the human brain by following its principles. As human brain is composed of billions of nerve cells called neurons similarly, an artificial neural network consists of many artificial neurons called nodes which behave as the same way a biological neuron does. Biological neurons are consisting of three parts.

Dendrites - They accept the input from the previous layer.

Axon - Neurons are connected to each other through the axon.

Synapses - They transfer the output to the next layer neurons.

The input is taken by dendrites and sent to the nucleus, then nucleolus decides whether to generate an output signal or not. If an output is fired by the nucleus it goes to synapses through the axon to be passed onto the next layer neuron. The information received from dendrites is passed onto the cell nucleus, and then it decides whether to transfer stimuli to the next layer or to reject the signal based on some threshold value. Similar way, an ANN also consists of neurons called nodes, they are connected to each other by links and each link is associated with some weight. a nodes receiver input from many nodes and based upon the activation function used in a node and action is taken or rejected. If the sum of input values reaches up to a threshold value then the output is generated ad passed onto the next layer input. Otherwise the input is rejected is rejected and no action against it is taken.

1.1.2.1 Advantages of ANN

- ANNs can learn by themselves and can create nonlinear complex relationships. Nonlinear relationships are very useful to construct as all the real-life relationships between inputs and outputs are generally non--linear input-output graph.
- ANN can be smart as humans, as after learning from a set of big training data, They can create unseen relationships' between inputs and outputs. Thus, they can be very useful in prediction of output on never seen input data.
- ANN never put restrictions on input data variables. Also, it is studied that ANN produces better results on heterogeneous data which has very high volatility and non-constant variance.

1.1.2.2 Applications of ANN

- Image processing and character recognition - As images are usually consist of non-linear instructed patterns, so ANN can work effectively on identifying the never seen objects in images based on some previous learning. Also recognizing characters is a task where unseen images are need to be processed, so based on some mathematical calculations and prediction ANN finds the class to which an object has most compatibility.

- Forecasting- This is also one of the major applications of ANN, as forecasting in any fields of life is mostly unpredictable so ANN tries find that unpredictable behavior and can forecast the outcomes before the happening of the event, which sometimes prove to be very useful- e.g. weather forecasting, stock market forecasting, score predictions etc.

1.1.2.3 Disadvantages of ANN

- Long detaining times - As ANN is used for predictions of unseen data this is the reason it requires a long training time to train the model. This is especially true if you are using CPU for the training instead of a GPU.
- Needs lots of data - Models with many layers need lots of data to completely build the network model. As there are lots of weights and connections, therefore it requires adjusting weights according to the input data given.
- Architecture must be fast tuned - Only having a good data set doesn't assure you the best result. It also requires the weights of the network to be adjusted optimally so that the predictions made later can be more accurate.

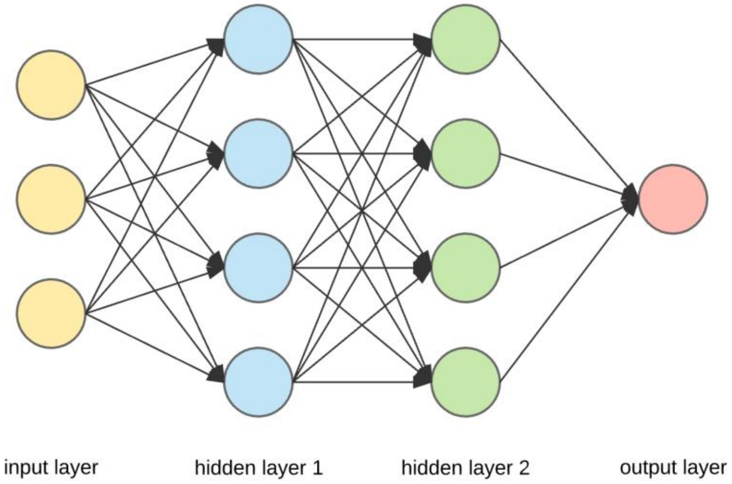


Figure 1.1 Architecture of ANN

1.1.3 Feature Selection

Feature selection can be defined as a problem of choosing the minimal set of features that are able to address the problem in a more effective, compact and computationally efficient manner. Feature selection involves creating new features existing ones, removing redundant and insignificant features, combining several features to a minimal count, as well as splitting a feature to several features.

1.1.4 Image segmentation using machine learning

Image segmentation is an important task in machine learning and computer vision applications. Semantic segmentation tries to partition the image into meaningful parts and label each part to one of a predefined class. There are infinite numbers of tasks that we can achieve with the help of image segmentation. For e.g. self-driving systems, background detection, robot controlling system, fruits and vegetables quality monitoring systems, production line quality maintenance, body cells diagnosis system etc. In image segmentation each pixel in image is labeled to different classes. This pixel labeling task is also called as dense prediction. Suppose in an image there are various objects available like cars, trees, signals, animals. So, image segmentation will classify all trees as a single class, all animals to a single class and all signals to a single class. One important thing to consider in image segmentation is that it considers two objects of the same type as a single class. We can differentiate objects of the same type using instance segmentation.

1.2 Need of Image Segmentation

Suppose, you are crossing a road and you see various objects around like vehicles, traffic lights, footpath, zebra-crossing and pedestrians. Now, while crossing the road your eyes instantly analyze each object and process their locations to take decision of whether to cross the road at that moment or not.

Can computers do this task? The answer to this question was ‘No’ since a long time before the breakthrough inventions in computer vision and object identification. But now with the help of image segmentation and object identification techniques it is possible for the computers to see the real-world objects and based on their positions they can now take necessary actions.Also, in the task of identifying cancer cells, image segmentation plays an important role. As it’s crucial to identify the shape of cells in blood and any unusual growth in shape of blood cell will be diagnosed for the presence of cancer cells. This makes the recognition of cancer in blood at an early stage so that it can be cured within time.

1.3 Why image segmentation with CNN is better

Among all the neural networks CNN (Convolutional Neural Network) is the most used and most trending neural network. As, nowadays everything works upon artificial vision and automation of tasks. People want to make self-drive cars, robots that can travel by themselves. All this is possible only with the help of CNN. CNN proves to the best method to process images, to identify objects in it, to detect scenes, human faces and other details in images.

1.4 Various ways to do Image Segmentation

Region Based Image Segmentation
Edge-detection based Image Segmentation
Clustering based Image Segmentation
CNN based Image Segmentation

1.4.1 Region Based image segmentation

Separates the objects into different regions based on some threshold value(s). This technique makes use of the image pixel intensities. It segments the similar intensity regions out of higher intensity regions. A threshold is selected for differentiating. Pixel intensity less than a threshold come under one region and above the threshold value pixels area comes under another region. Similarly, more one threshold can be selected and an image can be segmented into more than two regions based on pixel intensities.

It is the simplest way of segmenting an image based on its pixel values. This makes use of the fact that there will be a huge difference in pixel values at the edges of an object from the pixel values of its background pixels. So, in such cases we can set a threshold value, the difference in pixel values falling below the threshold can be segmented accordingly. In case there is one object in front of a single background only one threshold value will work. But in case there are multiple object or overlapping among objects we might need multiple threshold values. This technique is also called as threshold segmentation.

1.4.1.1 Advantages of Region Based

- Simple calculations
- Fast operation speed.
- When the object and background have high contrast, this method performs really well.

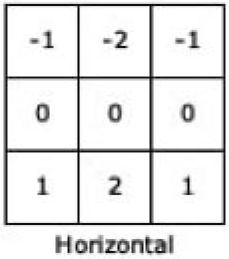
1.4.1.2 Limitations of Region Based

This method doesn’t perform well when objects in an image have their edge pixels overlapping with each other. It is not able to identify the boundaries of both objects and may consider both objects as one.

1.4.2 Edge Based Segmentation

In an image, what divides an object with its background or other object is an edge. So, this technique makes use of this fact that whenever a new edge is encountered while scanning an image, we are detecting a new object. This detection of edge is done with the help of filters and convolutions. Here, we have a filter matrix which we need to run over the image to detect the specified type of edge for which the filter was made. Edge based image segmentation is one of the most used and easy to implement image segmentation technique. As most of the task in image processing involves object identifications, which can be accomplished with detecting the edges in an image. It makes use of discontinuous local features of an image to detect edges and hence define a boundary of the object. This technique works well if there are only one or two objects in the image and also there is no overlapping between them. In other cases, this technique fails to identify objects correctly. These are also called kernels. These may be of many types. Some of the types are as follows:

- Sobel Filter (horizontal) - This filter is used to detect horizontal edges in the image.

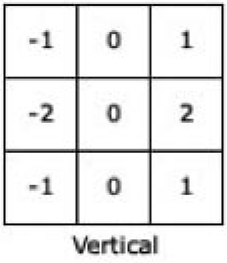


-1	-2	-1
0	0	0
1	2	1

Horizontal

Figure 1.2 Sobel Filter (horizontal)

- Sobel Filter (vertical) - This filter is used to detect vertical edges in the image.

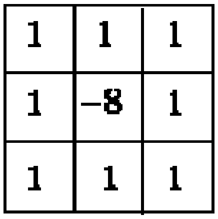


-1	0	1
-2	0	2
-1	0	1

Vertical

Figure 1.3- Sobel Filter (vertical)

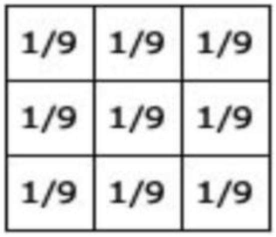
- Laplace Filter (both horizontal and vertical) - This filter is used to detect horizontal edges as well as vertical edges in the image.



1	1	1
1	−8	1
1	1	1

Figure 1.3- Laplace Filter (Both horizontal and vertical)

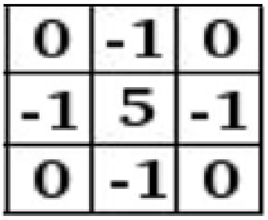
- Blurring filter - This filter is used to blur out an image.



1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Figure 1.4- Blurring Filter

- Sharpening filter – This filter is used to sharpen an image.



0	-1	0
-1	5	-1
0	-1	0

Figure 1.5- Sharpening Filter

1.4.2.1 Advantages of Edge Based Segmentation

- It is good for images having better contrast between objects.
- Fast calculations.

1.4.2.2 Limitations of Edge Based Segmentation

- Not suitable when there are too many edges in the image and if there is less contrast between objects.
- It doesn’t work correctly on images having overlapped object boundaries.

1.4.2.3 Step-by-step process Edge Based Segmentation

- Take the weight matrix
- Put it on top of the image
- Perform element-wise multiplication and get the output
- Move the weight matrix as per the stride chosen
- Convolve until all the pixels of the input are used.

1.4.3 Clustering based image segmentation

This technique work in the way that it divides the data points in an image in some clusters having similar pixel values. Based on the number of objects that might present in the image the value of number of clusters is selected. Then during the clustering process similar kind of data points were classified into a single cluster. In the end image gets classified into several regions. Although this might be a time taking process but it provides accurate results on small datasets. Image segmentation using clustering is based on the principle of divides the pixels of the image into homogeneous clusters. It selects the random cluster centers throughout the image, and then keeps increasing the size of clusters based on the similarity parameter. In this way the image gets segmented into regions equals to the number of clusters chosen. This technique work in the way that it divides the data points in an image in some clusters having similar pixel values. Based on the number of objects that might present in the image the value of number of clusters is selected. Then during the clustering process similar kind of data points were classified into a single cluster. In the end image gets classified into several regions. This might be a time taking process but it provides accurate results on small datasets.

1.4.3.1 Advantages of Clustering Based Segmentation

Works really well on small datasets and generates excellent clusters.

1.4.3.2 Limitations of Clustering Based Segmentation

- Computation time is too large and expensive.
- K-means is a distance-based algorithm. It is not suitable for clustering non-convex clusters.
- Selecting the optimal value of k can be tricky sometimes.

1.4.4 CNN based image segmentation

This method is currently the state of art technique in image segmentation field of research. It works on images which are of 3 dimensions i.e. height, width and number of channels. First two dimensions tell us the image resolution and third dimension represent the number of channels (RGB) or intensity values for red, green and blue colors. Usually images which are fed into the neural network are reduced in dimensions which reduce the processing time and avoid the problem of under fitting. Even though if we take an image of size 224*224*3 which when converted in to 1 dimension will make an input vector of 150528. So, this input vector is still too large to be fed as input to the neural network.

Image segmentation task using CNN is becoming very crucial nowadays. This technique is currently the state of art technology in image segmentation field. This is basically used for semantically segment an image into regions. Semantic segmentation is useful in scenarios where we need to train a car for self-drive. For doing this we use bunch of images and manually label all the images according to objects in it. Like we label trees, cars, signals, people, footpaths, trucks, cycles, animals etc using different masks and can later use this labeled data to train our CNN. Then when any new such image is fed into the network it will then be able to identify such objects and can take driving decisions according to objects identified in the scene. On the recent survey, it is found that deep learning is becoming a very important part of image segmentation techniques and algorithms increasingly. And it is continuously contributing in making the task of computer vision and automatic surveillance systems well trained and smart.

1.4.4.1 Advantages of CNN Based Segmentation

- It outperforms the traditional image segmentation methods.
- It can operate of any size of input
- The trained model is able to pixel wise predictions.

