A

PROJECT REPORT

ON

Automated Crop Pests and Diseases Detection using Deep Learning Techniques

SUBMITTED

TO

SHIVAJI UNIVERSITY KOLHAPUR

IN PARTIAL FULFILMENT OF THE DEGREE

BACHELOR OF ENGINEERING IN COMPUTER SCIENCE ENGINEERING

BY

GAMBHIR JINENDRA D.

PATEL NAVEEN M.

PATIL SHRINIVAS P.

TAKALE PRATHAMESH A.

Under the guidance of

Dr. A.M. CHOUHULE

SANJAY GHODAWAT GROUP OF INSTITUTIONS

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2020-21

A

PROJECT REPORT

ON

Automated Crop Pests and Diseases Detection using Deep Learning Techniques

SUBMITTED TO
SHIVAJI UNIVERSITY KOLHAPUR
IN
PARTIAL FULFILLMENT OF THE DEGREE

BACHELOR OF ENGINEERING IN COMPUTER SCIENCE ENGINEERING

BY

201808035	Jinendra Gambhir
2018080837	Naveen Patel
2017068677	Shrinivas Patil
2017068033	Prathamesh Takale

Under the guidance of

Dr. A.M. Chougule

SANJAY GHODAWAT GROUP OF INSTITUTIONS

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2020-21

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project report entitled

"Automated Crop Pests and Diseases Detection using Deep Learning Techniques"

Submitted by

201808035	Jinendra Gambhir
2018080837	Naveen Patel
2017068677	Shrinivas Patil
2017068033	Prathamesh Takale

is a bonafied work carried out by the above mentioned students under the guidance of Dr. A.M. Chougule and it has been completed successfully.

Sou. Sushila Danchand Ghodawat Charitable Trust's
Sanjay Ghodawat Group of Institutions
ATIGRE

Dr. A.M. Chougule
Project Guide

H.O.D.
Department of Computer Science
& Engineering
Sou.Sushila Danchand Ghodawat Charitable Trust's
Sanjay Ghodawat Group of Institutions

Dr. A.M. Chougule
HOD CSE

Jinendra Gambhir
Project Coordinator SGI

Acknowledgments

We express our sincere thanks to Dr. A.M. Chougule(H.O.D, CSE) whose supervision, inspiration and valuable guidance, helped us a lot to complete our project work. Their guidance proved to be the most valuable to overcome all the hurdles in the fulfillment of this project work.

Last but not the least, this acknowledgement would be incomplete without rendering our sincere gratitude to all those who have helped us in the completion of project work.

Sincerely,

GAMBHIR JINENDRA D.

PATEL NAVEEN M.

PATIL SHRINIVAS P.

TAKALE PRATHMESH A.

Declaration of honesty and integrity by student

I declare that this project report which is submitted for partial fulfillment for the degree of Computer Science Engineering for Sanjay Ghodawat University, Kolhapur is presented in my own words and where other's ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misinterpreted or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above cause for disciplinary action by the institute ad can also evoke penal action from sources which have not been properly cited or from whom proper permission has not been taken when needed.

Sign:

Name:

Roll No, class and Division:

Table of contents

List of Figures

List of Tables

Abstract

Agriculture plays a vital role in the Indian economy. India is the second largest producer of wheat and rice, the world's major food staples. India is currently the world's second largest producer of several dry fruits, agriculture-based textile raw materials, roots and tuber crops, coconut, sugarcane and numerous vegetables. It is necessary to provide an interface to farmers to identify this problem and get a solution on that particular problem from agriculture experts. We provide an interface in the form of android application to upload the images from their farm directly. As well as, we provide a web interface to display the uploaded images with their disease or pest type. The uploaded images and their locations stored on the web cloud storage and then classified into disease or pest type. Later this model will be implemented within the application for faster working of the whole ecosystem.

Keywords: Agriculture, Pests, Diseases, Machine Learning, Convolutional neural networks.

CHAPTER 1

INTRODUCTION

Sustainable agriculture is an important field where not much attention is given though it is highly necessary, to monitor the growth of crops for their efficient growth in most nutritious ways. For effective growth of crops, a lot of chemicals like fertilizers and pesticides are used, however, excessive usage of them results in damage to land and water resources. The attack of pests is a major criterion which affect crop yield. Various crop monitoring technologies are available which are highly expensive and not all farmers can afford. Moreover, in India, farmers are not capable of understanding the operation and handling of such sophisticated technology. In this project, we propose a system which is cheaper and easy to operate with multiple applications.

The major agricultural output of the state comes from the Konkan and Vidharbha region. These regions harvest banana, sugarcane, pulses, rice, wheat, black gram, green gram, groundnut, gingelly, maize and a variety of vegetables and fruits etc. and nearly 65.42% of the total workforce is dependent upon agriculture.

Insects are the big threats to agriculture industry. They are responsible for major kind of damages to growing crops. The first is direct injury to the plants caused by eating leaves and burrowing holes in the stem, fruits and roots. The second is indirect damage where insects themselves do little or no harm but transmit bacterial, fungal or viral infection to crops. It is necessary to control the pest growth as per crop's health concern so that at correct phase of time farmers get the solution on the pest problem that they are facing.

Disease and Pest detection can be done manually with the help of agriculture experts but it is time consuming process. As well as it is affordable for many farmers because lack of money and availability of experts. But using more pesticide for plants without analysing how much quantity of pesticide is needed for particular crop it is harmful for plant and human health. So, it is necessary to provide an interface which is accessible to all farmers which time saving and cost effective.

CHAPTER 2
LITERATURE REVIEW

In most agricultural systems, one of the major concerns is to deal with the diseases and pest attacks on it. Wide scale prevalence of diseases affects the production and quantity. This is where modern agricultural techniques and systems are needed to detect and prevent the crops from being affected by different diseases. The naked eye observation is the commonly
used method for detection of pest and identification of plant diseases. This needs continuous monitoring. But it is not practical in the case of large farm. Also, it is not accurate, expensive and time consuming. Here we propose methodologies that helps farmers for identifying crop diseases are discussed.

Author Manisha Bhange [1] mentioned her work on disease detection on Pomegranate fruit particularly Bacterial blight disease. Here web-based image processing approach was given focus on. They used 2 databases for disease detection, one for training and other for testing. For Image comparison CCV (Color Coherence Vector) was used. Along with this K-means clustering was used to train the data set as it gives great efficiency while dealing with large datasets. The experimental results varied upon the input images and stages of the disease. The average accuracy of the system was82%.

The author Aravind Rangarajan [2] suggested to use two pretrained deep learning models named AlexNet and VGG16 net to classify 6 different disease in tomato plants. Application of pre-trained deep learning models to classify new objects i.e., transfer learning was used. The above-mentioned models are more accurate as compared to Support Vector Machine algorithm which yielded an accuracy of 97.29% for VGG16 net and 97.49% for AlexNet. When provided with optimised datasets. Alexnet provided low execution time with good accuracy compared to VGG16 net.

Kamlesh Golhaniet.al [3] discussed importance of Neural Networks which can be used for disease detection. There are different NN models available which we can incorporate in our system. He also mentioned that to get accurate results the networks require best trainer sets of images.

The author Karen Lucero et.al [4] discussed common pests which attacked potato and bean crops. Mexican Bean Battle and Colorado Potato Beetle were the two pests. The pest images were collected from various sources. For the AI recognition techniques two neural classifiers were used namely RSC (Random Threshold Classifier) and LIRA classifier (has four layers: The S input layer; the I intermediate layer; the A associative layer, and the R output layer). Both perform feature extraction and were developed and programmed in C++. The RSC classifier was more stable in comparison to LIRA response.

Author M.P. Bange et.al [5] suggests a decision support system which is used to assist with integrated pest management with crop nutrition and other aspects. EntomoLOGIC is a system which converts number of insects in a particular area into densities of pest. It predicts the life stage and mortality of the insects using daily forecasted temperatures. It also calculates the predatory insects and shows predator to pest ratio. This dynamic to predict pest populations will this help to improve pest control timings and greatly reduce insecticide and pesticide usage.

Author Lin Jiaoet.al [6] here describes that traditional methods of agriculture pest detection have low efficiency and accuracy. To overcome this an anchor-free region convolutional neural network (AF-RCNN) is introduced for precise recognition of 24-classes of pests. Firstly, a fusion module is used to gain information especially on small pests. After all this, anchor-free region convolutional neural network (AF-RCNN) is employed to detect the different pests. Also, during training optimizations are done which improves localization accuracy of pests. These experimental results show detection is accurate and in speed.

Author Preetha Rajan et.al [7] has given a detailed analysis on Disease and pest detection using image processing. She has discussed about basic steps used for disease detection algorithm namely, 1. Image Acquisition 2. Image Pre-processing 3. Feature Extraction 4. Classification 5. Diagnosis. An overview on different Classification Techniques with their Advantages is also mentioned here. An important challenge faced here is using this concept in greenhouses, the lighting changes according to weather conditions and it affects color profile of the image. The author concluded that the SVM classification technique provided better results.

Yu Sun et.al [8] Suggested an image-based pest detection and monitoring system. Traps were place over a field/ forest which would capture the pests. The traps had a camera installed in them to acquire images and also had detector to detect the pests. For processing on these images, deep learning detector RetinaNet with some downsizing was used. The models were trained on end-to-end GPU workstation and deployed on embedded devices having minimal processing. This method showed promising results despite having limited computation and also introduced a applicable solution for the problem.

Author Yun Hwan et.al [9] has mentioned prediction of crop pests using machine learning technology. The paper introduces us Machine Learning algorithms like SVM (Support Vector Mechanism), Multiple Linear Regression, Neural Network and Bayesian Network. Multiple Linear Regression was used to 1. Predict Leaf Moisture 2. Evaluation of wheat pest. Bayesian Technique was used for pest prediction and for prediction of coffee rust disease. SVM was used for prediction of wheat stripe rust and of leaf miner infection. The author concluded that the techniques can be extended to apply for various crops.

Author Yanfen Li [10] here has introduced Deep Convolutional Neural Networks (CNNs) that can accurately recognize ten common species of pests. The dataset was collected and validated manually. Also, to remove complicated background form the images GrabCut and watershed algorithms were used. Many models like VGG-16, VGG-19, ResNet50, ResNet152, and GoogLeNet were investigated. Here GoogLeNet was found to be accurate, robust. Along with it an additional accuracy of 5.91% was obtained after the model was fine tuned.