1.4.4.2 Limitations of CNN Based Segmentation

- Need of a large dataset.
- If network is pretty deep then each step takes a long time.
- Performance of the network highly depends upon parameter tuning

1.5 Applications of image segmentations in various fields

- In medical fields to identify tumors and cancer cell generations.
- cell segmentation task in light microscopic images
- Scene recognition
- Motion detection
- Video surveillance
- Geo sensing- for identifying target resource areas in satellite images
- Autonomous Driving
- Automatic braking
- Parking sensors
- Robotic path detection
- Facial segmentation- for identifying facial emotions, features or estimation of gender etc.
- In fashion for categorizing the clothing items.
- Agriculture – for segmenting the area in farms which needs to be herbicide. (reduces manual monitoring)
- Rotten vegetable identification
- Identifying neuronal structures in electron microscopic recordings.

1.6 Role of Deep Learning in image segmentation

As all other machine learning techniques work in a way of taking the training data and based of that training, they process new input and take the decisions. But as oppose to this deep learning works with neural networks and can make the conclusions of its own without the need of labeled training data. This method is useful for a self-driving car, so that it can differentiate between a sign board and a pedestrian. Neural networks use algorithms which are present in network side by side and output of one algorithm is subject to the outcome of another algorithm. This creates a system which can think as a human being for taking the decisions. And this makes a model which is a perfect example of an artificial intelligence system.

1.7 CNN (Convolutional Neural Network)

CNN is used for the classification of images. The input images which are used in CN are of 3 dimensions i.e. height, width and number of channels. First two dimensions tell us the image resolution and third dimension represents the number of channels (RGB) or intensity values for red, green and blue colors respectively. Usually, images which are fed into the neural network are reduced in dimensions which reduce the processing time and avoid the problem of under fitting. Even though if we take an image of size 224*224*3 which when converted in to 1 dimension will make an input vector of 150528. So, this input vector is still too large to be fed as input to the neural network.

Layers of CNN are:

- Convolution layer
- Activation layer (e.g. using ReLu)
- Pooling layer
- Batch norm Layer
- Drop out Layer
- Fully connected Layer

1.7.1 Convolution Layer:

A filter or kernel is used to run over the image in fixed gap intervals called strides. Selecting the size of stride is crucial to achieve desired results. During running the filter over the image, dot product of filter with part of image on which filter lies is calculated. Then sum of all values of product matrix is copied to the corresponding position in convolved feature map matrix. Thus, we get a reduced dimension feature map of image. Filters may be of many kinds, where each filter is used to extract different kind of feature from the image. For e.g. one filter may be responsible for extracting one kind of feature from the image based on shapes and edges and another filter may be used to extract feature based on color intensities.

1.7.1.1 Parameters which helps in adjusting CNN's performance:

- Stride **-** This defines the number of pixels by which we have to move our filter over the image so that we can focus on a new set of pixels while doing convolution. Stride's value ranges from 1 to 3 depending upon the amount of loss which we can be accommodated during convolution. The amount of loss in image increases with the increasing value of stride.
- Padding **-** It is a process of adding zeroes around the border of original image symmetrically. This helps us obtaining the feature map output to be of size as per our requirement. Commonly it is used to preserve dimension of image after convolution.
- Filters - These are also called kernels. These may be of many types. Each filter increases the depth of the output generated after convolution. So, if we are using 3 filters then the depth of the output will be 3. There are 3 parameters on which the output of convolution depends i.e. Stride, Depth and Padding. We must finely tune these parameters to obtain the desired output.

1.7.1.2 Dimension reduction in CNN

While convolution operation, the dimension of the input image is reduced after a filter matrix is convoluted over the input image. It also increases the depth of the image as this helps in identifying the regions in the image. e.g. a 9*9 RGB image with 3, 3*3 kernels at stride 1(per channel), will produce an RGB feature of 7*7*3. Here 3rd dimension represents the depth of the convolved image.

1.7.2 Pooling Layer

This layer operates a small kernel on the image at fixed stride. It is used to pick the pixel with highest intensity and discard other pixels. The resultant matrix will be a reduced dimensional matrix of feature image. This helps in reducing the unnecessary sparse cells of image which are of no use in classification. Max pooling helps reduce the dimensionality of the network (or image), but this may cause some information loss. The concept behind these is that adjacent or nearby pixels can be approximated by the maximum information carrying pixel.

1.7.3 Activation Layer

This layer mostly uses ReLu as an activation function. ReLu is a function which is used to set all negative values to zero and keeps positive value as it is. This step is usually followed by convolution and pooling layer. In numerous fields such as Computer Vision they are becoming the state of art achieving near human or better performance. They sound fascinating but designing a CNN is a herculean task in itself. Till now there is no fixed formula for the design of CNN. Many researchers have come up with the general suggestions such as. But they don’t always hold, as the task is dependent as importantly on data as it does on the algorithm. CNN exploits the spatial hierarchical features of data, extracting features and help classify them into different classes. This has led to development to a stream of data augmentation and pre-processing to increase the data, as more data allows for chance of better training and avoiding over fitting. This helps build models that are more robust to new samples as we try to make it more generalized to noise at training phase.

1.7.4 Dropout Layer

Dropout Layer is usually applied after the layer containing neurons in the fully connected network. Dropout layer is a regularization layer. It randomly drop out the weights of some input neurons in every iteration so that the other neurons get more weightage in that iteration. Usually people take dropouts of 20-50%. By using dropout layer, normally the performance of a network increase 1 to 2 %.

1.7.5 Fully Connected Layer

The structure from top to down usually forms a pyramid structure, the number of parameters in these layers keep on converging till they finally reach the number of desired classes. Increasing the number of hidden units in the layer can increase the learning ability of the network, but there is saturation of the increase in accuracy of the network. There is no formulation of the units you choose as it is a hit and trial usually. These also pile up with the depth of the fully connected subset of the network. Most of the networks in research usually perform well with number of units in multiple of 64. Two to three layers networks are good if there are enough patterns being passed to the network after flattening the outputs of the Convolutional layers.

1.7.6 Data Pre-processing and Augmentation

CNN capture spatial hierarchical features, but an algorithm’s capability is also limited by the pattern that exists in the data. One of features needed in a good algorithm is generalization. The training data tends to over fit and give high accuracy results on the training data while giving poor performance on the unseen data or testing data. One of the ways to improve it is data augmentation and pre-processing. Techniques such as flipping, rotation, translation, channel drop, Gaussian noise and cropping, this helps create more synthetic data and make it more robust to real world unseen data and classify it with better accuracy. The other benefit it provides is more data which usually allows opportunity for optimizing the parameters of CNN in a better manner. In this study, we investigate the specific case of image/video analysis. On the other side, many techniques exist, which try to provide a cleaner image rather than intentional noises such as Gaussian Noises. See in the dark, super resolution and reconstructing the blurred images are among them to name a few.

1.7.7 Basic Architectural diagram of CNN

This section discusses the design structure of CNN based on data and the optimization techniques that can help most of the CNNs with low dependency on the structure. We are actually going to see the techniques in a reverse manner similar to the flow of a back propagation to demonstrate the learning cycle of a CNN.

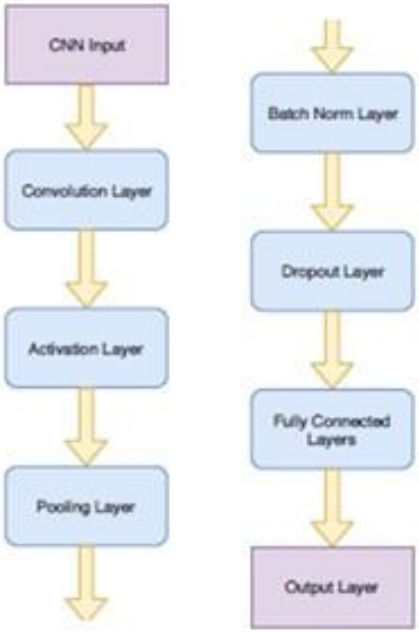


Figure 1.7 Architecture of CNN

1.7.8 OUTPUT OF CNN

The output of a CNN in tasks such as classification is usually the probability of different outputs. For example, if the final output of a CNN has 4 units (usually denoted as classes), they are normalized into probability of occurrence of these classes and the unit with maximum probability is labeled as the output class. This may actually vary for task specific cases such as facial recognition, where the output is an encoded form of the input rather than the representation of output for particular class. Back propagation helps minimizing the error by reducing the loss/cost function.

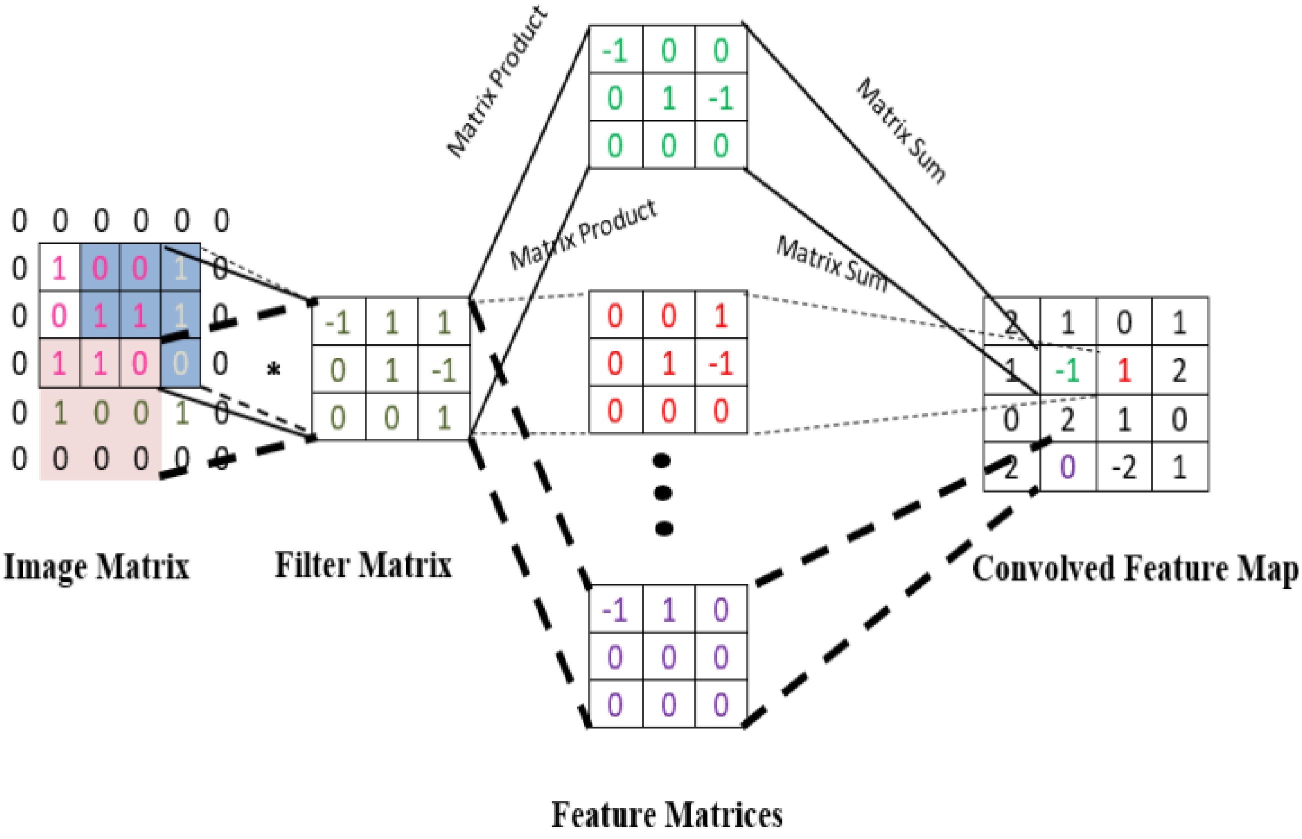


Figure 1.8 Work flow diagram of CNN

1.8 Objective of work

Here, in this work we are going to use image segmentation using CNN in plant disease detection. As we know farmers in our country are under crisis. And one of the main reasons of their situation is the deterioration of crops due to diseases. Most of the farmers in the country are illiterate so it is impossible for them to identify such diseases in crop at an early stage. So, till the time they get to know about the disease in their plants. The crop mostly crosses the time up to which they could be cured. So, in this approach we are intended to develop an automatic way to identify the plant disease that may be found in any kind of plant leaves. We are using several kinds of plant leaves as our dataset. And then any leave can be tested on the model to detect any such disease automatically. This way we can reduce the pressure of having availability of plant health expertise and risk of losing crop outcomes. Here, we are going to use several machine learning techniques such as CNN and edge detection and clustering for analysis and finding useful results of such classification image techniques.

1.9 Scope of work

As we know India is one of the largest crops producing country, but still the crop which it produces is not enough for the population which reside in it. So, it becomes necessary to produce hygienic disease-free crops for the people of country. A country progresses with its people. If its people are healthy and well fed then only, they can give a contribution in the wellness of country’s economy. So, the work which we are going to do have a big scope in the industry as it automates the process of identifying diseases in plants only with the help of images. So, a farmer can do a quick analysis of the kind of crop that he/she is growing. This will ultimately help in good crop production and also contribute to better country.

1.10 Organization of Dissertation

Chapter 1 describes the background of this work, which includes all the technologies that underlie it. The need and importance to do this work is explained in details. The work-flow diagram is also displayed for better understand of the reader. Then in Chapter 2 a detailed literature review has been done explaining all the major work that has already been done in this field of work. Then in the end of chapter 2, our approach to solve the problem has been described. In Chapter 3, all the steps of implementation are explained. The architecture that we have used for CNN is also explained. Then in Chapter 4, a detailed analysis and review of the implemented methods have been done. All the results have been discussed and areas of improvement have been mentioned. Results are displayed with help of graph plots and accuracy and other factors are being discussed as well. In Chapter 5, the conclusion and the future scope of this work has been discussed. A detailed discussion on how to extend the use of this technology has been done. Then Chapter 6 contains the references to all the resources that have been used to gather the information to work on this project.

CHAPTER 2

LITERATURE REVIEW

2.1 Work done in medical field

Olaf Ronneberger et al. [1] have done Image segmentation on biomedical images dataset. They have done segmentation of neural structures found in microscopic images of human blood samples. Cell growth tracking in a human body is performed automatically using this approach. The typical task of CNN is classification, where output to single image is a class label. But in many other applications, localization information should be included in output. That is each pixel in an image need to be assigned to a class label. In their work they have used CNN to segment the image to identify different cell types and to track the growth of some specific cells in the body over a period of time. Along with identifying the cells in the body, the location of those cells is also important to monitor the cell growth properly. They have applied segmentation task on a raw image which classify objects with different colors then a black & white segmentation mask is generated using white for foreground and black for background. Then the image is mapped with pixel wise loss weight to identify the border pixels. So, for that reason ceresin et al predicted the class label of pixels in images by providing a region which is local to the pixel in that image and with the help of a sliding window setup to train the network. Accuracy is 77.5%, training time is 10 hours [1].

Image segmentation is a very useful technique in medical area. As in MRI images or x-ray images, it is very difficult to identify different parts of body. Therefore, image segmentation helps in segmenting different part of the body from each other. **Pim Moeskops et al**. [4] applied image segmentation on MRI (magnetic resonance imaging) images of brain, breasts and cardiac CTA). The classical procedure of segmentation is to train the model with some hand-crafted features and then use the model on input data to classify the objects, whereas in this approach CNN automatically extracts features which are required at the hand for the task of classification. Here CNN is used to classify medical images of knee cartilage, brain regions, pancreas and coronary arteries. In their work they have done architecture extraction using segmentation of 3 orthogonal patches of 51 * 51. Extraction of features was done for each of such patches with the help of deep stack of Convolutional neural network have 25 layers. An output layer of 9 classes with two fully connected layers were implemented as well in this architecture. In convolution layer, 32 kernels for each input patch are used to extract features out of them, so in total 32*3= 96 filters were used to extract feature maps [4]. There was a split on the dataset based on three applications that is brain MRI, breast MRI and cardiac MRI. The split ratios of them were 14/20, 14/20 and 4/6 respectively. Four results were obtained from the network, one when the network was trained only for brain MRI, second only for breast MRI, 3rd only for cardiac MRI and 4th when network is trained for all together. 25000 batch of images in a each task in used for training the network. [4].

Jie Chang et al. [5] have done segmentation of MR images of the brain using CNN for developing a method of automatic brain tumor detection. They have developed a two-way path model which contains one average pooling layer and other max pooling layer in different paths. Then, finally the CNN model is combined to a fully connected layer to predict optimized results. As **MRI** scans are very useful in detecting inner body dysfunction, the need of automatic region detection becomes an essential in the medical field. Pre-processing-Not having properly normalized and quantifiable pixel intensities interpretation is one of very big defects in MRI data. So, we need to preprocess the training data in order to extract meaningful quantifiable data. Firstly they reduced the noise data of scanner then they corrected the variance shift by the bias field distortion. Then They applied CNN by build a model using directed graph having nodes and edges where nodes represents the image pixels and edges represents the kernels which will be used to create feature maps on the image [5]. Here a perceptron which is multi-layered is designed specifically for the task of recognizing the 2D patterns in images with having a very high degree of image transformations like skewing, scaling, translations or any kind of distortion. Here each layer of CNN consists of 6 feature maps. Filters in this given CNN can be of any specified size. Here, random weight selection is done with the help of normal distribution of standard deviations. The input to the CNN is a raw EM image. Features of the image are automatically detected by the CNN. CNN automatically learns by adjusting its weights using a stochastic gradient descent learning algorithm [5].

Dan C. Cirean et al. [7] have segmented neuronal structures in the stacks of electron microscopy images. The label of each pixel in the image is predicted by the pixel centered at square window in which the pixel exists. Then the input layer maps each pixel to a neuron. Then convolution and max pooling layers are applied to preserve the 2D information and to extract features of the image. This approach outperforms the competing techniques with a large margin in all three categories [7]. The solution is based on DNN (Deep neural network) used as a pixel classifier. The probability of a pixel being a membrane is calculated.

Venugopal K.R. **et al.** [10] have combined CNN with CC (connected component) algorithm to segment the SEM images. As we know CNN is used to extract features directly from raw images with minimal pre-processing. Also, CNN is able to recognize patterns in the image which have not even provided as training data before. Provided it resembles one of the training data images. Accuracy (F-score) of the model is found to be 78% [10].They have done detection of neuron tissues in done on SEM (scanning images microscopy) dataset. SEM provides dataset of images whose resolution is almost perfect to identify the components in dense neuropil. Here, an automated approach to identify the affinity graph using CNN is discussed. Furthermore, this affinity graph can be combined with any partitioning algorithm to accurately segment the image. Neurons are particularly difficult to segment as they are branched, intertwined and very closely packed. Axons in neural structures are very thin that only electron microscopy has enough resolution to reveal them. In this approach all steps of parameter tuning is automated. So, in order to do automate and avoid human interaction in parameter tuning CNN has been introduced. Only CNN has the capability to identify features that has not even been introduced earlier, although it must resemble one of the training input samples. They have represented the CNN with the help of a directed graph where pixels are belonging to nodes and filters or kernels belong to the edges. In order to recognize 2D patterns with very high degree of images transformations techniques a CNN is used as a multi layer perceptron. Six features maps are constructed in each layer of this CNN model. All the filters in the CNN are given a size of 5 * 5. [10]. Random selection on the basis of normal distribution of standard deviations is one to calculate the weights of the edges in the model. The input to the CNN is a raw EM image. Detection of features in the image is done automatically by the CNN. CNN automatically learns by adjusting its weights using stochastic gradient descent learning algorithm.

Image segmentation is a very useful technique in medical area. As in MRI images or x-ray images, it is very difficult to identify different parts of body. Therefore, image segmentation helps in segmenting different part of the body from each other. **Moeskops P. et al. [12]** have applied image segmentation MRI (magnetic resonance imaging) images of brain, breasts and cardiac CTA).The classical procedure of segmentation is to train the model with some hand-crafted features and then use the model on input data to classify the objects, whereas in this approach CNN automatically extracts features which are required at the hand for the task of classification. Here CNN is used to classify medical images of knee cartilage, brain regions, pancreas and coronary arteries. In this work 3 orthogonal patches of 51 * 51 voxels were extracted for segmentation. For each of these patches, features were extracted using a deep stack of 25 convolution layers. Along with two fully connected layers and an output layer of 9 classes are implemented in the architecture of CNN. In convolution layer, 32 kernels for each input patch are used to extract features out of them, so in total 32*3= 96 filters were used to extract feature maps [12]. The training and test data was split for brain MRI, breast MRI and cardiac MRI into 14/20, 14/20 and 4/6 respectively. Four results were obtained from the network, one when the network was trained only for brain MRI, second only for breast MRI, 3rd only for cardiac MRI and 4th when network is trained for all together. Each network is trained using 25000 batches of images per task [12].

Bullock et al. [13] have segmented x-ray images to identify various regions in an x-ray image like - bones, soft tissues, open beam regions etc. This segmentation is required to quickly identify the region in an x-ray image. As all regions in an x-ray image are of similar shade therefore it becomes very crucial and precisely done task. X-ray image segmentation is an important task in many medical applications such as image enhancement, computer assisted surgery and anomaly detection. These applications normally require medical images to be segmented into three categories i.e. open beam, soft tissue and bone. Current methods rely on classical processing methods such as clustering, line fluctuation analysis or entropy-based methods which normally requires the tuning of hyper-parameters for each body part separately. They have designed a unique CNN architecture to perform the segmentation by extracting fine grained features, alongside with controlling the number of training parameters to control the over fitting problem [13].This model attains an overall accuracy of 82% (TF: true positives)

Shan E Ahmed Raza et al. [15] have used CNN based deep architecture for the segmentation of cells, nuclei and glands in fluorescence microscopy and histology images. They have used extra Convolutional layers so that is can work on variable input intensities and object sizes to make the model robust to noisy data. This model helps us to build molecular profiles of individual cells. Input data set is using fluorescent images which are usually noisy and it makes segmentation task difficult. Also, variable size of cells makes it more challenging for segmentation algorithms. This technique produces better results on such noisy data. And the results produced by this model helps in building molecular profile in multiplexed fluorescence images. It learns features at multiple image resolutions for better understanding which makes the model more robust and flexible to.

Cem M. Deniz et al. [16] have proposed a method to measure bone quality and access fracture risk. As we know, manual segmentation of Magnetic Resonance (MR) is time consuming and error prone process. So, in order to deal with this problem and to take full advantage of MR scan technology an automatic femur segmentation technique has been developed. This method is totally based on Image Segmentation using Convolutional Neuron Network. This method greatly helps in detection of proximal femur in bones. The data set used is Microsoft COCO. It contains 2.5 million instances of labeled images [16]. This dataset contains around 20000 images of training set and 2000 images of validation set along with 3000 images of testing set data. They used different types of CNN architectures to obtain the segmentation of proximal femur images. First they used 2D Convolutional network and 3D Convolutional network for segmentation and analyzed their performances. Then, the best performing model of both was improved by concatenating dilated convolutions with each time dilation having another value of dilation rate [16]. This technique helped in observing the effect of changing the parameters in designing the architecture of the model. Their model achieved the highest accuracy on segmenting the proximal femur images without post processing. It achieved the accuracy of 97.8% and exceeded the performance of 2D CNN by a large margin [16]. The performance of the model improved with adding of more number of layer and feature maps to the model.

ZhuolingLi et al. [18] have developed a method to adapt domain called **CLU-CNN**, which is specifically designed for medical images. As oppose to normal object identification task which are designed for pedestrian, cars or trees detection, here the task is detecting object in medical images like some deformities in eye cornea images or some object in lungs x-ray images etc. As we know, medical images have apparently very different features from normal images, so they needed to be handled differently and model training also needs to be done while keeping this thing in mind. Another problem with medical images is that deep neural networks suffer from a very big amount of parameters and this result in slow convergence rate. To overcome this problem sparse methodology and new learning rate schemes have been introduced. Also, medical images are too expensive to obtain, this causes the size of training data to be not enough as the general images training data. The size of training data might be as low as less than 400 which cause not covering all the possible cases while training the model. Also medical images may be taken at different hospitals and different environmental conditions, so images relating to same problem might have a lot of dissimilarity. As a result of these shortcomings with medical data set a very deep neural network model cannot be used, as it has a lot of parameters to be trained and would need a larger dataset to be able to work accurately. So as a result they have developed a framework named Clustering Convolution neural networks (CLU-CNN) which aims specifically at medical images segmentation task [18]. This model is relatively small and faster than the R-CNN. It consists of FCNN and Agglomerative Nesting Cluttering Filtering (ANSF) which is used to aim at medical single target image object detection. The two major contribution of their work are – one is domain adaptation method is developed requiring no extra training to be able to work efficiently and other is BN-IN net was designed by them to improve the model’s stability [18].

With the growing number of microscopic images automatic Nano particle detection has become an essential task to achieve. **AyseBetulOktay et al**. [19] have discussed a method to detect Nano particle in microscopic images using segmentation. They have proposed a method for the detection of Nano-particles and detection of their shapes and sizes with the help of deep learning algorithms. This method employees multiple output CNN and has two outputs. First is the detection output which gives the location of the Nano particle in the input image. Other is the segmentation output which outputs the boundary of segmented Nano particles. The final sizes of the Nano particles are determined by the Hough algorithm that works on the segmentation outputs. Here they have used **MOCNN** i.e. Multiple Output Convolutional Neural Network for the purpose of detecting, localizing and segmenting of Nano particles found in microscopic images [19]. MOCNN takes an input image as a window and it produces two outputs for it, from which one output tells us the location of the Nano particle in the image and second output tells us the distance of the object boundary to the window center.

Florent Marie et al. [24] have segmented the images of kidneys and nephron blastemal with the help of case based reasoning and CNN architecture. They have introduced two methods to achieve the required task. First one is CBR i.e. case based reasoning method for determining a region growth in process of image segmentation. And the second one is the method to perform image segmentation on a small dataset. Input images taken were are CT scans of kidneys of children having tumors in them. The result accuracy obtained is between 88 to 90% overall [24]. The CBR system that they have used comes from CT-scan and it searches for the closest image that has already been segmented in the base case. This technique helps in saving a lot of time and computations. In designing the CNN model the major problem arises is the need of a large training dataset. This however is not the problem in this case as here each tumor is unique and there is only a limited number of tumor segmentations available from the previous tests. Thus, they have developed a method to establish a dedicated deep neural network for each patient that will carry the task of tumor segmentations automatically without the need of human interventions [24].

Hai Xie et al. [25] have proposed a deep supervised Fully Convolutional Neural Network for the purpose of robustly segment the HEp-2 cell images data set. This Deep Supervised Fully Convolutional Neural Network (DSFCNN) is based on a very deep network which integrates a very dense deconvolution layer (DDL) and it also integrated the hierarchical supervision structure (HS) to able to start working as a whole. The high resolution of the original image is restored using the up sampling feature of the DDL which replaces the traditional way of up sampling using the deconvolution layer. Dataset used is 13A-2014 public dataset. Here, the purpose of the methods is to segment the human Epithrlial-2 cell images, which play a very important role in detection of anti-nuclear antibodies.