M.Rajeswari [11] has focused on crop recommendation based on the soil type. The system is based on analyzing the nutrients of soil. The system is based on rough data sets and fuzzy logic is implemented to increase the accuracy. This method was tested for 24 different crops having 16 inputs which were based mainly on nutrients of soil. Algorithms like AQ algorithm, CN2 algorithm, LEM2 algorithms were used out of which LEM2 had an accuracy of 92% and was also able to predict more than the others.

Author Isabel M. del Aguila et.al [12] has introduced decision support systems that are incorporated into software which help to make rapid and effective decisions for efficient crop growth. A generic meta model having decision schema which estimates pest infestation for given crop was developed. This system was implemented for plants like grapes and tomatoes infested by frankinella occidentalis, white y and aphids.

Yong-Wei Bao [13] in the paper published describes about a web-based system covering Chongqing, China. Data is collected from various sources and Geographic Information System along with database management is used to monitor, analyse, visualize and store information related to rice diseases and pests in Chongqing. This data includes districts, rivers, roads. The other data was records of rice pests and diseases. By combining all this data, a database was developed and this was later incorporated into a web tool to access it.

Vijai Singh et.al [14] here describes about different leaf disease as which prevail on plants and also suggests a solution for automatic detection and classification of it. The main concept used here is image processing with the help of genetic algorithm. Genetic algorithms generate solutions for optimization problems and has advantages as mentioned by author. Image processing techniques are applied to images taken form camera pre-processing is done on them to obtain optimized results. Another advantage is that the plant diseases can be detected in early stages of development of disease.

AuthorYan Li [15] introduces application of multifractal analysis. This is mainly used in greenhouses to detect small-sized pests. This analysis was compared with other method as such as Watershed and EGIBS, but the multifractal combined with its minima threshold (MF MIN) was proved to be best option. Dynamic processes are required are required to deal with images as there are different lighting and weather conditions which can vary the image quality.

Author here [16] presented his work on classification of soyabean pest images. A full monitoring system which consists of UAVs which fly over field and capture images with high-res cameras collects data. Then they are passed to deep learning modules to calculate result. 5 deep learning architectures named Inception-v3, Resnet-50, VGG-16, VGG-19 and Xception were used for classification. A total of 5000 images were used for training and with fine tuning the model's greater accuracy was found. The Resnet-50 model was considered better having accuracy of 93.82%. Overall, this architecture can be used to support farmers in monitoring soyabean crops

Wayne Goodridgeet.al [18] here describes a system Multi-Criteria decision-making technique that can diagnose diseases in plants. A disease model is created which contains all the characteristics of one disease. The diagnosis is done by acquiring answers from farmers. The responses allow to narrow down to a particular disease model associated with that plant. Overall, this forms a dialogue-based decision support system to diagnose plant diseases. This disease modelling can be implemented to any plant type.

Author Jeremy P.M. Whish [21] here describes linking of two simulator models which work together to simulate pest populations and simulate crop growth to investigate biotic constraints within farming systems. Farming system framework Agricultural Production Systems simulator (APSIM) is internationally recognized as a highly advanced platform for modelling and simulation of agricultural systems. It contains a suite of modules that enable the simulation of systems for a diverse range of plant, animal, soil, climate and management interactions. DYMEX is a modular modelling package that allows the user to develop and run deterministic population models of biological organisms rapidly.

A.K. Tripathy et.al [22] describes about an experiment conducted to analyse Leaf Spot diseases on groundnut crop. This was done through wireless sensors and field-level surveillance on the groundnut crops. Sensors were used to collect all the micro level weather parameters and weekly surveillance data were made to create raw data. This data was then applied to data mining techniques (Naive Bayes classification with Gaussian distribution, rapid association rule mining and multivariate regression mining) to turn data into correlations to understand crop-weather-environment-disease continuum. This whole research contributed toward the reasons which were responsible for Leaf Spot disease on groundnut crops.

Author Fangyuan Wang [18] here says that the recent deep learning technique has a reduced accuracy problem while dealing with large scale, multiple pest data. To overcome this a Deep Pest, a cascading convolutional neural network architecture is used for small object detection. This approach gains prior knowledge of multi-scale contextual information of the images and builds up a context-aware network for initial classification of pest images into crop categories. After this a multi-projection pest detection model (MDM) is used to generate the super-resolved feature of field pests. According to the results obtained, Deep Pest was able to outperform other pest detecting methods.

Chowdhury R. Rahman et.al [19] in this paper it is mentioned that deep- learning based CNNs have improved image classification and so are used for detecting diseases and pests from rice plant images. Three different training models were implemented on two large CNN architecture (VGG-16 and Inceptionv3) and three small CNN architecture (MobileNet, NasNet Mobile and SqueezeNet). Also, a new concept of two stage training method derived from fine tuning the above models i.e., Simple CNN which performed well in real life scenario and had an accuracy of 93.3% and had reduced model size.

Author Divya Sindhu [20] has given importance on image processing technologies for detection of plant diseases. These technologies include computer vision image processing (CVIP), colour co-occurrence matrix (CCM), neural network classifier, fuzzy clustering and image segmentation algorithms etc. Also, for detection of diseases artificial intelligence, artificial neural network, Bayer as classifier, fuzzy logic and hybrid algorithms have been used. Using all this method accurate detection and classification of diseases on different parts of plants were obtained with precision. Another advantage was the diseases at its initial stage can also be identified.

Author Jingcheng Zhang [23] here introduces us to remote sensing methods used to monitor plant diseases and pests. The content in the paper summarizes developments in Remote Sensing systems, features, monitoring algorithms for monitoring plant diseases and pests which include: visible near infrared spectral sensors (VIS-NIR); fluorescence and thermal sensors; and synthetic aperture radar (SAR) and light detection and ranging (Lidar) systems. Using the data acquired from these systems helps in detection and monitoring systems. The paper has also reviewed the algorithms used to identify, differentiate and determine the severity of plant diseases and pests.

Author Peng Chen [24] here has mentioned problem of cotton pests. To work on this recurrent neural network (RNN) was used. A bi-directional RNN with long short-term memory (Bi-LSTM) was used to predict cotton pests and diseases considering climate factors. It then adopted full connected layer to calculate the output of Bi-LTSM layer to gain final result. This model was built to predict cotton pests and diseases considering climate factors in future. The results also showed that the model was accurate and had an advantage in processing a small dataset and time-dependent problem.

Author Trond Rafoss [25] has introduced a Web Feature Service Transaction (WFS-T) which was implemented in GPS-enabled mobile phones to track down disease outbreaks in plants. The mobile phones were communicating with a GeoServer in backend through internet. The GeoServer stores all the information about the invasive plant disease in an area and displaysit on a map on user side.

Oluwafemi Tairuet.al [26] built a plant disease detection model using CNN for a Hackthon in 2018. He has mentioned a structured manner on how he built the project, what were the problems occurred and the working of the project. He also later deployed this model on an app using an API.

Author BurhanudinSyamsuri [27] here describes implementation of image recognition for detecting symptoms of plant diseases using deep learning CNN. This system was also implemented as app on mobile devices but choosing the right model for high accuracy and low resource consumption was the challenge. This experiment was carried out on coffee plants using three CNN models. Two models MobileNet and Mobile Nasnet (MNasNet) were specially used for mobile, and the remaining one was InceptionV3 for computers. When the results were compared, there was slight degradation in accuracy when executed on mobile. InceptionV3 was accurate of all but had biggest latency time compared with other two.

B Nithya Ramesh et.al [28] has given importance on image processing technology and using it to classify the disease using diseased images and study of classification algorithms. Discrete Wavelet transform has used for feature extraction of images and Support Vector Machine has used for classification using neural network analysis. Combining above methods, the results were promising and there was accurate detection of diseases.

Author Lawrence C. Ngugi [29] has mentioned importance on modern agriculture techniques for pest and disease detection. Main concepts which author has given importance on are Image processing and use of CNN for detection of disease. It has been proved that if sufficient data is provided for training, deep learning techniques are able to recognize pest and diseases accurately. The performance of 10 CNN models were compared based on 7 performance metrics which showed that DenseNet201, ResNet-101 and Inceptionv3 CNN architectures were the most suitable models for use in normal computing while ShuffleNet and SqueezeNet were the best suited architectures for mobile and embedded applications. Having a large dataset also tend to improve the accuracy of models.

Author Alvin R. Malicdem [30] has mentioned his views on rice blast disease in tropical regions of the world. Data from government agencies was obtained and modified for building predictive models. Weather conditions which also affect the disease was also taken into account. Using all these features, to predict the occurrence of rice blast disease Artificial Neural Network (ANN) and Support Vector Machine (SVM) classifiers were used. These models provided accurate results, but the SVM model showed more accurate prediction.

Author M.G. Hill [31] studied and demonstrated usefulness of Machine Learning algorithms to predict leafroller disease on kiwifruits. Five ML algorithms (Decision Tree, Bayes, Random Forest, AdaBoost, Support Vector Machine) were used to forecast insecticide application for leafroller disease by predicting the amount of insecticide was sprayed or not. Orchard management attributes were important for models for forecasting accuracy.

Author Cheng-Long Chuang [32] here mentions about the pests The Oriental Fruit Fly (OFF), Bactrocera dorsalis (Hendel)which have caused hefty damage on fruits in Taiwan. It has been found that the pest was corelated to weather conditions and previous population of pests. Using this weather and population data and incorporating them into analysis program using Vector Auto-Regressive and Moving-Average model with eXogenous variables (VARMAX) was proposed to disclose the mechanism between the pests and climate factors. The model provided 7-day forecast for the population of Oriental Fruit Fly. This model was useful for understanding the dynamics of Oriental Fruit Fly and other pest related problems.

CHAPTER 3
PROBLEM DEFINITION AND SCOPE

3.1. Problem Definition

Crop production rate depends on geographical region, weather condition, soil type, soil composition and harvesting methods. However, this rate is also mainly affected by attack of different pests, and other crop related diseases. Crop diseases can be fungal, bacterial, viral and can damage plant parts above or below the ground. Identifying symptoms and knowing when and how to effectively control diseases is an ongoing challenge for many farmers.

It is necessary for farmers to control the growth of disease and pest affected areas before it damages entire crop. Also, the use of excessive and below required amount will affect the human health and crop health respectively. Therefore, to overcome this problem alternative way is to design a system that will provide an interface for farmers to upload images directly from field ad store on cloud storage. As well as design a website to display stored data according to disease type and pest type. Identify diseases of the plants and pest on the plant automatically by capturing images of affected area using machine learning techniques.

3.2. Proposed system

The goals here are to build an entire ecosystem where farmers are able to share the crop images, detect the pest and diseases and consult with a scientist in their area. This all will be accessible to farmers from their mobile phones. Since we don't have enough dataset images to train the machine Learning model, Farmers will be helping us collecting the same.

Images clicked by camera or images through mobile gallery is stored on the web cloud storage. The website will be used by the admins to view and sort the images which meet the image requirements. Later, these images are fetched on the websites and given as input to the machine learning and CNN model. Firstly, it will extract only disease and pest affected leaf part from the image and stored in the training data set. After CNN classification training model will detect whether image contains disease or pest. According to type of disease and pest it will label it.

3.3. Advantages:

1 Data Set Collection

Farmers will indirectly be helping us to collect data set on different crops and label them according to Crop type, Diseases and Pests. This process will go on for different crop cycles throughout the year and will help us make a strong data set which will be very useful.

2 Quick Detection

As the detection services are implemented directly on mobile, farmers will be able to detect pest and diseases withing fraction of seconds.

3 Take fast action on crops

As the diseases and pests on crop can be detected quickly, farmers can take spontaneous measures to prevent the spread of disease on other crops and cure the crops.

4 Minimize losses

Crop expenditure will be minimized and the produce obtained will be more due to preventive measures. Hence, losses will be minimized and good quality produce can be obtained.

5 Soil fertility will be maintained

Since there will be controlled use of pesticides and fungicides, they will not affect the soil fertility which will help to obtain good produce throughout years.

6 Quick contact with agricultural scientists

The application will come with a scientist allocated for a certain area. They will provide support to farmers directly through chat interface. Any difficulties faced by farmers will be solved in less amount of time.

CHAPTER 4

METHODOLOGY

4.1 Methods

4.1.1 Android application (Agrikanti)

It is necessary to collect the images from the farmers from their field to detect whether the plant has pest or disease. For that purpose, an android application is provided to farmers through which farmers can directly upload their images to cloud storage. This android application has two options to upload the images

(a) Image can be uploaded at real time.

(b) Image can upload from the gallery of mobile phones.

To get access to the application user needs to register using their mobile number. After it, to upload images first user has to enable location of device so that location is also send to cloud storage. To upload images from gallery, gallery access permission should be granted too. As well as user can keep track of uploaded images so it is beneficial for analysis purpose.

4.1.2 Website (Agrikanti.in)

Images uploaded through mobile application are stored on cloud storage and visible on web application. Also, website user can upload images from the website it provides multiple select upload option. As well as website provides services like delete images, location of the image. Admins of website can remove images if they don't seem valid. They can label and sort the images according to the diseases and pests. We provide machine learning and CNN model which classify the uploaded images according disease or pest type. After classification it will display on the website in the tabular format according to disease or pest type.

4.1.3 Machine Learning

4.1.3.1 Collection of Data

We developed android application ‘Agrikanti’ to collect images of plant diseases and pest. Images were collected directly from the farm. Anyone can upload the images from their smartphone which will store on cloud storage. Also, BAU (Bihar Agricultural University) also provided many images. We got nearly 3000 images of Rice pest and diseases. The given rice pest dataset consists of 1334 train images and 341 validation images with 11 classes as following:

1. BPH
2. Brevennia rehi (rice mealybug)
3. Eublemma
4. Gandhi Bug
5. Grass hopper
6. Horn Catpilar
7. Cnaphalocrocis medinalis
8. Rice Skipper Pelopidas mathias
9. Spoladea recurvalis,Beet Webworm
10. Stem Borer
11. Yellow Tail Moth Euproctis similis

4.1.3.2 Data Pre-processing

To identify the portion of rice plant infection, we will use images of 11 classes. The images are in RGB format that’s why we will use 3 as image input parameter. We have images of 11 types of pests captured directly from the field at various angles. Before classification images should undergo from augmentation. The given data is not splitted into training and validation format. We have nearly 1500 images of rice pests and all images were taken in way to avoid overfitting conditions. We have data which is not divided into train and validation data. We used split-folders python library to split data into the ratio of 70-30% (train-val). It takes each folder from dataset and split it into a specified ratio.

Fig 1.4.3. (a)BPH, (b)Brevennia rehi (rice mealybug), (c)Eublemma, (d)Gandhi Bug, (e)Grass hopper, (f)Horn catpilar, (g)Cnaphalocrocis medinalis, (h)Rice Skipper, (i)Beet (j)Webworm, (k)Stem Borer, (l)Yellow Tail Moth

4.1.4 Kisaansaathi (Android Application)

This is the final application of the ecosystem. It is a support system built for farmers which follows a Scientist – Farmer approach. The CNN model trained will be implemented in this application. This application requires location access and camera permissions for smooth working. Farmers can click photos of the affected crop to identify what disease it is suffering from. Along with it, we have created a chat interface for farmers to directly talk to agricultural scientists. This works by taking location of the farmer and allocates 3 agricultural scientist which are close to the farmer's area. Using the chat interface, farmers can directly ask their queries and obtain solutions. They can also upload images in case they need a visual representation/confirmation.

CHAPTER 5
SOFTWARE REQUIREMENT AND SPECIFICATION

1. FUNCTIONAL REQUIREMENT:

5.1.1 Agrikanti application:

1. The application should run smoothly on any type android phone.
2. User must be able to log in using their phone number which after sends an OTP for verification.
3. It must be able to access location, camera and gallery of the phone.
4. User should be able to select crop type and upload images accordingly.
5. Image should be uploaded with proper metadata associated to it.
6. User must be able to see the number of uploads they have made, the location and its id on the home screen.

5.1.2 Agrikanti Website:

1. Website users/admins should be able to access the website with their given credentials.
2. The website dashboard should display real-time statistics about the all the images.
3. Admins should be able to list, view all the images which are stored on google cloud storage.
4. They should be able to modify, change tags and delete the images.
5. Users should be able to upload images from the upload image section; they can upload zip files as well as standalone images.
6. Admins should be able to create new login access for other users.

5.1.3 Kisansathi Application

1. Users should be able to login as Admin/Farmer according to their role.
2. Farmers must be able to send location and get Scientists close to their location.
3. Scientists must be able to view all the farmers.
4. Farmers and Scientists should be able to chat via the chat interface provided to them.

5.2. NON-FUNCTIONAL ATTRIBUTE

5.2.1 Hardware for mobile:

1. A standard Android OS phone, version 5.0 (Lollipop) or above.
2. Minimum 5 MP (mega pixels) camera with a clear lens.
3. Minimum RAM of 2 GB or above and storage of 16 GB.

5.2.2. Hardware for standard computer:

1. Processor: Intel core i3.
2. RAM: 4GB minimum required.
3. A standard web browser.

5.2.3. Hardware Used for Machine Leaning system:

A standard AWS instance with specification of

1. Processor: Intel(R) Xeon(R) CPU @ 2.00Ghz.
2. GPU: Accelerator-GPU (Tesla P100).
3. RAM: 16 GB.
4. Storage: HDD of 74GB.

CHAPTER 6

SOFTWARE DESIGN

6.1. SYSTEM ARCHITECTURE

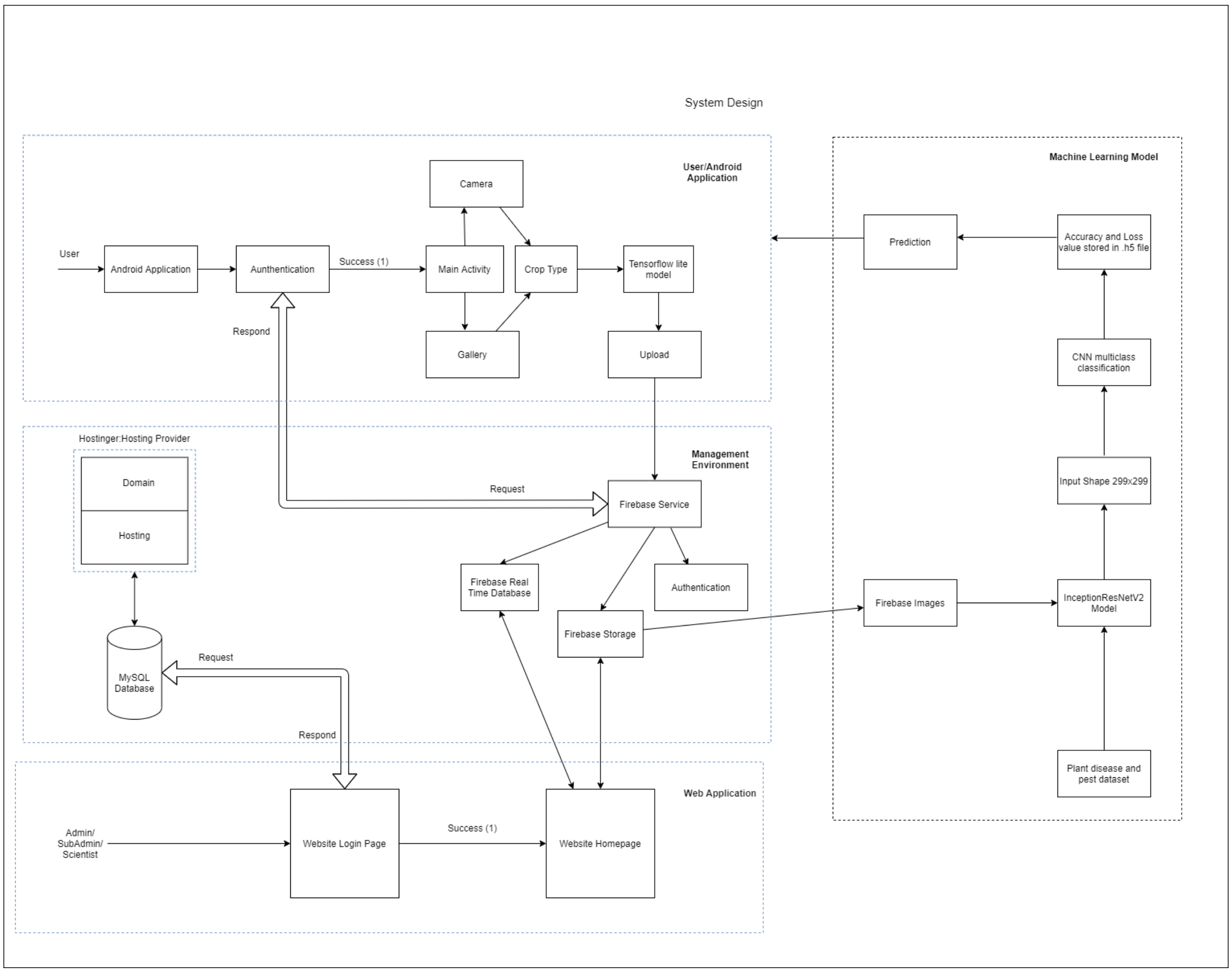


Fig 6.1(a): System Architecture.