Mostefa Bennaceur et al. [26] have discussed the segmentation of MR images of brain using CNN for developing a method of automatic brain tumor detection. They have developed a two-way path model which contains one average pooling layer and other max pooling layer in different paths. Then, finally the CNN model is combined to a fully connected layer to predict optimized results. As MRI scans are very useful in detecting inner body dysfunction, the need of automatic region detection becomes an essential in medical field. One of the major defects in MRI is the absence of normalized and quantifiable interpretation of pixel intensities. So, they needed to pre-process the training data in order to extract meaningful quantifiable data. First step was to reduce the noise data of scanner and second was to correct the variance shift by the bias field distortion. CNN is represented as directed graph in which nodes correspond to image pixels and edges correspond to filters. CNN is a multi-layer perceptron designed specifically to recognize 2 dimensional patterns with high degree of skewing, scaling, translations and other forms of distortions [26]. Here each layer of CNN consists of 6 feature maps. Filters in this given CNN can be of any specified size. Weights are randomly selected from a normal distribution of standard deviations. The input to the CNN is a raw EM image. Features of the image are automatically detected by the CNN. CNN automatically learns by adjusting its weights using stochastic gradient descent learning algorithm.

Cem M. Deniz et al. [27] have proposed a method to measure bone quality and access fracture risk. As we know, manual segmentation of Magnetic Resonance (MR) is time consuming and error prone process. So, in order to deal with this problem and to take full advantage of MR scan technology an automatic femur segmentation technique has been developed. This method is totally based on Image Segmentation using Convolutional Neuron Network. This method greatly helps in detection of proximal femur in bones. Dataset used- The data set used is Microsoft COCO. It contains 2.5 million instances of labeled images [23]. This dataset contains around 20000 images of training set and 2000 images of validation set along with 3000 images of testing set data. They used different types of CNN architectures to obtain the segmentation of proximal femur images. First they used 2D Convolutional network and 3D Convolutional network for segmentation and analyzed their performances. Then, the best performing model of both was improved by concatenating dilated convolutions with each time dilation having another value of dilation rate [27]. This technique helped in observing the effect of changing the parameters in designing the architecture of the model. Their model achieved the highest accuracy on segmenting the proximal femur images without post processing. It achieved the accuracy of 97.8% and exceeded the performance of 2D CNN by a large margin. The performance of the model improved with adding of more number of layer and feature maps to the model.

2.2 Work done on identification or classification of objects in surroundings

Ahmed Bassiouny et al. [2] have used CNN to categorize images into different categories on giving an input data set of different scenes. Here, an image segmentation map of the image is created and the purpose was to represent the images by its segmentation map itself. Image segmentation map is a one to one mapping of each pixel of image to a set of labels. This approach not only identifies the objects contained in the scene but also identifies shape, location and size of the objects. They have first extracted the images based on suggestions given then created a feature vector for image by encoding the images according to that. Then image vector classification is done. Feature extraction may be of 3 type i.e. high level, mid level and low level feature extractions techniques are available. Attributes or features such as color, texture, contrast, hue etc. which describes the physical properties in an images are considered to be low level feature extraction. Mid-level features may include extracting the shape and spatial relations without actual knowing the semantics of the shapes. It just compares various shapes available in an image with already labelled images and tries to find the similarities. Then there comes the high-level image features, on which we are mainly focusing. They primarily include mapping of image representation to a meaning.

High resolution remote sensing data image segmentation is done by **Xiaomeng Fu et al**.[3] with the help of fully Convolutional neural network. It is shown that FCN works better than CNN in automatic segmentation of remote sensing images. The accuracy achieved with FCN in this model is above 85%. As we know with increase of satellites, there is a sudden increment in remote sensing image data. And it is almost impossible to segment those images manually to find useful results. Now as the deep learning technique has advanced, it is now possible to extract low level to high level features from raw input data images automatically. Hinton has explained in his work that deep learning methods can obtain much more useful features than the existing methods, and also has the good classification ability. FCN can accept the input image of any size and can deconvolute the data to obtain the feature map of last convolution layer. The first layer in model is an image of dimension h*w and having number of color channels as d [3]. The upper layers in CNN correspond to the coordinates of their connections in the image. What CNN does is that it distorts and translates the basic components of the image. (I.e. convolution, pooling and ReLu). Typical pattern identification networks use fixed size inputs on CNNs to produce non-spatial outputs, i.e. output only contains classification information. But with FCN input of any size can be taken and classification image map is produced as output classifying each pixel of the image. FCN network combined with matrix expansion technique for better efficiency is used to semantically segment the high-resolution remote sensing images [3]. The roads, plantation, buildings and water bodies are segmented accurately in the output results.

One more interesting work in this field is to recognize the marine organisms present near the sea surface in sear surface images. Nowadays, automatic robotic capturing of underwater marine organisms' detection is becoming important day by day. As we know all the seafood is captured by human divers, which makes it difficult and dangerous for their lives. Many of the divers get injured during seafood capturing. So, in order to detect areas having availability of marine organisms we needed an automatic way to detect and capture them. This R-CNN model can be used later on robots to accomplish the task of capturing the seafood without endangering human lives. **Jie Chang et al. [6]** have applied **R-CNN** to detect and recognize the marine organisms in image restoration sea surface images. R-CNN requires a lot of labeled training data to work accurately and it is impossible to find such a big data set of marine organisms. So, there are three data augmentation methods for underwater imaging has been proposed in this paper. A way of inverse process is used for the simulation of different turbulences in the marine environment. Then the performance of each data augmentation method is compared. Although there have been so many advantages of robotic capturing, and some of the techniques have already been applied in practical. For e.g. Norway developed an ROV (Remote Operated Vehicle) for submarine harvesting but it requires an experienced operator who knows where and how to reach a location where lot of marine organisms can be found. So, this object detection and machine learning algorithm is becoming quite successful during the past decade which helps in automatic object detection and recognition.

A fast R-CNN works in a way that it takes input as an entire image and an object set and the whole image is processed with the help of base CNN network . As a result a 512 dimensional feature map is produced by this RCNN model. Then the mapping of each object is done where a region is allotted to reach object where this object has interest according to feature maps. Then in order to convert the features of region interest to a fixed spatial extent of 7 x7, the ROI pooling layered uses the max pooling technique. Then the sequence of fully connected layer branch receives this 7 x 7 feature vector into its two sibling output layers [6]. As shown by the results this model is able to achieve around 60% accuracy with this implemented model.

M. Yang et al. [8] have used CNN on 3D input to recognize human activity**.** They have used CNN to process raw input, thus making process of feature extraction automated. This model works on 3D data as most of other CNN models only process 2D data. So it works on both spatial and temporal dimensions to recognize the activity in a scene. It works on several frames of input thus generates multiple channels of information. The final result is obtained by combining all the input channels together detecting the motion activity in sequence of frames. As we know recognizing human actions has variety of domains such as automated video surveillance, customer attributes recognition, and shopping attributes recognition. But to accurately identify the actions there may come some problems like clustered background, occlusions or view point variations etc. When video was taken certain assumptions about the circumstances of the situation are made because of the issues like small scale and changes in the viewpoint. But such kind of assumptions hardly holds in real world scenario. All these techniques follow conventional pattern recognition paradigm in which we have to manually extract features from the input the only we can give the input to classifier. But in our approach the model automatically identifies which features are needed to be extracted from the input on the basis of that input is fed to the model to identify the actions. The extraction of high level features to low level features that is indentifying a hierarchy of features can be done by Deep Learning models, thus this make the task of feature extraction automatic and time efficient.

Ji Shunping et al.[9] have used 3D CNN to automatically classify the crops in remote sensing images taken by satellites. In order to structure the multi-spectral remote sensing data, kernels are designed according to that. Also, fine tuning of 3D CNN's parameters are done in order to train the model with samples of the crops which leads the model to learn spatial and temporal representations. Full crop growth cycle samples are preserved for training the model again for the fully-grown crops. **Crop classification** from remote sensing images taken by satellites or **UAV**s (unmanned aerial vehicles) is becoming a fundamental task for yield estimation, economic assistance and crop transportation [9]. In addition to remote sensing data, SAR (synthetic aperture Radar) images data can also be used as input to the CNN model. Nowadays, deep learning is becoming the main algorithm for the classification tasks. As this method can automatically identify the feature representations with multiple levels (from low to high level), at low level – edges and shapes and at higher levels- structure, color and other detailed features.

Sadegh Karimpouli et al. [11] have used **CNN to segment rock images**. As segmentation in rock images in a critical step as images are present in grey-scale format. Conventional methods of segmentation such as thresholding or watershed algorithm don't work accurately with digital rock images data. As these methods work color contrast to segment the regions with each other, so the problem comes when two phases of similar color and contrast found to be side by side. Then these algorithms consider them as a single phase. Then several machine learning approaches have also been introduced but the Convolutional auto encoder was found to be producing more accurate results than others. However, available dataset for digital rock images is very small, but they have used data augmentation method to increase the size of training dataset to train the model. In order to create a training seed which consists of manually and semi manually segmented images is constructed with the help of a data set of images from Berea sandstone. Then the dataset is dividing among groups of training, testing and validation. From which only 10% of the data was required and used for validation purpose and another 10% was used for the testing purpose and rest of remaining data i.e. 80% was for used to train the model accurately [11]. Then this dataset is given to stochastic image generator which makes the size of dataset bigger to train the model more effectively. Then CNN model was implemented which provided results with an accuracy of 96% [11].

Yan Song et al. [14] have discussed segmentation and synthesis methods for sonar images. Also, SSS image **segmentation is done for which features are extracted using CNN of multi-pathways**. The advantage of CNN with multi pathways is that it can learn local and global features adaptively. SSS i.e. side sonar scanner provides essential high-resolution underwater images which is useful for many tasks like- mine detection, oceanographic research and rescuing underwater. SSS images can be segmented into areas like object highlights, object shadows or sea floor areas. But there is lot of noise and in homogeneity in SSS images that makes the task of segmenting SSS images very difficult and inaccurate. The method used here for segmenting SSS images combines feature of well-trained CNN of multiple pathways with ELM. This method allows model to use local and global features [14]. ELM is sort of a regression algorithm. Then applied methods describe the effectiveness of ELM in SSS image segmentation and synthesis.

Despite very accurate and one of the best ways of image segmentation by CNN, it does have some uncertainties which nobody likes to tackle. **Guotai Wang et al. [20]** have analyzed these different kinds of uncertainties for CNN based 2D and 3D image segmentation. Additionally, they have analyzed a test time augmentation-based uncertainty to analyze the effect of different kind of image transformations on segmentation outputs. They have proposed a test- time augmentation based uncertainties in order to monitor the effect various transformations on the input images. Their task consists of image and noise transformations. Then they compared and combined the proposed aleatoric uncertainty with uncertainty of the model. They concluded that test time augmentation based aleatoric uncertainty provides better results than any other kind of uncertainty calculation method.

Nowadays, UAV are most useful in monitoring of ocean areas rather than radars or satellites. As UAVs provides high resolution real-time images and this is not possible with other two sources. **Wang, S. et. al. [21]** have developed a technique to finding the Ulva prolifera regions around oceans by extraction of UAV images and applying image segmentation using CNN [21]. Here first of all raw images are processed with super-pixel algorithm to get the local multi-scale patches. Then CNN classification is applied on those patches. By combining super pixel algorithm segmentation and CNN classification they have produced much better results as compared to pixel level segmentation or instance aware segmentations. First of all, a reflection of light is removed as a pre-processing step to process raw UAV images. Then we choose super pixels elements extracted via energy driven sampling. It iteratively refines the super-pixels by updating their boundaries. Here energy function is defined as the sum of parts H(s) and G(s) [21]. "**E(s) = H(s) + y G(s)**" Where, H(s) is based on the likelihood of the color of super pixels. G(s) = is a prior of the shape of the super pixel. In the end they have compared different numbers of super pixels to evaluate and improve the accuracy and efficiency of the model.

Hao Wu et al. [22] have categorized the images into different categories according to the list of categories given with the input set. On giving a set of images and a set of categories the model will assign a category to each image from the list of available categories. The approach is to represent an image by its semantic segmentation map, which is a mapping from each pixel to one of pre-defined sets of labels. This approach not only identifies the objects contained in the scene but also identifies shape, location and size of the objects. They have first extracted the images based on suggestions given then created a feature vector for image by encoding the images according to that. Then image vector classification is done. Feature extraction may be of 3 type i.e. high level, mid level and low level feature extractions techniques are available. Attributes or features such as color, texture. Mid-level features may include extracting the shape and spatial relations without actual knowing the semantics of the shapes. It just compares various shapes available in an image with already labeled images and tries to find the similarities [22]. Then they have worked on the high-level image features, on which image segmentation technique mainly focuses. They primarily include mapping of image representation to a meaning.

Florent Marie et al. [26] have segmented the images of kidneys and nephron blastemal with the help of case based reasoning and CNN architecture. They have introduced two methods to achieve the required task. First one is CBR i.e. case based reasoning method for determining a region growth in process of image segmentation. And the second one is the method to perform image segmentation on a small dataset. Input images taken were are CT scans of kidneys of children having tumors in them. The result accuracy obtained is between 88 to 90% overall. The CBR system that they have used comes from CT-scan and it searches for the closest image that has already been segmented in the base case. This technique helps in saving a lot of time and computations. In designing the CNN model the major problem arises is the need of a large training dataset. This however is not the problem in this case as here each tumor is unique and there is only a limited number of tumor segmentations available from the previous tests. Thus, they have developed a method to establish a dedicated deep neural network for each patient that will carry the task of tumor segmentations automatically without the need of human intervention

Feng-ping An et al. [23] have proposed a computational model of feedback mechanism in Deep Convolutional Neural Network. They have developed two algorithms for this task, i.e. first is for learning and extracting the deep features of the image in order to construct feedback mechanisms for the CNNs. Secondly they have in order to segment the medical images they have used the model to classify the pixel block samples. As we know, medical image segmentation has received lot of attention from researchers and clinicians. Although there are already lot of medical image segmentation methods available but they focused machine learning method of image segmentation primarily. As there were already many types of semantic image segmentation available like region based, threshold based, and deformation model based or graph theory based but none of these methods proved to as efficiently accurate as deep machine learning method of segmentation. They have also solved feedback optimization problem with the help of greedy technique. They have proposed an optimization method which is used to update the feedback neuron state of each layer [23]. They have done this without the need of changing contribution coefficients. A bottom up method is used to optimize the objective function.