The architecture of our framework is shown in fig.1(a) Consists of three major modules user application, management environment and web application. User application is where the farmers will upload images and as well as where they will get the result. Web application is specifically designed for admins of our ecosystem. Management environment shows all the back end of the ecosystem. The Machine learning model shows how the machine leaning model works.

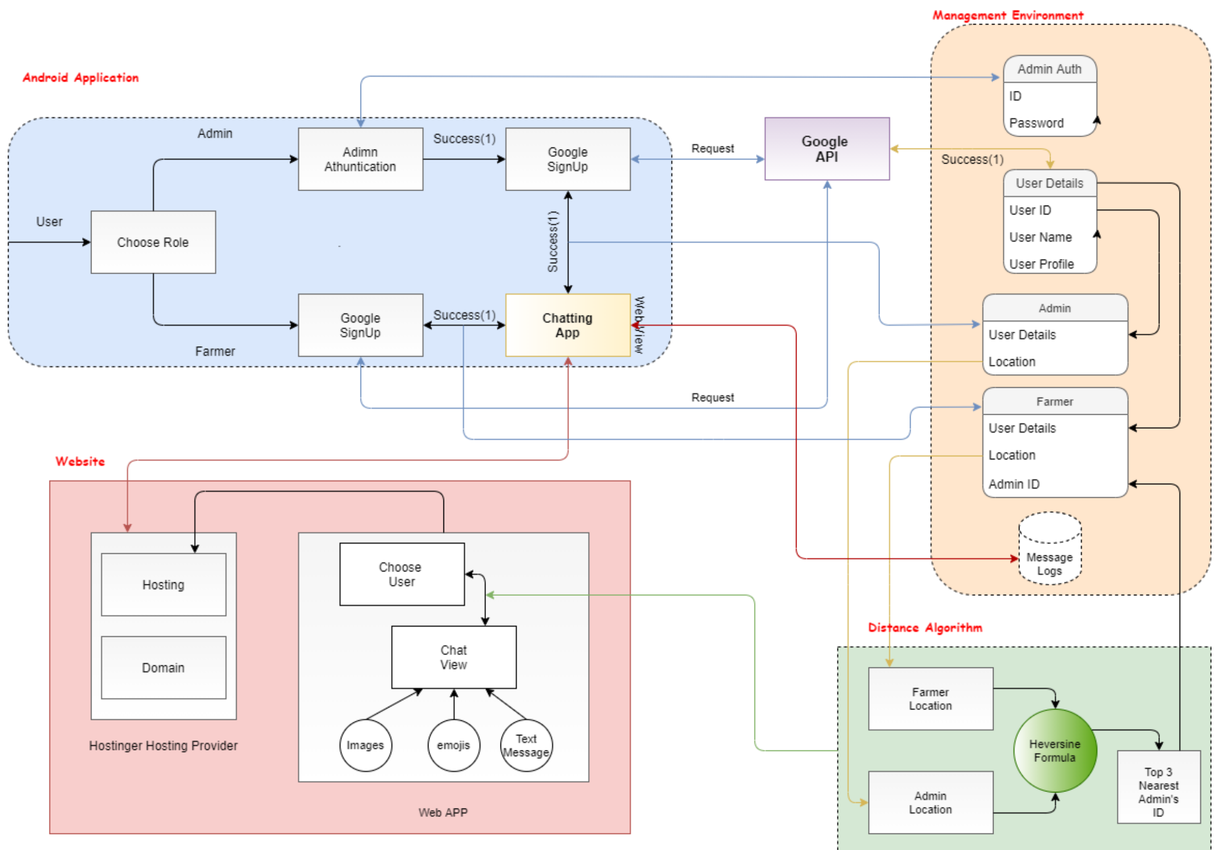


Fig 6.1(b): System Architecture.

Fig 1(b) shows exclusive architecture of the next Android application Kisansathi. This is a chat web application for farmers and scientists merged with android, where farmers can discuss their issues with allocated scientists for a certain geographical area. The scientists are allocated by using a distance algorithm used in our web application. The back end is managed on google services.

6.2 SYSTEM ANALYSIS

6.2.1 Data collection

As the name suggests, this module works on data collection section of our project. We have developed one android app ‘Agrikanti’ for data collection. Farmers and scientist can upload the images from their farms directly with the help of that application. The uploaded data store on firebase cloud storage which is accessible to BAU scientist. We have created website though they can see and manage the image and other metadata. BAU scientist also provided diseases and pests images. We got nearly 3000 images of both rice diseases and rice pests.

6.2.2 Data Preprocessing

It is next step after loading images for pest detection. We will perform some manipulation operation on image dataset. CNN is straight forward for data that we have provided. If we provide rotated or shifted image to neural network it will confusing task for CNN to predict images. Data augmentation is the process of generating replica of each image in the dataset by performing some manipulations.

We will provide some manipulation operations for augmentation such as rescaling, zooming, shearing, horizontal flip. We rescale the image by dividing pixels by 255 because TensorFlow converts PIL image to with a range of [0-255]. We increased image size by factor of 0.2 and provide transformation in factor of 0.2. For test data we only provide rescaling by dividing 255.

Fig 6.2.2: Represents original image and its augmentation.

6.2.3 Machine learning engine

We used InceptionResNetV2 transfer learning model for training purpose. Transfer learning is method in which we use pre-trained weights are used, that have been trained on huge dataset nearly about 1000 classes. When we use pre-trained weights, it is useful for our dataset to achieve higher accuracy. In transfer learning process, we transfer the knowledge from one person to another person. We used InceptionResNetV2 model which has 164 layers with prediction ability of 1000 classes. The default input size for this model is (299,299). That's why we implemented square fitting of 299x299 on our dataset. We removed last layer of transfer learning model which has 1000 neurons. We specified our new layer with 11 neurons as last layer. For that purpose, we used to include_top=false method.

When we summarized model, we get 1081355 trainable parameters among 55418091 total parameters that are available in InceptionResNetV2. We have used flatten layer to connect input layers of model and our specified output layer.

6.2.3.1 Neural Network

A convolutional neural network, or CNN, is a deep learning neural network designed for processing structured arrays of data such as images. Convolutional neural networks are widely used in computer vision and have become the state of the art for many visual applications such as image classification, and have also found success in natural language processing for text classification.

Convolutional neural networks are very good at picking up on patterns in the input image, such as lines, gradients, circles, or even eyes and faces. It is this property that makes convolutional neural networks so powerful for computer vision. Unlike earlier computer vision algorithms, convolutional neural networks can operate directly on a raw image and do not need any preprocessing.

A convolutional neural network is a feed-forward neural network, often with up to 20 or 30 layers. The power of a convolutional neural network comes from a special kind of layer called the convolutional layer.

Convolutional neural networks contain many convolutional layers stacked on top of each other, each one capable of recognizing more sophisticated shapes. With three or four convolutional layers it is possible to recognize handwritten digits and with 25 layers it is possible to distinguish human faces.

The usage of convolutional layers in a convolutional neural network mirrors the structure of the human visual cortex, where a series of layers process an incoming image and identify progressively more complex features.

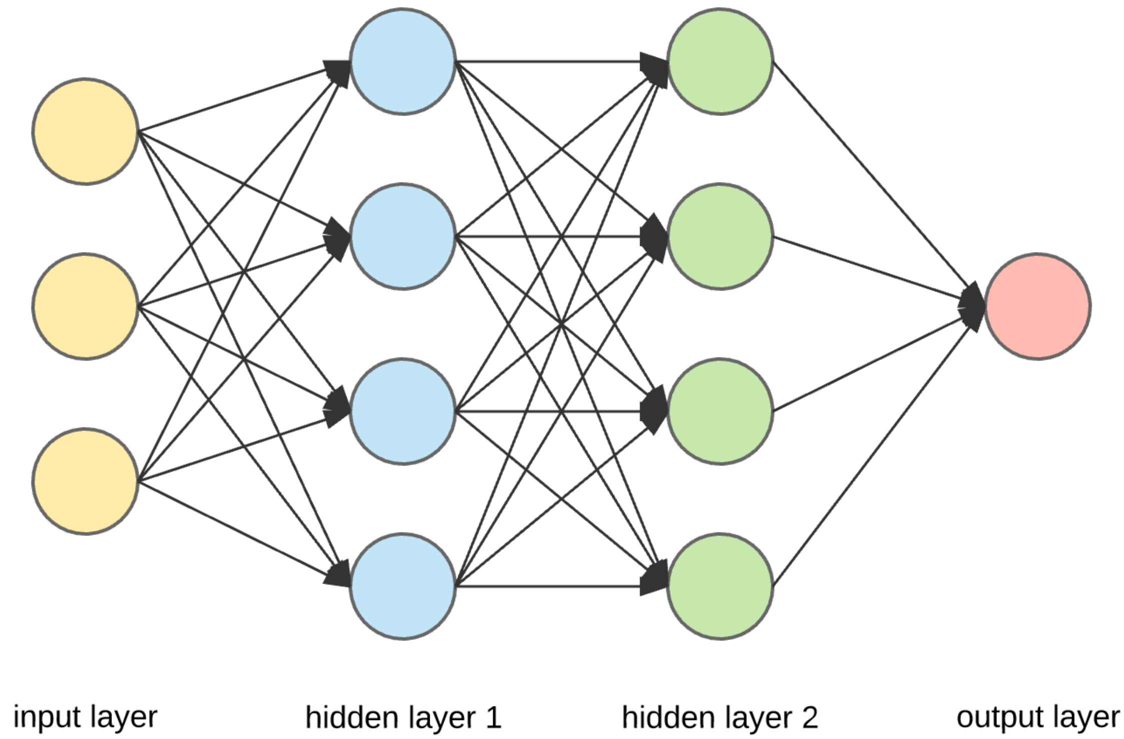


Fig 6.2.3.1 Neural Network

6.3 DETAILED SYSTEM DESIGN

6.3.1 Working of Machine Learning model

Our aim is classification of rice pest images based on their type. We have created CNN model for prediction of various pests on rice leaf. Also a deep learning libraries TensorFlow and keras is used, which provides pre-trained model which has good accuracy and control on overfitting conditions. The CNN model has nearly 164 layers out of which we will remove last layer. Also model I sable to classify 1000 different objects including keyboard, mouse, pencil, and many animals. CNN model trained in this study is shown in fig1. Once CNN model get downloaded, we erased all previous weights. Once images are read in the input size of 299x299 which default input size for given transfer learning model with channel size of 3, since the images are RGB images. After those images are extracted and rescaling is done in order to convert it into 256 pixels. Also, some augmented operations performed such as shearing of image, zooming of image and horizontal flip.

We removed last layer of InceptionResNetV2 CNN model and add two last layers which are flatten layer and dense layer. We can conclude from summary of model that we have total parameters of 54,729,956. Among of them only 393,220 parameters are trainable for our dataset. Dense layer has 11 classes for rice pests ('BPH, Brevennia rehi (rice mealybug), Eublemma, Gandhi Bug, Grass hopper, Horn catpilar, Cnaphalocrocis medinalis, Rice Skipper, Beet Webworm, Stem Borer, Yellow Tail Moth'). It has 'softmax' activation function which is generally used in multiclass classification.

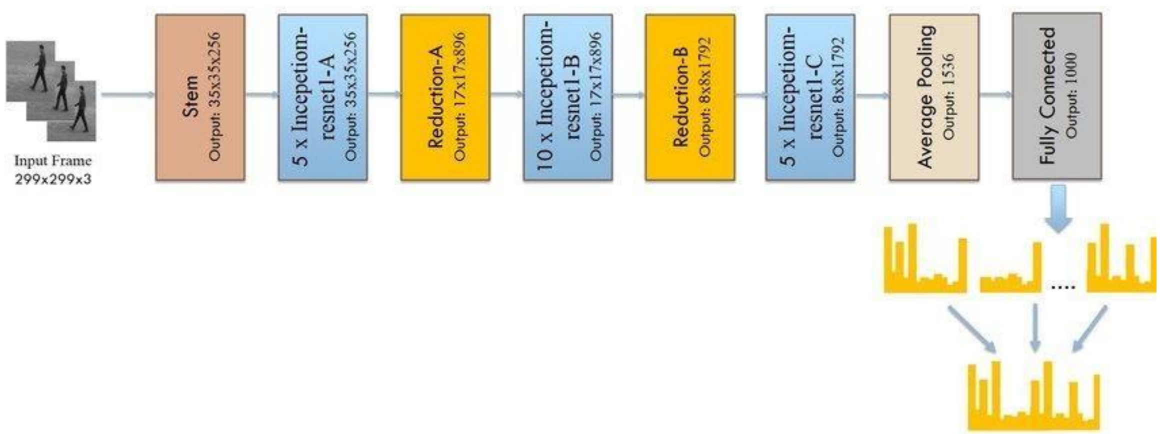


Fig. 6.3.2: Above figure shows general architecture of InceptionResNetV2.

We have used various methods for training and compiling our model such as model checkpoint and reduce learning rate. Before compiling model, we have specified checkpoint factor for minimum validation loss with save best only factor. It means model will save only minimum validation loss among all epochs. While training model, there are many steps occur during each epoch that are forward and backward propagation. This factor is beneficial to save best value even if u get less value in upcoming epochs.

We have also used reduce learning rate factor which monitors validation loss with minimum value. We have provided patience factor as 3 and rate of 0.2. This factor monitors training values for each 3 epochs if there is no improvement in 3 epochs it will reduce learning rate of model by 0.2.

6.4 Data Flow Diagram

6.4.1 DFD Level 0

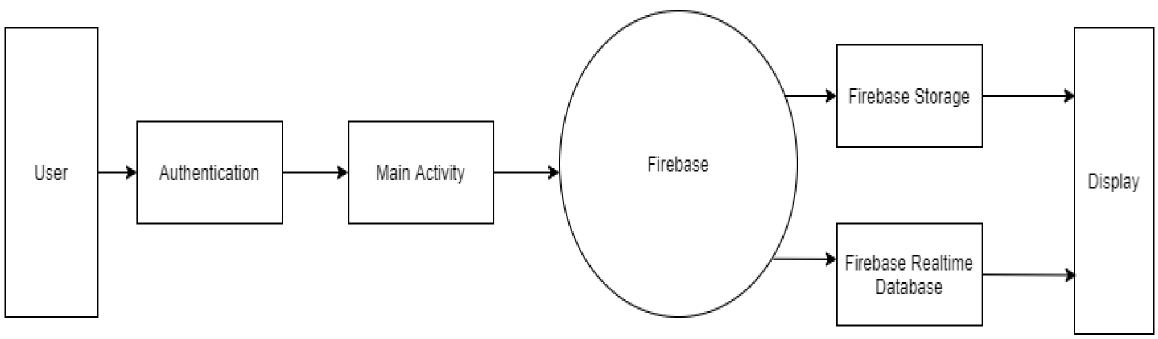


Fig 6.4.1. Level 0 DFD

In DFD level 0 This Project deals with the Agrikanti application and Agrikanti website. To use the Agrikanti web application, user has to complete their registration using their mobile number. After registration they can go straight to the main activity which is uploading images. They have to select crop type first, the touch on camera icon to capture image or they can select image from the gallery and upload it. The uploaded image goes into google firebase service for storage. The whole image gets stored into firebase storage while the tags like location, crop type, date, image URL, pest/disease. This data will be accessible on website for admins where they will be able to sort out the data.

6.4.2 DFD Level 1

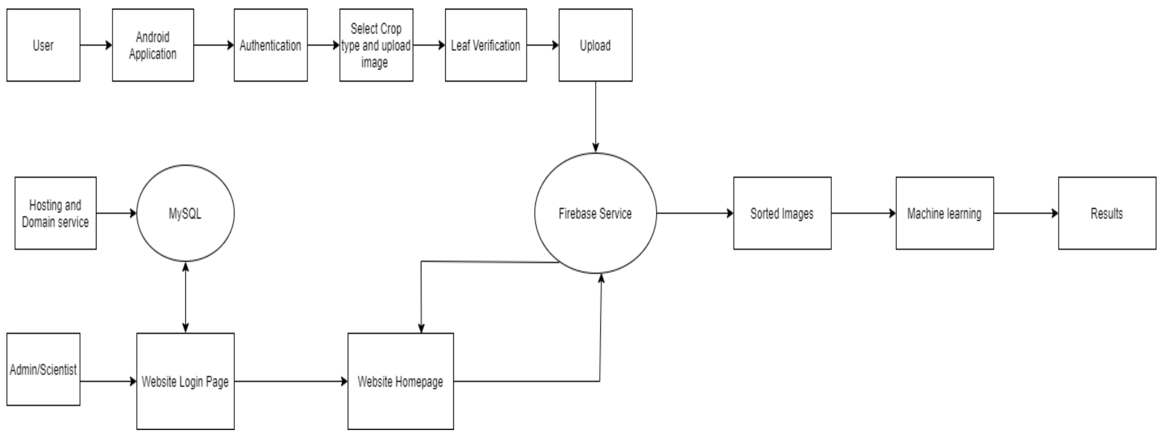


Fig 6.4.2 Level 1 DFD

Level 1 DFD shows more detailed structure of the application and the website. While uploading images, the application checks weather the image uploaded is of crop or not. The admins view the images on website and sort it by selecting the pest or disease which suits it the most. Once the image is opened and viewed on website, the realtime tag associated with the image changes from “Unseen” to “Seen”. This is implemented because the website is used by many users and the other users to have an idea weather the image is viewed and and sorted or not.

6.5 UML Diagrams

6.5.1 Use Case Diagram

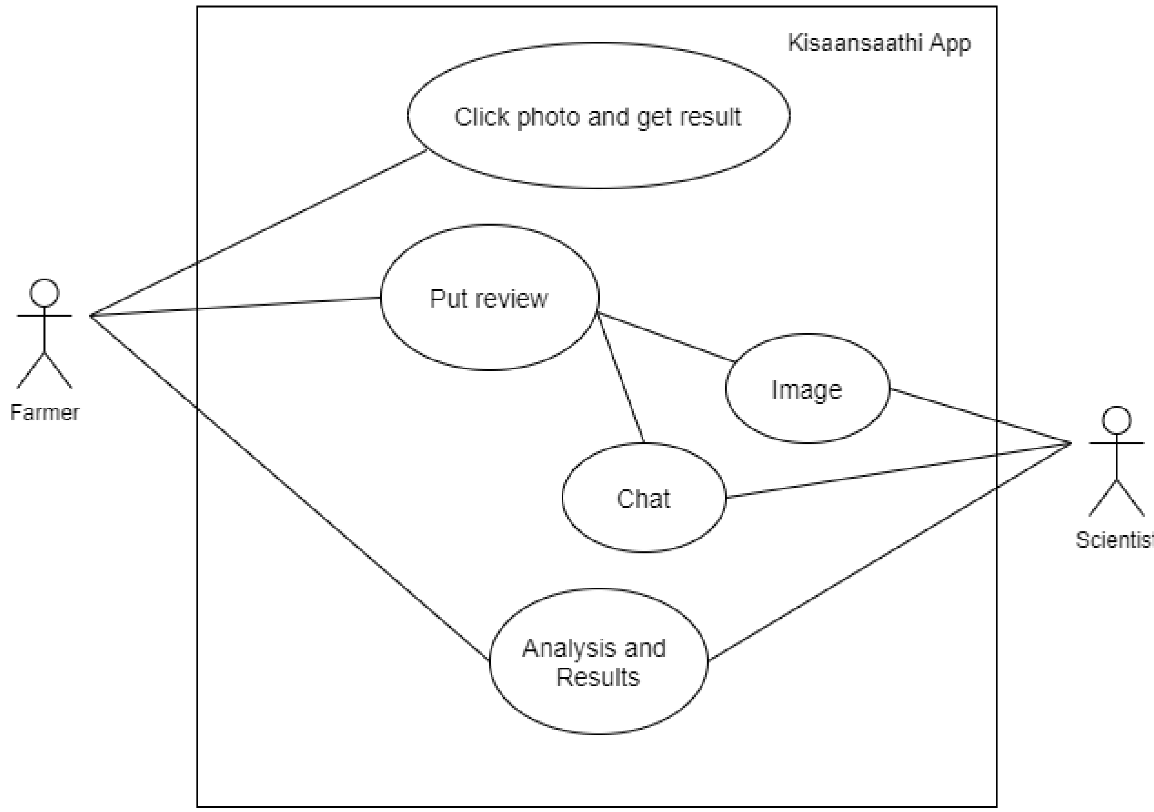


Fig 6.5.1. Use case diagram for Kisansathi

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different Types of users of a system and the various ways that they interact with the system.