2.3 Our work

Here, in this work we have compared various image segmentation techniques and after researching various techniques we have found the CNN is one the most powerful tool in image segmentation techniques. Then we have applied CNN to the plants datasets to identify the diseases that might present in them. There are already various techniques to identify diseases in plants but either they are not automatic or they have very little accuracy in them. So, in order to provide good quality and accurate results we have developed a methodology that provides high accuracy as well as real time results which, helps in early detection of plant disease that can be cured within the time. We have developed a CNN based machine learning model which can segment an image into various parts and this helps also in reducing the number of tasks required for identification of diseases in plants.

`

CHAPTER 3

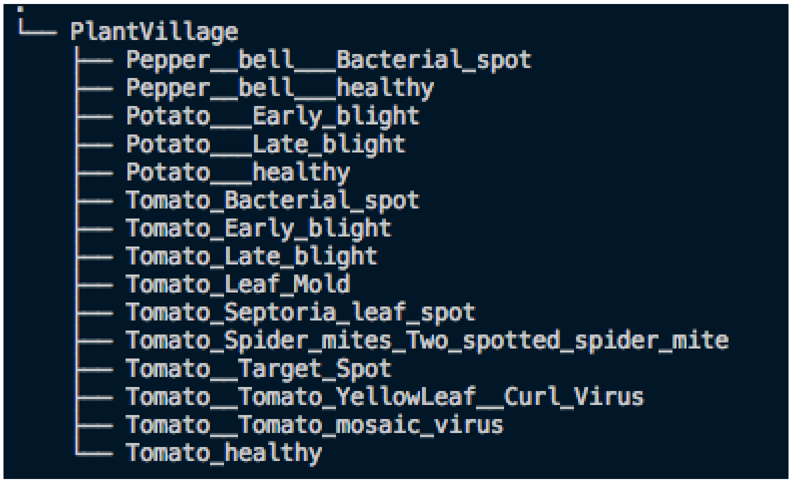
IMPLEMENTATION

3.1 Problem Statement

Here, we are trying to detect those plants which are suffered from some plant tissue disease by looking at their image samples taken from different angles. The main purpose of doing this task is to minimize the loss occurred due to disease affected plants which ultimately results in growth of crop production. For solving this problem, we are using the image segmentation technique with the help of CNN deep neural network architecture.

3.2 Dataset

The data has been collected from crowdAI from the Plant Village Disease Classification Challenge. It has been downloaded the dataset suing the link and command. The dataset consists of leaves of various categories of diseases found in plants. In this dataset each category consists of around 1000 images of same kind plant diseases. There are total of 15 kinds of diseases which are used to categorize input images. The data structure format is shown in the figure 3.1.



```
.
└── PlantVillage
    ├── Pepper__bell___Bacterial_spot
    ├── Pepper__bell___healthy
    ├── Potato___Early_blight
    ├── Potato___Late_blight
    ├── Potato___healthy
    ├── Tomato_Bacterial_spot
    ├── Tomato_Early_blight
    ├── Tomato_Late_blight
    ├── Tomato_Leaf_Mold
    ├── Tomato_Septoria_leaf_spot
    ├── Tomato_Spider_mites_Two_spotted_spider_mite
    ├── Tomato__Target_Spot
    ├── Tomato__Tomato_YellowLeaf__Curl_Virus
    ├── Tomato__Tomato_mosaic_virus
    └── Tomato_healthy
```

Figure 3.1 Input Dataset Structure

3.3 Sample input image dataset

The images which are collected in the dataset are all of different kinds of plants. Images having leaves of same kind of diseases are categorized into one folder. Later when model is training if any new input data comes having same features as of these data images, then that image will also get included in the existing database. And model gets smarter by the new features of the input image which may useful to categorize future input data. Some of the sample input image data is shown in the figure 3.2.

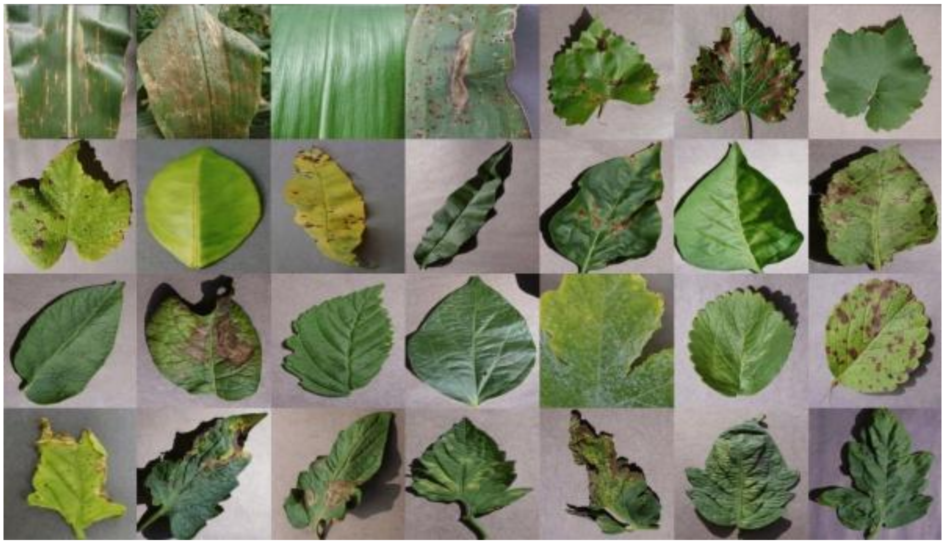


Figure 3.2- Example input image from the Dataset

3.4 Tools used

As we know, developing any project or thing in common requires some tools to be used. Here while working on this project we too needed some tools to be able to carry our work out forward. Tools which were used during the development of this project are given below.

3.4.1 Jupyter Notebook as an IDE

To be to work on machine learning project using python, it becomes easier to write code if you use an IDE. So, In order to carry out this work the code is written in python using Jupyter Notebook. Jupyter provides a good interactive development platform where viewing output and results become much easier and enhanced.

3.4.2 Keras python library

Keras is a python library which has functions needed to implement deep neural networks and other visualization of results techniques. We can use already created deep neural networks model and can train them using our dataset. By doing this we can be assured that our model has the efficient execution time and will be trained efficiently.

3.5 Technology used

There are various technologies available nowadays for doing any single task. But to choose the most optimal and suitable technique according to our need becomes the crucial task. Here, we are using machine learning deep neural network method in order to do image segmentation. Although, there are many other image segmentation methods available, but after deep analysis we have found that CNN is the most suitable method to do the image segmentation.

3.5.1 Machine Learning using CNN

We have used CNN in order to implement the solution for this problem, i.e. plant disease detection using image segmentation. This technology is becoming very famous in the field of machine learning. So, with the advancement in technology our approach is going to be matching the industry standards available.

3.5.2 Image segmentation

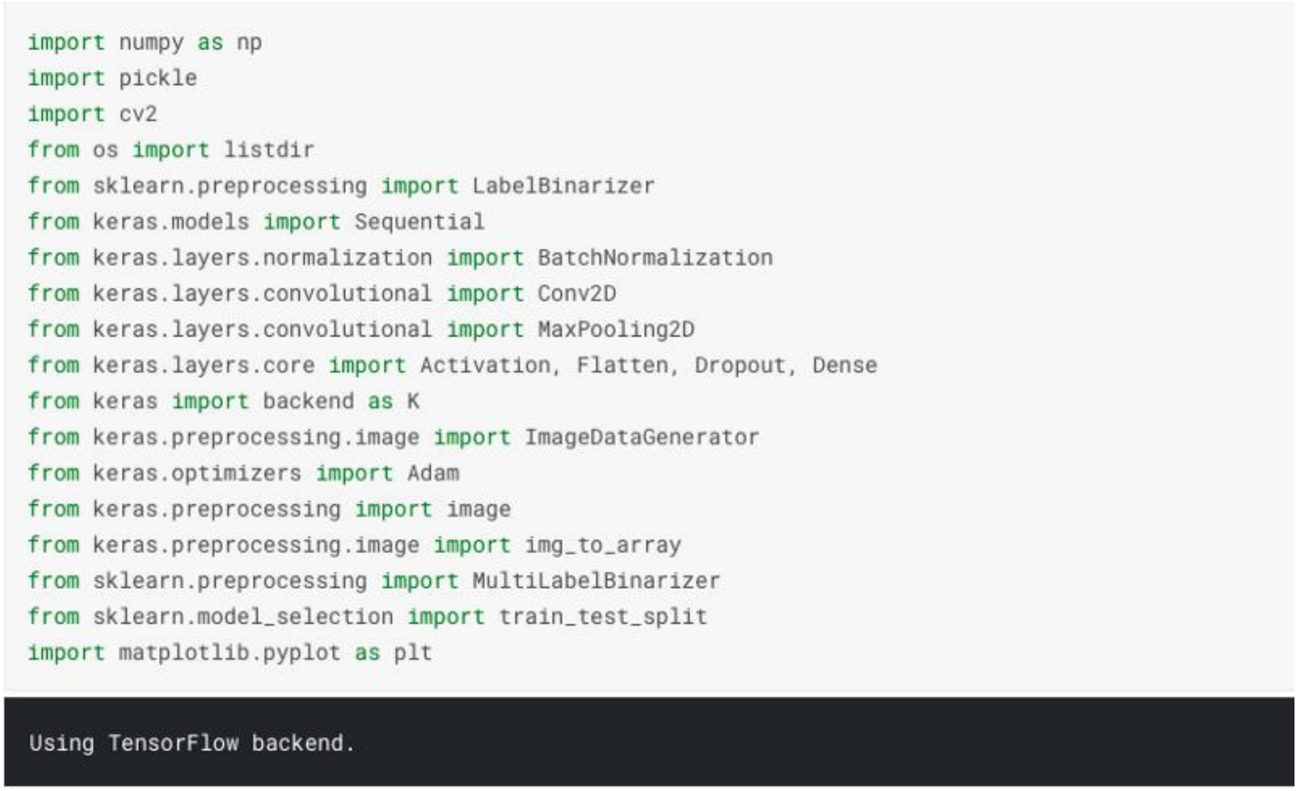
Image segmentation is the task of allocating each pixel of image to a class label. It is much different than mere identifying the object or recognizing them. It is to solve the problem that where exactly is that object present in the image. So, this task plays a very crucial role in identifying the diseases in the plant leaves, by which we can identify where in the image does the part of leaf is infected.

3.6 Steps of Implementation

Here, in this section we have discussed all the steps required to implement the project.

3.6.1 Importing Libraries

Here, we imported all the python libraries which will be required in developing the project.



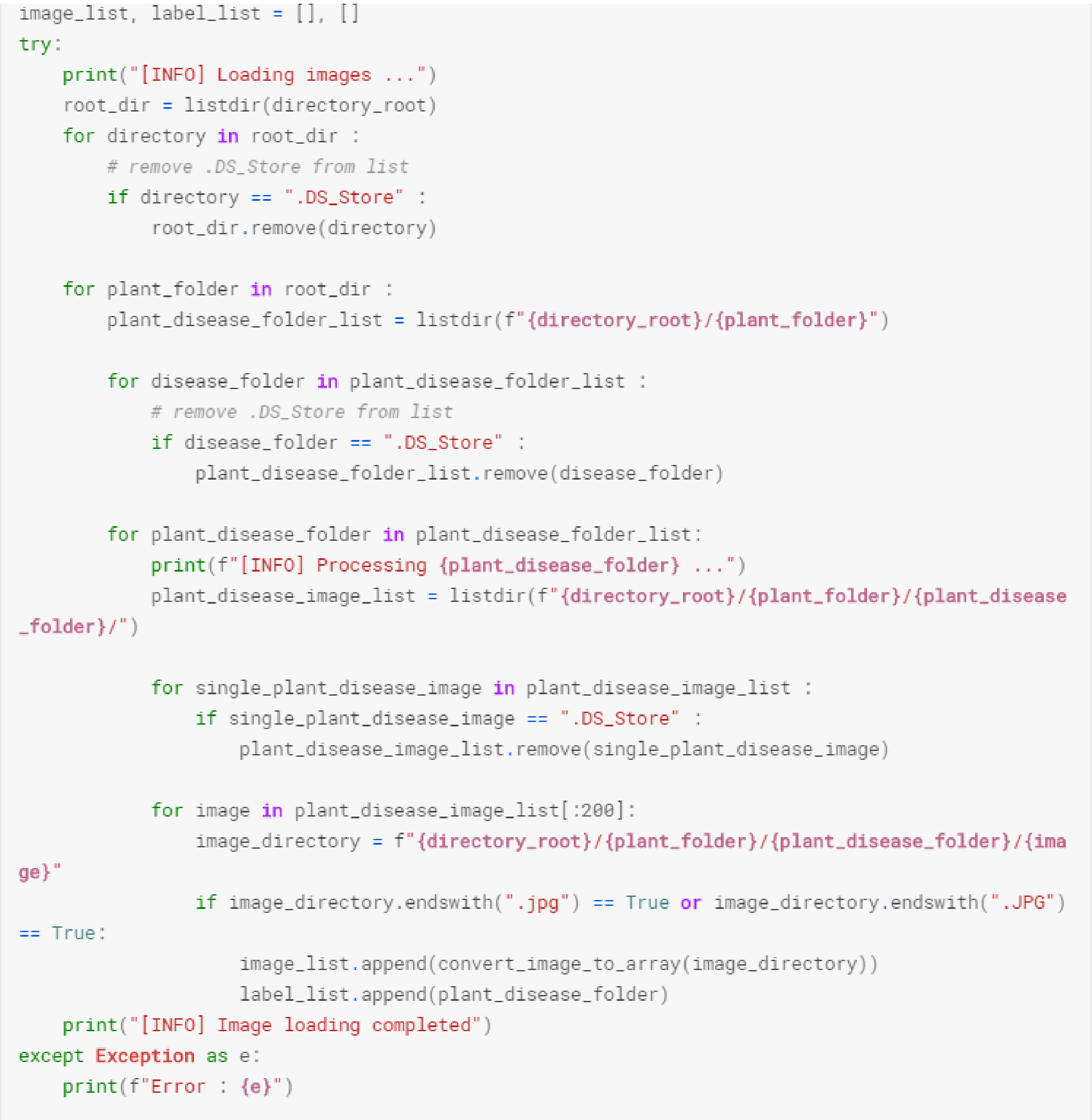
```
import numpy as np
import pickle
import cv2
from os import listdir
from sklearn.preprocessing import LabelBinarizer
from keras.models import Sequential
from keras.layers.normalization import BatchNormalization
from keras.layers.convolutional import Conv2D
from keras.layers.convolutional import MaxPooling2D
from keras.layers.core import Activation, Flatten, Dropout, Dense
from keras import backend as K
from keras.preprocessing.image import ImageDataGenerator
from keras.optimizers import Adam
from keras.preprocessing import image
from keras.preprocessing.image import img_to_array
from sklearn.preprocessing import MultiLabelBinarizer
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

```
Using TensorFlow backend.
```

Here, numpy library is used perform array operations on image pixel array. Keras libraray has almost all the features required to implement the deep convolutional neural network. Skllearn libraray is used to preprocessing on the input data. Preprocessing is also as important as training the model because without proper preprocessing the model may learn the input incorrectly and may start producing improper results. Thus, in order to get corret and accurate results pre processing needs to be done.

3.6.2 Loading Image Dataset

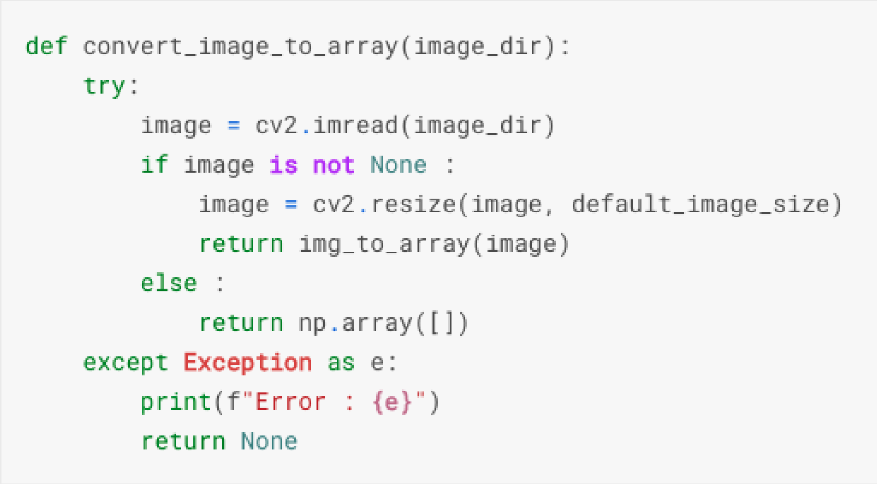
Here in this section we are loading the image dataset into the memory in order to feed in to the model for training. The data will stay in main memory till the training has been completed. Once the model has been trained then we can release the memory resources containing this dataset. This might take some time to process. The code shown below is taking 200 images from each folder to train the model. Then the images are being converted into array.



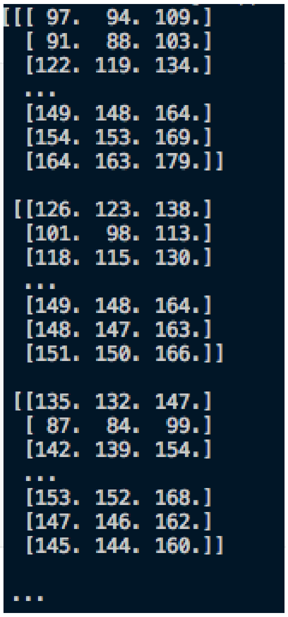
```
image_list, label_list = [], []
try:
    print("[INFO] Loading images ...")
    root_dir = listdir(directory_root)
    for directory in root_dir :
        # remove .DS_Store from list
        if directory == ".DS_Store" :
            root_dir.remove(directory)

    for plant_folder in root_dir :
        plant_disease_folder_list = listdir(f"{directory_root}/{plant_folder}")

        for disease_folder in plant_disease_folder_list :
            # remove .DS_Store from list
            if disease_folder == ".DS_Store" :
                plant_disease_folder_list.remove(disease_folder)

        for plant_disease_folder in plant_disease_folder_list:
            print(f"[INFO] Processing {plant_disease_folder} ...")
            plant_disease_image_list = listdir(f"{directory_root}/{plant_folder}/{plant_disease
_folder}/")

            for single_plant_disease_image in plant_disease_image_list :
                if single_plant_disease_image == ".DS_Store" :
                    plant_disease_image_list.remove(single_plant_disease_image)

            for image in plant_disease_image_list[:200]:
                image_directory = f"{directory_root}/{plant_folder}/{plant_disease_folder}/{ima
ge}"
                if image_directory.endswith(".jpg") == True or image_directory.endswith(".JPG")
== True:
                    image_list.append(convert_image_to_array(image_directory))
                    label_list.append(plant_disease_folder)
    print("[INFO] Image loading completed")
except Exception as e:
    print(f"Error : {e}")
```

3.6.3 Converting image to array

Here, images are converted into numerical arrays for the sake of calculations in the model. As the CNN mathematical model only understands numerical values. So, It will update its weights according to image pixel values obtained in the image array.



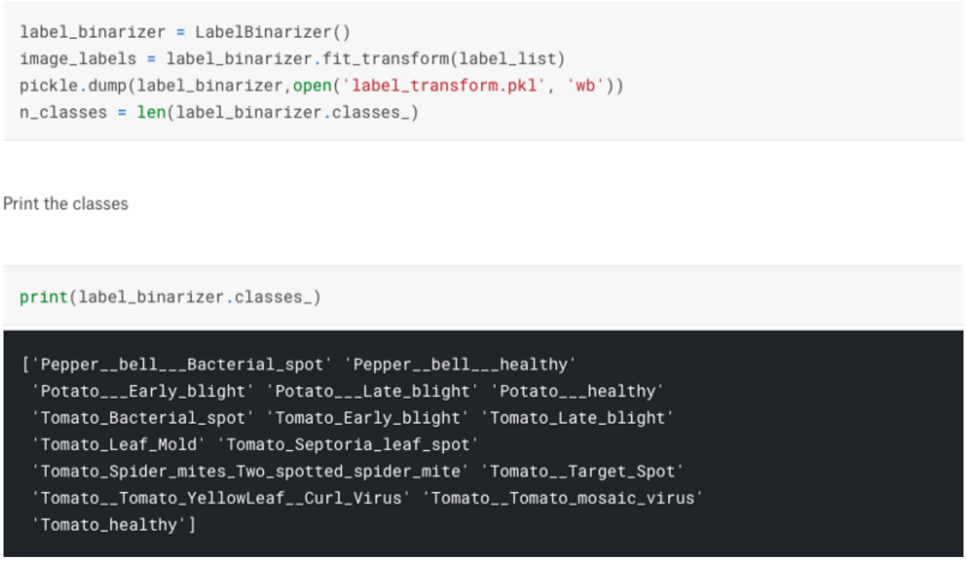
```
def convert_image_to_array(image_dir):
    try:
        image = cv2.imread(image_dir)
        if image is not None :
            image = cv2.resize(image, default_image_size)
            return img_to_array(image)
        else :
            return np.array([])
    except Exception as e:
        print(f"Error : {e}")
        return None
```



```
[[[ 97.  94. 109.]
  [ 91.  88. 103.]
  [122. 119. 134.]
  ...
  [149. 148. 164.]
  [154. 153. 169.]
  [164. 163. 179.]]

 [[126. 123. 138.]
  [101.  98. 113.]
  [118. 115. 130.]
  ...
  [149. 148. 164.]
  [148. 147. 163.]
  [151. 150. 166.]]

 [[135. 132. 147.]
  [ 87.  84.  99.]
  [142. 139. 154.]
  ...
  [153. 152. 168.]
  [147. 146. 162.]
  [145. 144. 160.]]

 ...
```

3.6.4 Converting Image labels into Binary levels

Image labels are converted into binary levels for the purpose that the model can be able to identify each label easily and fast. So binary representation for each label has been encoded and assigned to the label class. This also makes labeling the image less time consuming and easy to interpret.



```
label_binarizer = LabelBinarizer()
image_labels = label_binarizer.fit_transform(label_list)
pickle.dump(label_binarizer,open('label_transform.pkl', 'wb'))
n_classes = len(label_binarizer.classes_)
```

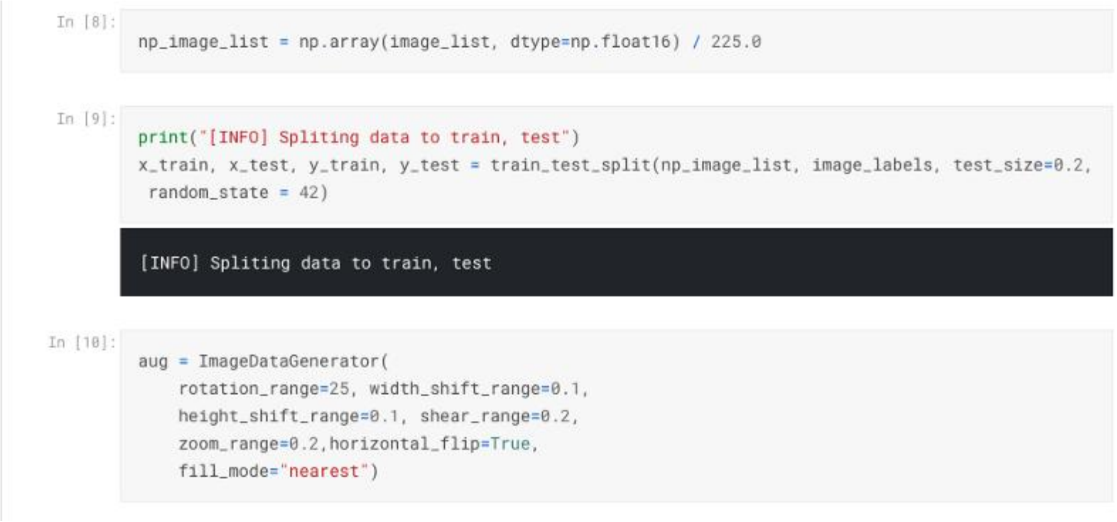
Print the classes

```
print(label_binarizer.classes_)
```

```
['Pepper__bell___Bacterial_spot' 'Pepper__bell___healthy'
 'Potato___Early_blight' 'Potato___Late_blight' 'Potato___healthy'
 'Tomato_Bacterial_spot' 'Tomato_Early_blight' 'Tomato_Late_blight'
 'Tomato_Leaf_Mold' 'Tomato_Septoria_leaf_spot'
 'Tomato_Spider_mites_Two_spotted_spider_mite' 'Tomato__Target_Spot'
 'Tomato__Tomato_YellowLeaf__Curl_Virus' 'Tomato__Tomato_mosaic_virus'
 'Tomato_healthy']
```

3.6.5 Splitting of Input

While designing a machine learning model it is always necessary to spit the input into train, test and validations sets. In order to train, test and validate the input data. Therefore, input is split into test and training sets, where the test set is taken as 20% of data and the train set is taken as 80% of the data.



In [8]:

```
np_image_list = np.array(image_list, dtype=np.float16) / 225.0
```

In [9]:

```
print("[INFO] Spliting data to train, test")
x_train, x_test, y_train, y_test = train_test_split(np_image_list, image_labels, test_size=0.2,
 random_state = 42)
```