This diagram represents “Kisansathi” Android application. In this diagram there is two actors that is Farmer and Scientist. Famer can take photo of crop to check the pest or disease from which it is suffering. Farmer can get result of the same, but if no data is available on the pest or disease, then it can contact with closest scientist nearby and chat with him for guidance. Based on the images/ chat the scientist can help the farmer accordingly.

6.5.2 Class Diagram

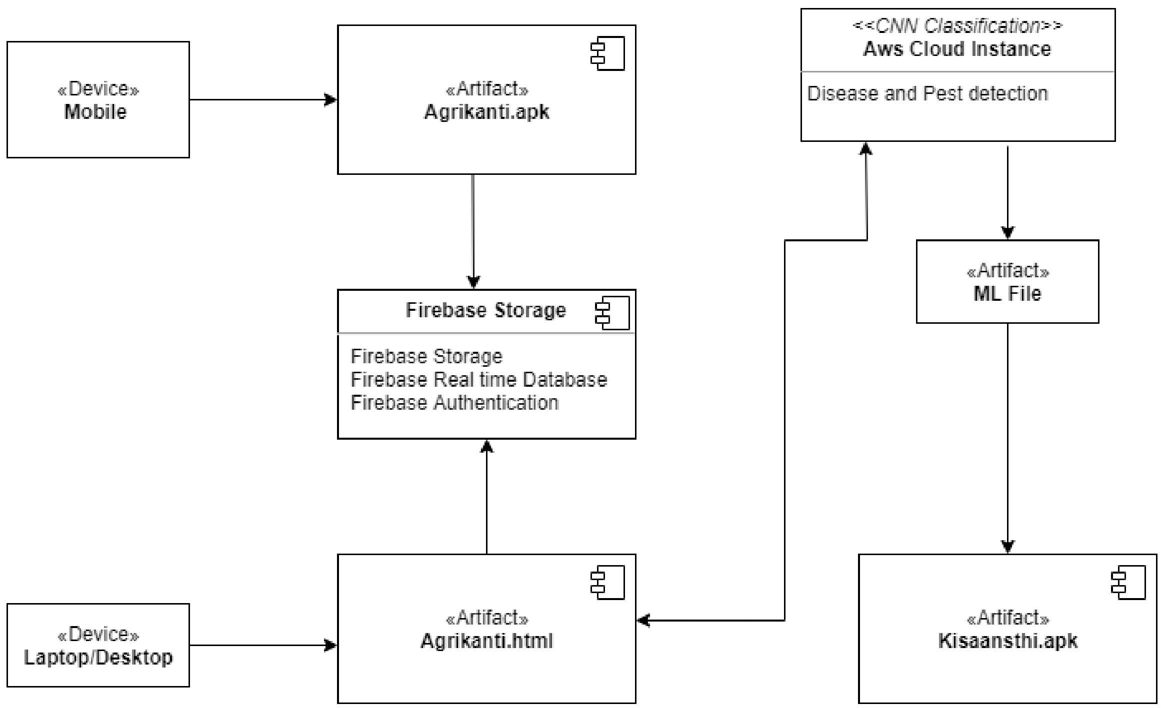


Fig 6.5.2 Class Diagram

The system contains five classes Agrikanti web app, Agrikanti website, AWS cloud instance and Kisansathi android app. Data collection happens through Agrikanti web application and website. All the data is handled through Firebase Storage. The ML part is done on AWS cloud instance which is then incorporated into Kisansathi Android application.

6.5.3 Sequence Diagram

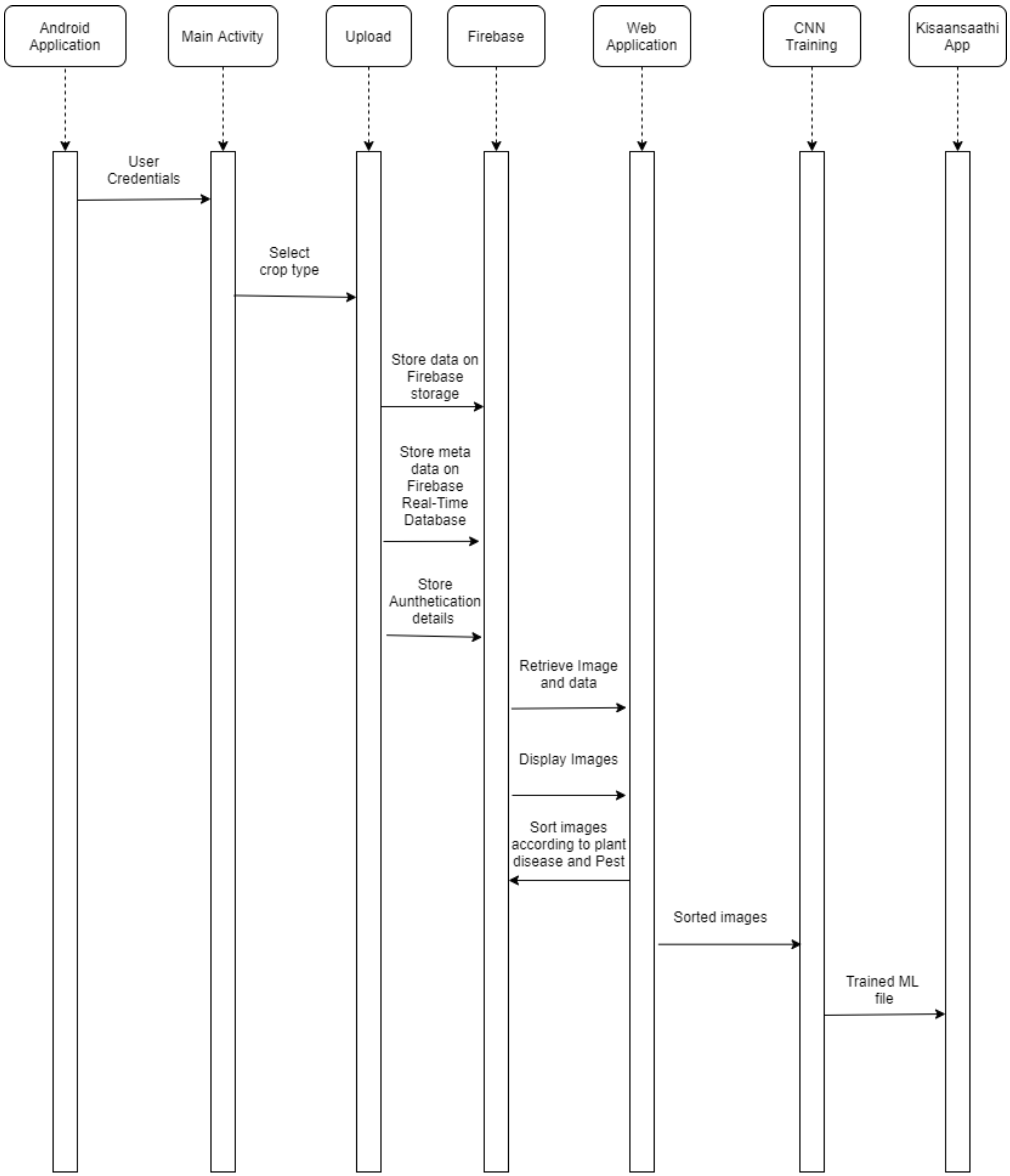
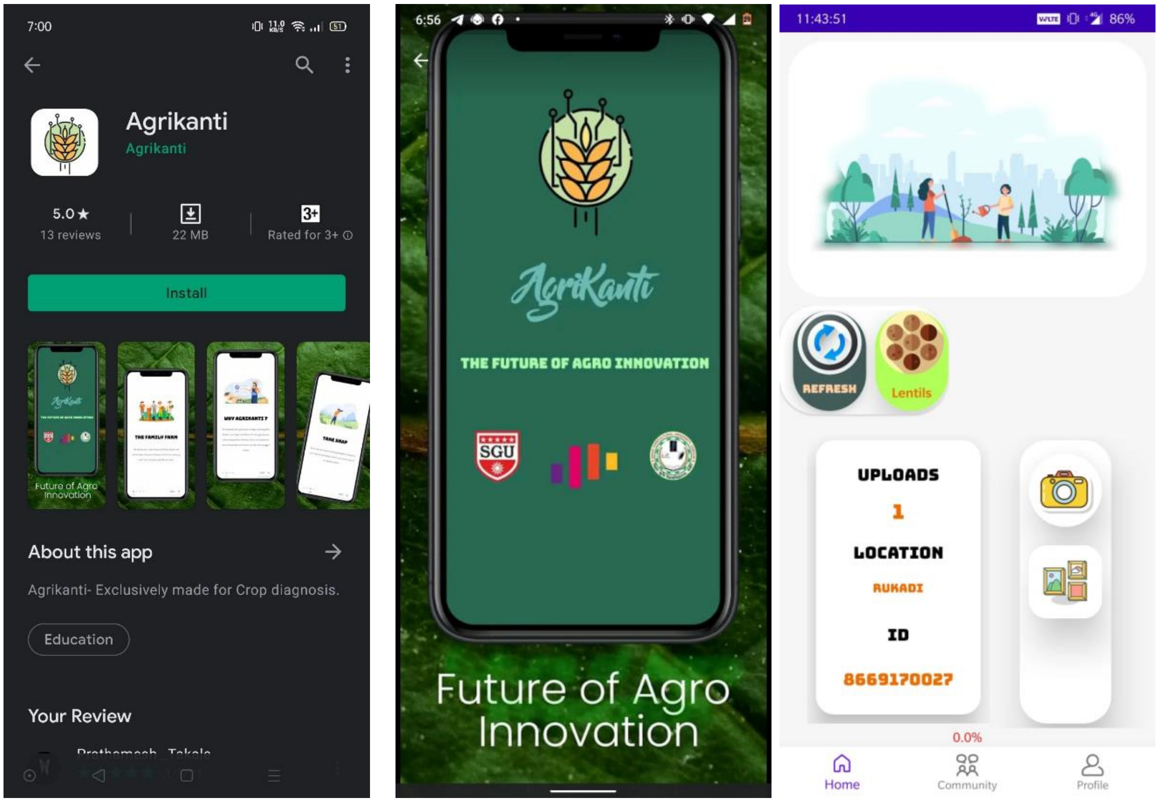


Fig 6.5.3 Sequence Diagram for flow of system

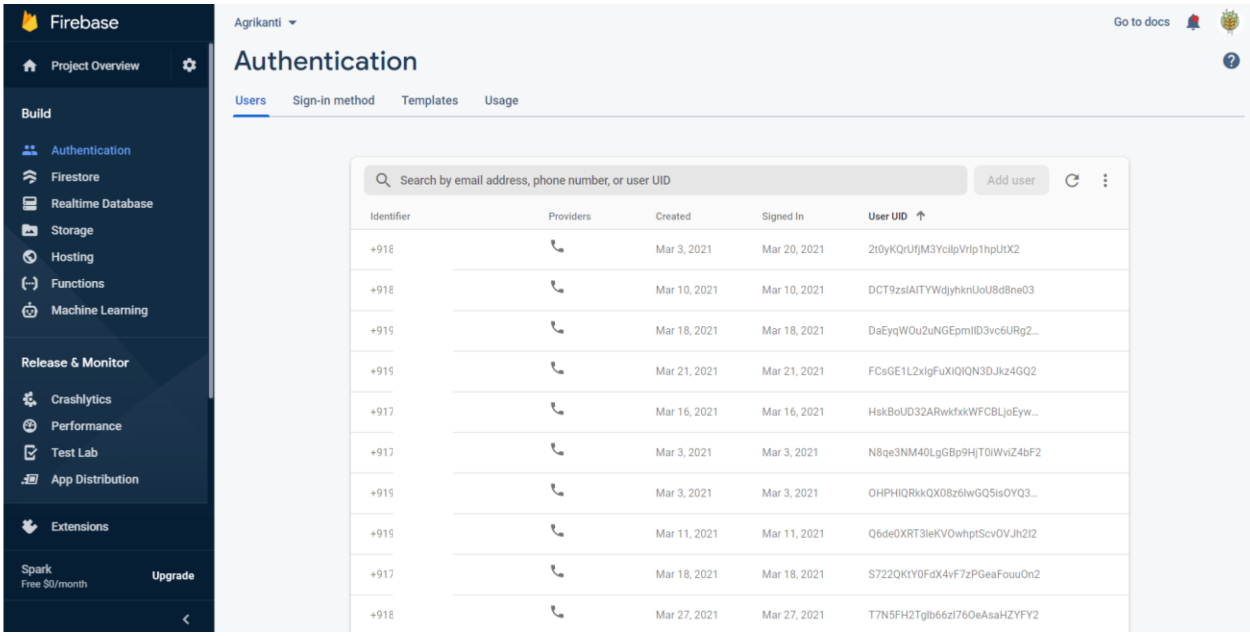
A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence. As our sequence flows from farmer who uploads images from application for data collection, which is sorted by admins on website. The clean images are sent to ML training where, a TensorFlow lite file will be obtained. This file will be implemented on application where users and scientist can check for diseases or pests and can have chat for suggestions and help.

6.6 IMPLEMENTATION

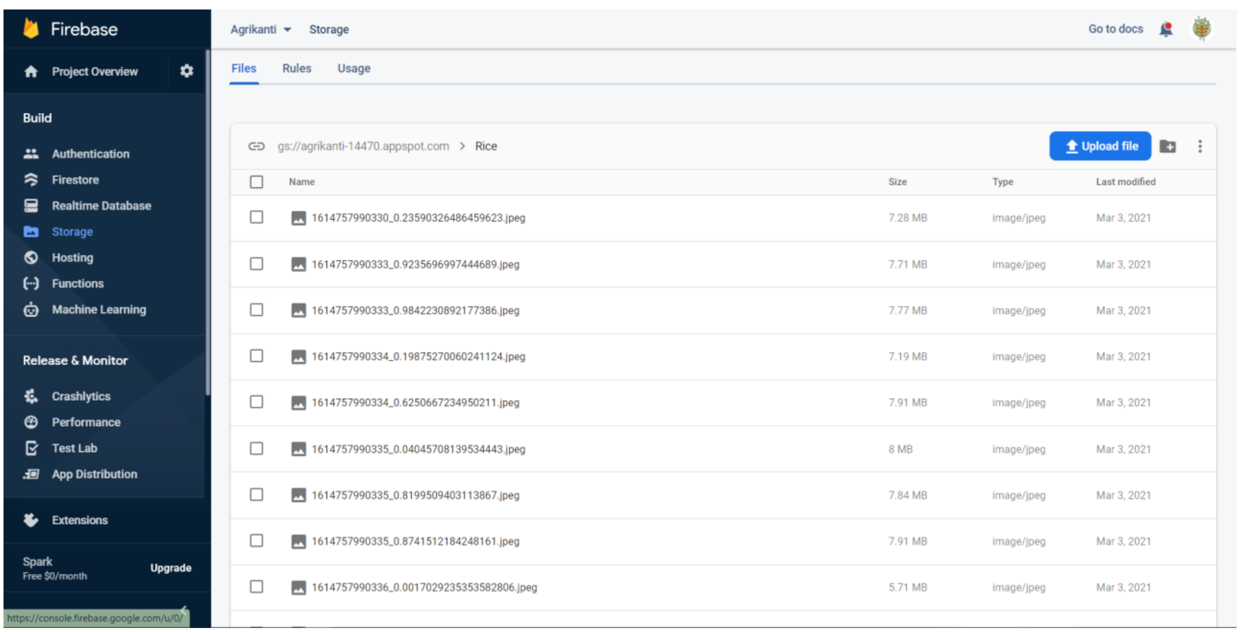


Snapshot a. Agrikanti Application

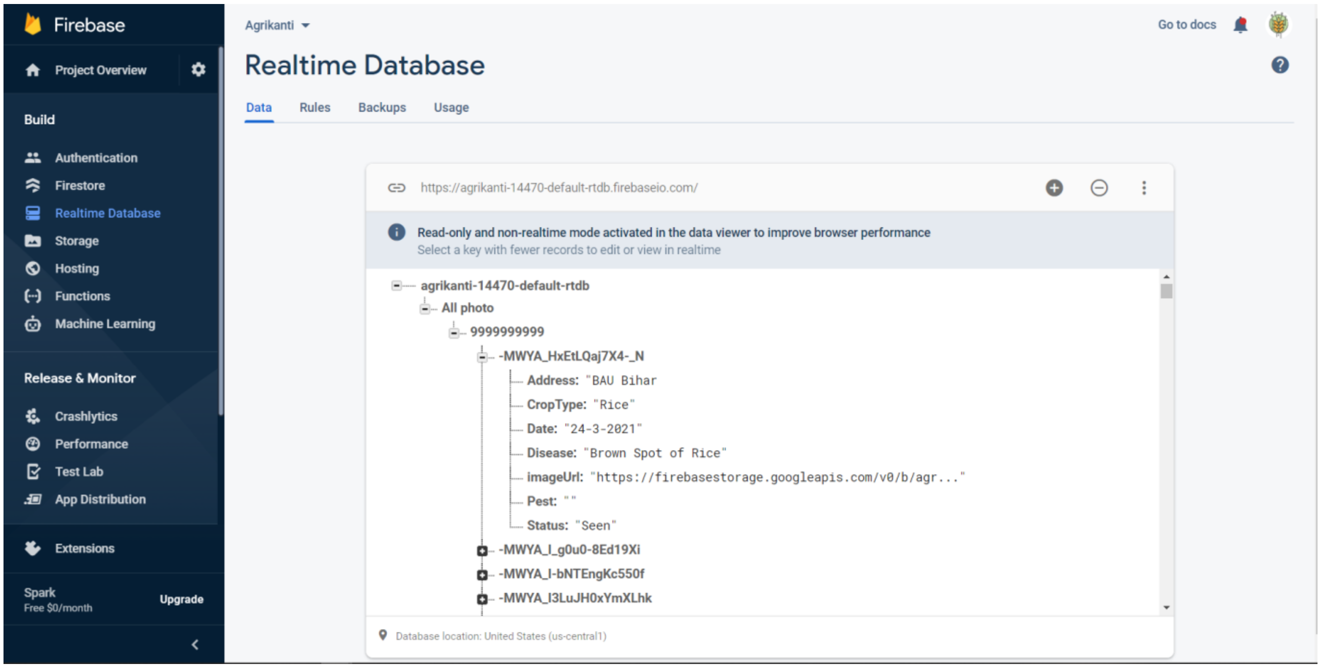
Above snapshot represents Agrikanti application. First image represents app store listing of application. Second shows the startup page and the third image represents the main activity page of the application.