```
[INFO] Spliting data to train, test
```

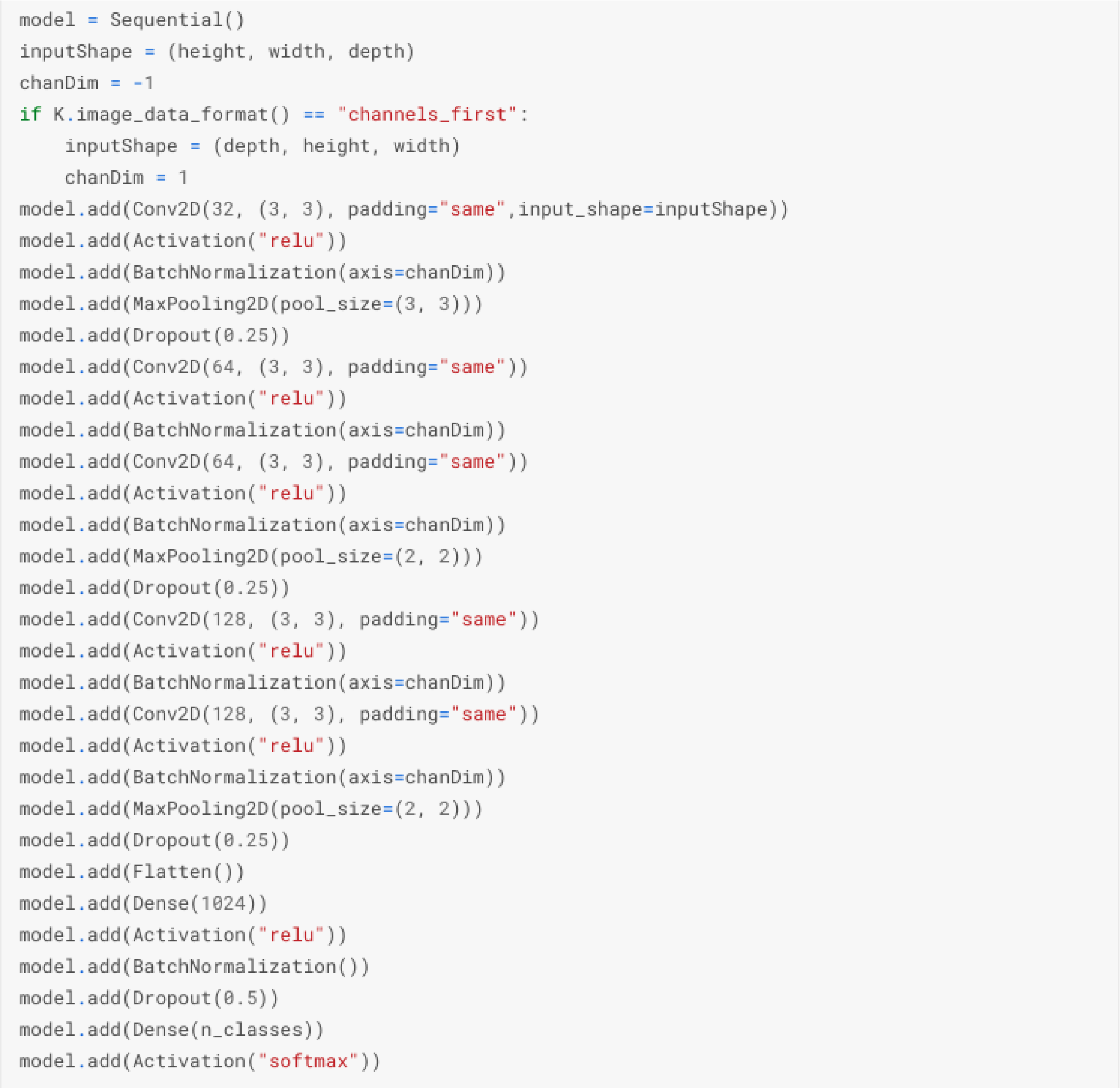
In [10]:

```
aug = ImageDataGenerator(
    rotation_range=25, width_shift_range=0.1,
    height_shift_range=0.1, shear_range=0.2,
    zoom_range=0.2,horizontal_flip=True,
    fill_mode="nearest")
```

Here, augmented input array is created after splitting the data into test, train and validation sets. Augmented array is created by translating the input images by rotating or flipping the images randomly. This helps model to learn the images with the need for them to align perfectly. The model can read and classify rotated or flipped images after that as well.

3.6.6 Model Building

In this step the model is being built, the layers of the model are defines. And the final model consists of a repeated pairs of convolution, pooling and ReLu layers. Here, in this model there are 5 convolution layers, 6 ReLu or Activation layers and 3 max pooling layers.

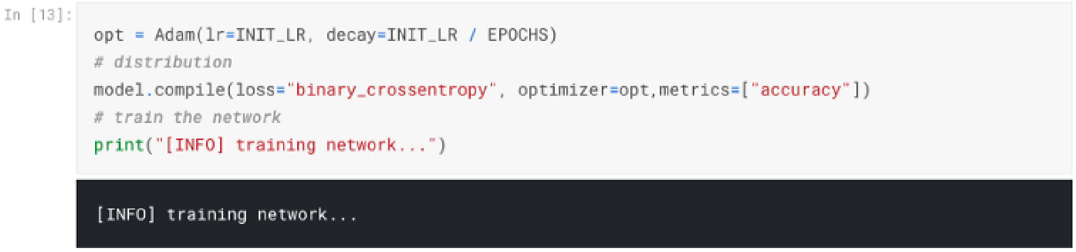


```
model = Sequential()
inputShape = (height, width, depth)
chanDim = -1
if K.image_data_format() == "channels_first":
    inputShape = (depth, height, width)
    chanDim = 1
model.add(Conv2D(32, (3, 3), padding="same",input_shape=inputShape))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(3, 3)))
model.add(Dropout(0.25))
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(1024))
model.add(Activation("relu"))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(n_classes))
model.add(Activation("softmax"))
```

3.6.7 Model compiling

Here, in this set the defined model is compiled as a one using ‘model.fit.generator’. The model starts to build after this statement which takes some time to completely train and build.

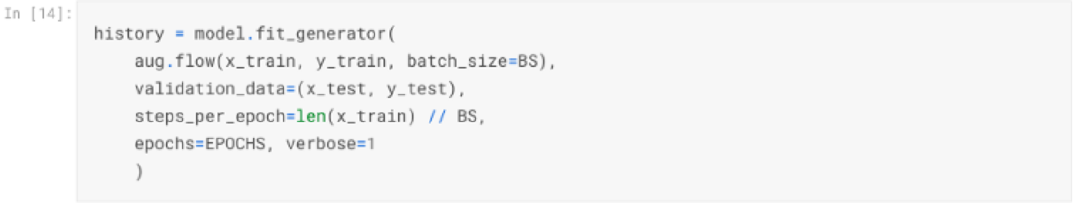
‘opt’ variable is used to store configuration for the model according to the training will be done. It contains number of epochs the model will do to train itself.



In [13]:

```
opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)
# distribution
model.compile(loss="binary_crossentropy", optimizer=opt,metrics=["accuracy"])
# train the network
print("[INFO] training network...")
```

```
[INFO] training network...
```

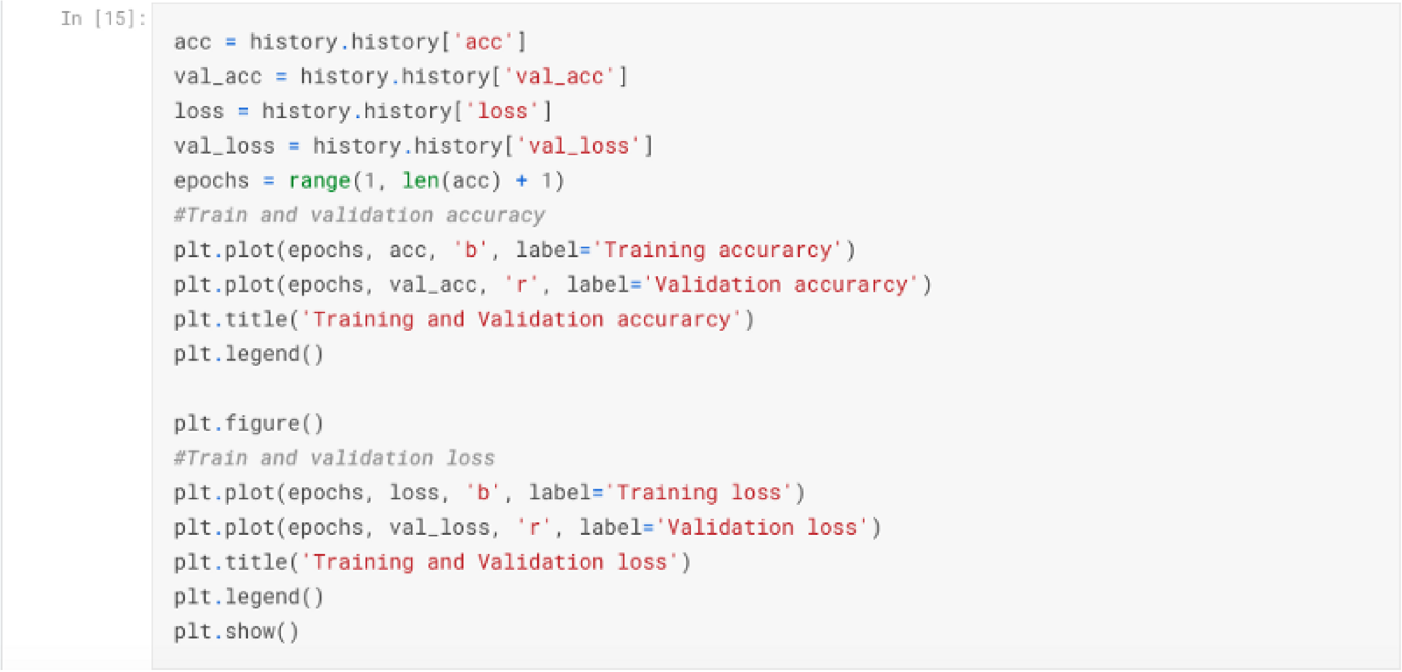


In [14]:

```
history = model.fit_generator(
    aug.flow(x_train, y_train, batch_size=BS),
    validation_data=(x_test, y_test),
    steps_per_epoch=len(x_train) // BS,
    epochs=EPOCHS, verbose=1
    )
```

3.6.8 Plotting the Graph

Here, in this step after building the model the results are plotted on to the graph. Here first of all training and validation accuracies are calculated after each epoc. And the growth in accuracy is plotted according showing the improvement in model training with time.



In [15]:

```
acc = history.history['acc']
val_acc = history.history['val_acc']
loss = history.history['loss']
val_loss = history.history['val_loss']
epochs = range(1, len(acc) + 1)
#Train and validation accuracy
plt.plot(epochs, acc, 'b', label='Training accurarcy')
plt.plot(epochs, val_acc, 'r', label='Validation accurarcy')
plt.title('Training and Validation accurarcy')
plt.legend()

plt.figure()
#Train and validation loss
plt.plot(epochs, loss, 'b', label='Training loss')
plt.plot(epochs, val_loss, 'r', label='Validation loss')
plt.title('Training and Validation loss')
plt.legend()
plt.show()
```

3.7 Graph plot results

Graph is plotted in order to visualize the performance of the model better. Here graphs for analyzing the accuracy and training loss is visualized with respect to the increase in number of epochs with time.

3.7.1 Accuracy Graph Plot

This graph represents the change in model's prediction accuracy using test set and validation set with the increase in number of epochs done while training the model.

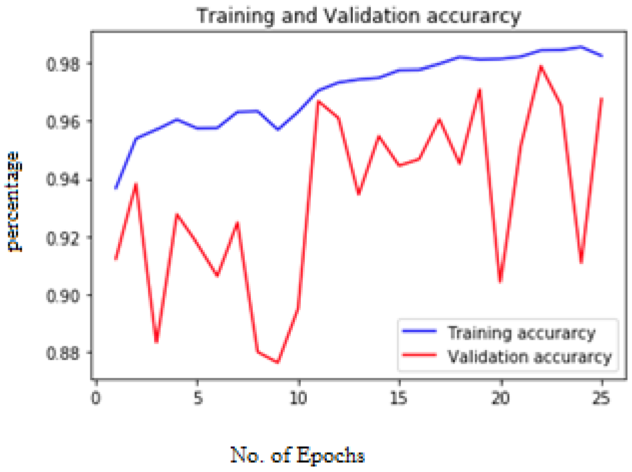


Figure 3.2 Training vs. Validation Accuracy vs. No. of Epochs Graph

Here, we can see how training and validation accuracy is increasing with the increase in number of epochs. With each epochs there is a small improvement happening in the accuracy.

3.7.2 Loss Graph Plot

This graph represents the change in the value of training loss using test set and validation set data with the increase in number of epochs.

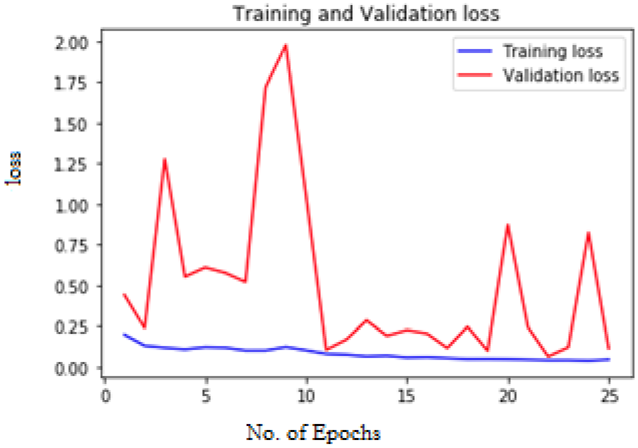
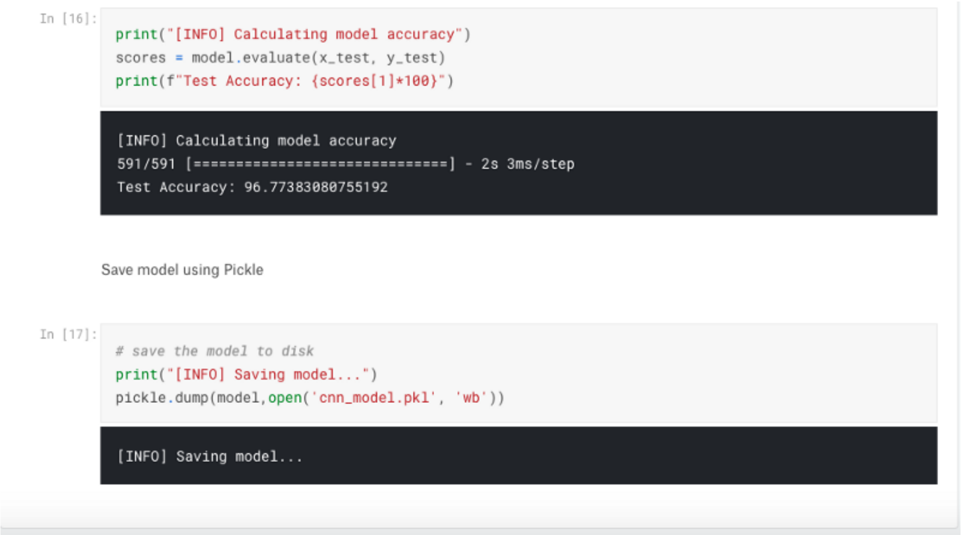


Figure 3.3 Training and validation Loss vs. Number of Epochs

Here, we can see how the validation loss and training loss reaches to zero with the increase in the value of number of epochs.

3.8 Calculating Model’s Accuracy

The model accuracy is calculated with the help of validation and test data and the model is saved for use again in future. The test set accuracy after the 25 epochs comes to be 96.77 %.



In [16]:

```
print("[INFO] Calculating model accuracy")
scores = model.evaluate(x_test, y_test)
print(f"Test Accuracy: {scores[1]*100}")
```

```
[INFO] Calculating model accuracy
591/591 [==============================] - 2s 3ms/step
Test Accuracy: 96.77383080755192
```

Save model using Pickle

In [17]:

```
# save the model to disk
print("[INFO] Saving model...")
pickle.dump(model,open('cnn_model.pkl', 'wb'))
```

```
[INFO] Saving model...
```

CHAPTER 4
RESULTS AND ANALYSIS

4.1 Comparison

Here, we are comparing various methodologies that can be used for image segmentation. By comparing we found that for the task where image sizes are small and there separate objects available, that is there is no overlapping between objects then region based segmentation is suitable. And for the cases where edges are clearly visible despite of having overlapping of objects there we may use edge based segmentation. And where there are many objects spread across the region there using cluster based segmentation would be efficient. And if we want most accurate results in segmenting the objects then we use CNN based segmentation.

Advantages			
Region Based	**Edge Based**	**Cluster Based**	**CNN Based**
Simple calculations.	Good for images having better contrast between objects.	Works really well on small datasets and generates excellent clusters.	Provides highly accurate results.
Fast operation speed.			
Disadvantages			
Region Based	**Edge Based**	**Cluster Based**	**CNN Based**
If there is no significant grayscale difference or an overlap of the gray scale pixel values, it becomes difficult to get accurate segments.	Not suitable when there are too many edges in the image.	Computation time is too large and expensive.	Takes comparably longer time to train the model.

TABLE 4-1 Comparison and Analysis of different Image Segmentation Techniques

4.2 Flow of action

‘Flow of work’ starts with getting know about all the image segmentation techniques available. After a detailed survey review of the existing methods have been done to decide the method to follow in order to achieve desired results. Then it comes to deciding the problem to solve with the existing latest technology. After that we collect the required data set needed to visualize the solution to the given problem. Then it comes the turn to develop the model by writing some efficient code and tweaking its parameters to reach a final model that can provide best results out of all.

4.3 Result and Accuracy

In this work, we get really good results with high accuracy. As the requirement of the problem was to predict whether a plant shows symptoms of any diseases, this makes our task very crucial and difficult as we don't have any prior knowledge of environmental conditions where those plants are growing. So, we can't take any kind of assumptions for any data. All analysis is to be done on the image virtual data. And as we know processing images for such complex task is itself a very big task to achieve. Our model does a very good job in solving the problem effectively. The accuracy provided by our model on given input set is around 96%. It may be possible that the model doesn't provide such good accuracy with further data sets taken randomly from around the world. Also, we know that in any work there is always space for a change or improvement. So here also there might be possibility that this model may not produce those much accurate results on some unknown dataset. But with proper sampled training we can make it learn any new dataset prior to use it for classification purpose.

CHAPTER 5

CONCLUSION & FUTURE SCOPE

5.1 Conclusion

In this work we have compared various image segmentation techniques and after researching various techniques we have found that the CNN is one the most powerful tool in image segmentation techniques. Detailed analysis of CNN is also done here explaining different layers and workings of each layer. We have explained all the possible advantages and fields where CNN can be used in our daily life. As we know CNN technology is at a boost of implementation nowadays in making the human life more and more convenient and less manual. Yet there is still a lot of work to be done in the making those automatic monitoring systems more accurate and reliable. There is a need to improve the accuracy of such systems to an extent that they can be relied upon to do crucial tasks such as monitoring unidentified activities in restricted areas such as country borders or ministerial offices, where a slightest inaccuracy may prove to be disastrous. Some work also needs to be done in the field of making various implemented model to combine as a one such that they can be fed to a robot by which it can act and do the tasks more intelligently and accurately In this work we have compared various image segmentation techniques and after researching various techniques we have found that the CNN is one the most powerful tool in image segmentation techniques. Detailed analysis of CNN is also done here explaining different layers and workings of each layer. We have explained all the possible advantages and fields where CNN can be used in our daily life. As we know CNN technology is at a boost of implementation nowadays in making the human life more and more convenient and less manual. Yet there is still a lot of work to be done in the making those automatic monitoring systems more accurate and reliable. There is a need to improve the accuracy of such systems to an extent that they can be relied upon to do crucial tasks such as monitoring unidentified activities in restricted areas such as country borders or ministerial offices, where the slightest inaccuracy may prove to be disastrous. Some work also needs to be done in the field of making various implemented model to combine as one such that they can be fed to a robot by which it can act and do the tasks more intelligently and accurately.

5.2 Future Scope

There is always room for improvement, innovation or change of existing technique in any research field. So, despite availability of so much quality research work in this field yet there is still a lot of work to be done in the making of those automatic monitoring systems more accurate and reliable. There is work needed to be done in handling the uncertainties where images are of bad quality or the boundary pixels of the segmented objects are overlapping. There is a need to improve the accuracy of such systems to an extent that they can be relied upon to do crucial tasks such as monitoring unidentified activities in restricted areas such as country borders or ministerial offices, where the slightest inaccuracy may prove to be disastrous. Some work also needs to be done in the field of making various implemented model to combine as one such that they can be fed to a robot by which it can act and do the tasks more intelligently and accurately.

CHAPTER 6

REFERENCES

[1] Olaf Ronneberger, Philipp Fischer and Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”, MICCAI 2015: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 pp 234-241.

[2] Ahmed Bassiouny and Motaz El-Saban, “Semantic segmentation as image representation for scene recognition”, 2014 IEEE International Conference on Image Processing (ICIP).

[3] Xiaomeng Fu and Huiming Qu, “Semantic Segmentation of High-resolution Remote Sensing Image Based on Full Convolutional Neural Network”, 2018 12th International Symposium on Antennas, Propagation and EM Theory (ISAPE) DOI: 10.1109/ISAPE.2018.8634106

[4] Pim Moeskops, Jelmer M. WolterinkBas, H. M. van der Velden, Kenneth G. A. Gilhuijs, Tim Leiner, Max A. Viergever and Ivana Išgum, “Deep Learning for Multi-task Medical Image Segmentation in Multiple Modalities” MICCAI 2016: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016 pp 478-486

[5] Jie Chang, Luming Zhang, Naijie Gu and Xiaoci Zhang, “A Mix-pooling CNN Architecture with FCRF for Brain Tumor Segmentation”, Journal of Visual Communication and Image Representation 58 · December 2018, DOI: 10.1016/j.jvcir.2018.11.047

[6] Hai Huang, Hao Zhou, Xu Yang, LuZhang, LuQi and Ai-Yun Zang, “Faster R-CNN for marine organisms detection and recognition using data augmentation”, Neurocomputing, Volume 337, 14 April 2019, Pages 372-384

[7] Dan C Cirean, Alessandro Giusti, Luca M Gambardella and Schmidhuber, “Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images”, Advances in neural information processing systems 25 · January 2012

[8] S. Ji, W. Xu, M. Yang and K. Yu, "3D Convolutional Neural Networks for Human Action Recognition," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 1, pp. 221-231, Jan. 2013.doi: 10.1109/TPAMI.2012.59

[9] Ji Shunping & Chi, Zhang & Xu, Anjian & Shi, Yun & Duan, Yulin.(2018). 3D Convolutional Neural Networks for Crop Classification with Multi-Temporal Remote Sensing Images.Remote Sensing. 10. 75. 10.3390/rs10010075.

[10] M. L.S., V.K. G. (2011) Convolutional Neural Network Based Segmentation. In: Venugopal K.R., Patnaik L.M. (eds) Computer Networks and Intelligent Computing. ICIP 2011.Communications in Computer and Information Science, vol 157. Springer, Berlin, Heidelberg

[11] Sadegh Karimpouli, Pejman Tahmasebi, Segmentation of digital rock images using deep convolutional autoencoder networks, Computers & Geosciences, Volume 126, 2019,Pages 142-150,ISSN 0098-3004,https://doi.org/10.1016/j.cageo.2019.02.003.

[12] Moeskops P. et al. (2016) Deep Learning for Multi-task Medical Image Segmentation in Multiple Modalities. In: Ourselin S., Joskowicz L., Sabuncu M., Unal G., Wells W. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016. MICCAI 2016. Lecture Notes in Computer Science, vol 9901. Springer, Cham.

[13] Bullock, Joseph & Cuesta, Carolina &Quera-Bofarull, Arnau. (2018). XNet: A convolutional neural network (CNN) implementation for medical X-Ray image segmentation suitable for small datasets.

[14]Yan Song, Bo He, Peng Liu, Tianhong Yan, Side scan sonar image segmentation and synthesis based on extreme learning machine, Applied Acoustics, Volume 146, 2019, Pages 56-65, ISSN 0003-682X

[15]Shan E Ahmed Raza, Linda Cheung, Muhammad Shaban, Simon Graham, David Epstein, Stella Pelengaris, Michael Khan, Nasir M. Rajpoot, Micro-Net: A unified model for segmentation of various objects in microscopy images, Medical Image Analysis, Volume 52, 2019, Pages 160-173,ISSN 1361-8415,

[16]Deniz CM, Xiang S, Hallyburton RS, Welbeck A, Babb JS, Honig S, Cho K, Chang G. Segmentation of the Proximal Femur from MR Images using Deep Convolutional Neural Networks. Sci Rep. 2018 Nov 7;8(1):16485. doi: 10.1038/s41598-018-34817-6. PubMed PMID: 30405145; PubMed Central PMCID: PMC6220200.

[17]Shan E Ahmed Raza, Linda Cheung, Muhammad Shaban, Simon Graham, David Epstein, Stella Pelengaris, Michael Khan, Nasir.Rajpoot,Micro-Net: A unified model for segmentation of various objects in microscopy images,Medical Image Analysis,Volume 52,2019,Pages 160-173,ISSN 1361-8415.

[18]Zhuoling Li, Minghui Dong, Shiping Wen, Xiang Hu, Pan Zhou, Zhigang Zeng,CLU-CNNs: Object detection for medical images,Neurocomputing,Volume 350,2019,Pages 53-59,ISSN 0925-2312,

[19]AyseBetulOktay, Anıl Gurses,Automatic detection, localization and segmentation of nano-particles with deep learning in microscopy images,Micron,Volume 120,2019,Pages 113-119,ISSN 0968-4328,

[20]Guotai Wang, Wenqi Li, Michael Aertsen, Jan Deprest, SébastienOurselin, Tom Vercauteren,Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks,Neurocomputing,Volume 338,2019,Pages 34-45,

[21]Wang, Shengke& Liu, Lu & Qu, Liang & Yu, Changyin& Sun, Yujuan& Gao, Feng & Dong, Junyu. (2018). Accurate Ulva Prolifera Regions Extraction of UAV Images with Superpixel and CNNs for Ocean Environment Monitoring.Neurocomputing. 10.1016/j.neucom.2018.06.088.

[22]Hao Wu, RongfangBie, JunqiGuo, Xin Meng, Chenyun Zhang,CNN refinement based object recognition through optimized segmentation,Optik,Volume 150,2017,Pages 76-82,ISSN 0030-4026,

[23]Feng-Ping, Liu Zhi-Wen,Medical image segmentation algorithm based on feedback mechanism convolutional neural network,Biomedical Signal Processing and Control,Volume 53,2019,101589,ISSN 1746-8094,

[24]Florent Marie, Lisa Corbat, Yann Chaussy, Thibault Delavelle, Julien Henriet, Jean-Christophe Lapayre,Segmentation of deformed kidneys and nephroblastoma using Case-Based Reasoning and Convolutional Neural Network,Expert Systems with Applications,Volume 127,2019,Pages 282-294,ISSN 0957-4174,

[25]Hai Xie, Haijun Lei, Yejun He, Baiying Lei,Deeply supervised full convolution network for HEp-2 specimen image segmentation,Neurocomputing,Volume 351,2019,Pages 77-86,ISSN 0925-2312,

[26]Mostefa Ben naceur, Rachida Saouli, Mohamed Akil, Rostom Kachouri,Fully Automatic Brain Tumor Segmentation using End-To-End Incremental Deep Neural Networks in MRI images,Computer Methods and Programs in Biomedicine, Volume 166, 2018, Pages 39-49,ISSN 0169-2607,

[27]Deniz, Cem & Hallyburton, Spencer & Welbeck, Arakua & Honig, Stephen & Cho, Kyunghyun & Chang, Gregory. (2018). Segmentation of the Proximal Femur from MR Images using Deep Convolutional Neural Networks. Scientific Reports. 8. 10.1038/s41598-018-34817-6.