Snapshot b. Application users list on firebase

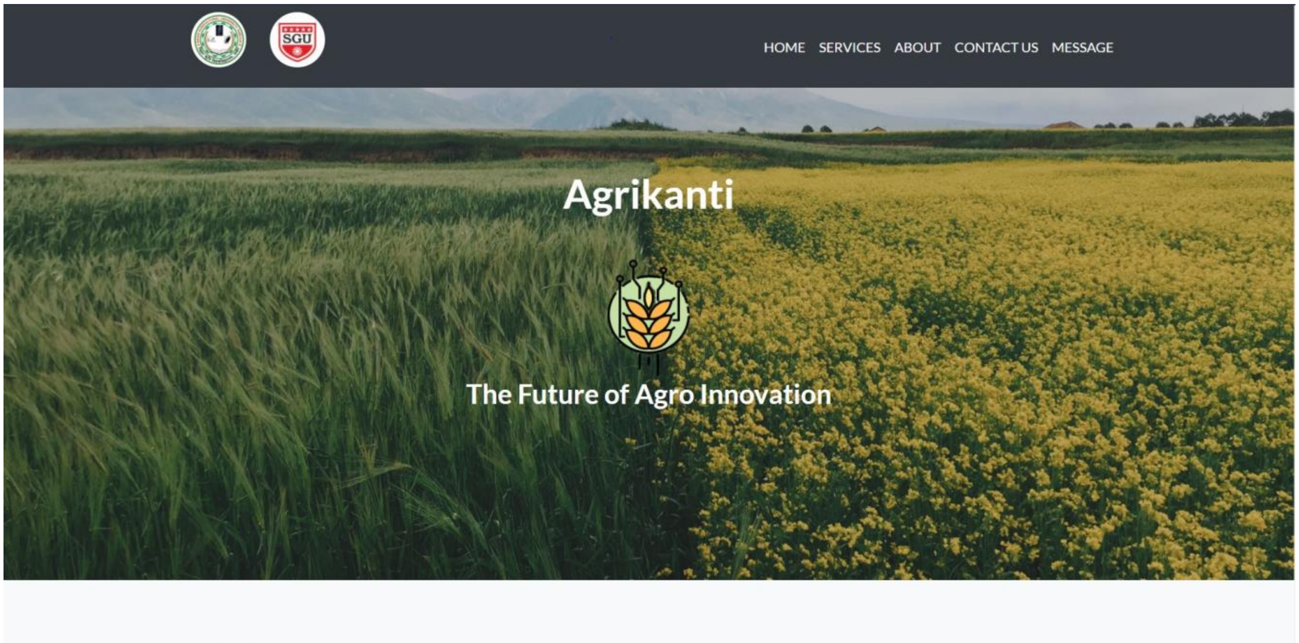


Snapshot c. Images stored in firebase storage

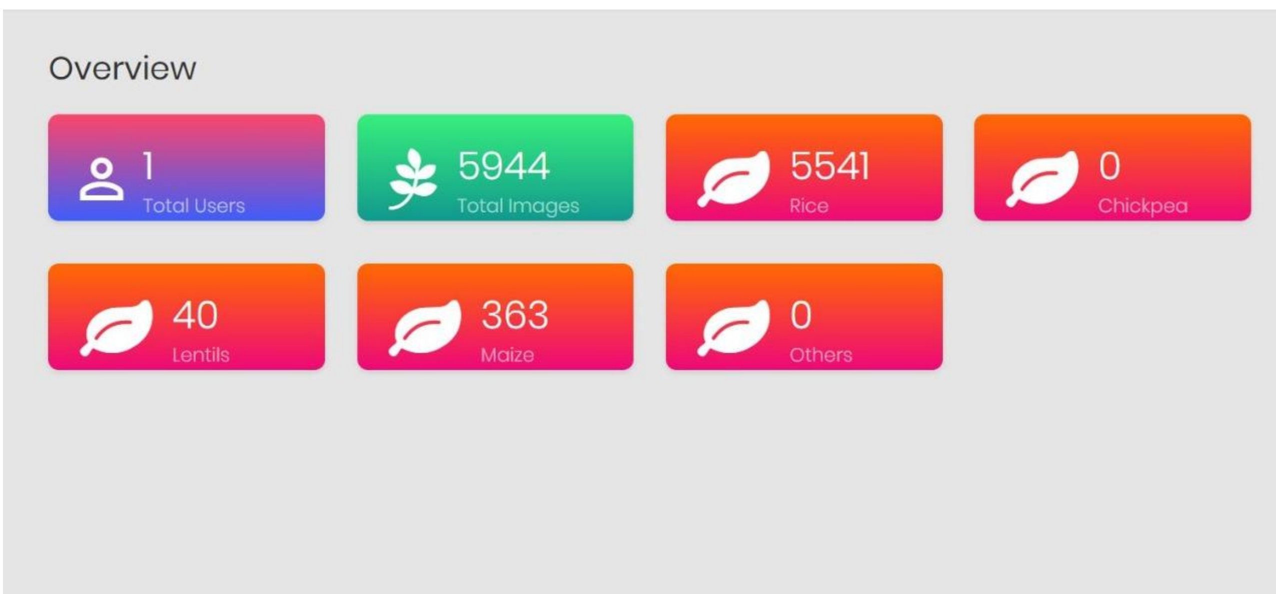


Snapshot d. Image tags on Realtime database

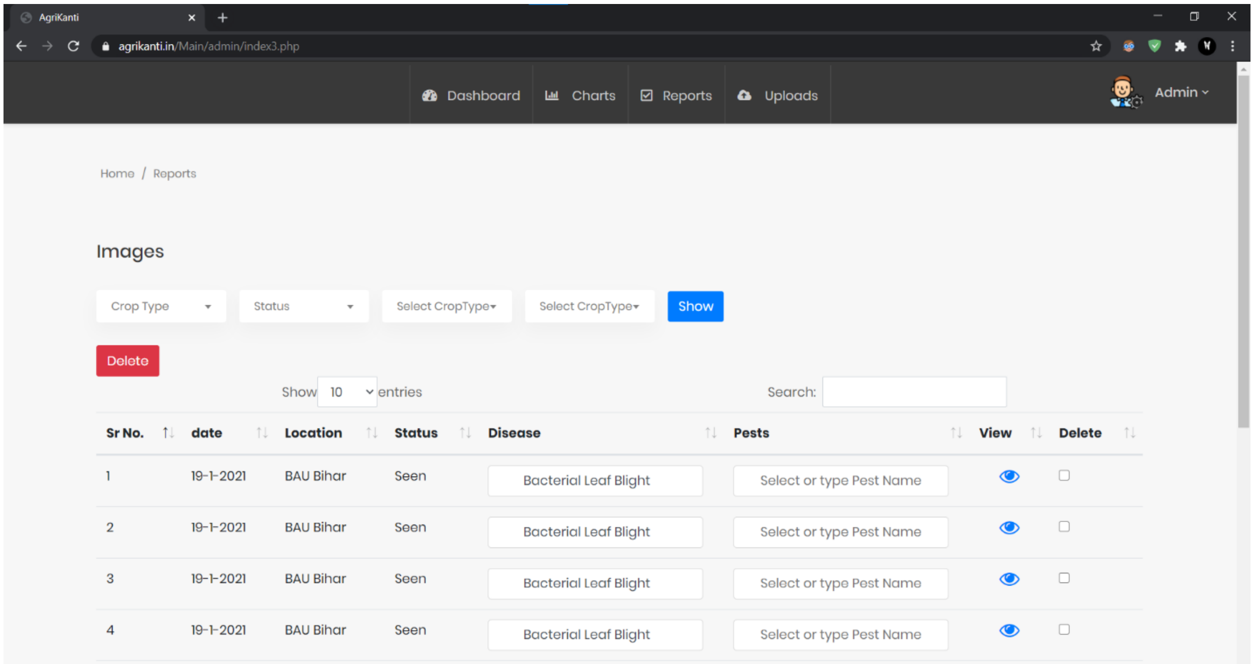
Snapshots b represents the users who registered for the application. While snapshot c represensts all the images uploaded to the Firebase storage. Snapshot d represents meta data tags associated with one on the umage.



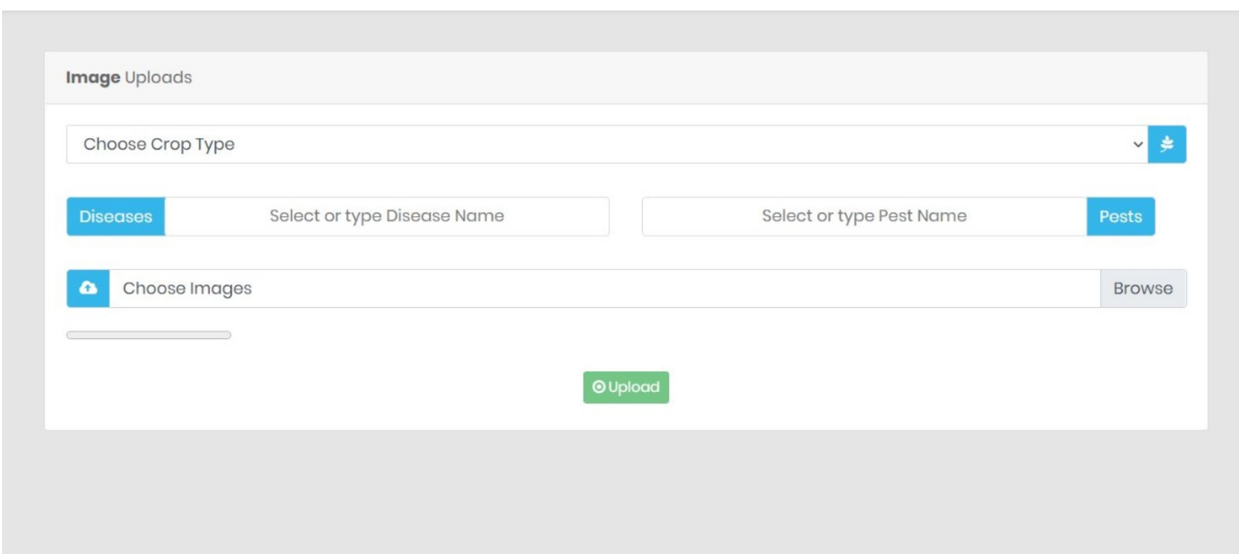
Snapshot e. Agrikanti website portfolio



Snapshot f. Website dashboard for stats



Snapshot g. Image viewing/sorting section



Snapshot h. Image upload through website

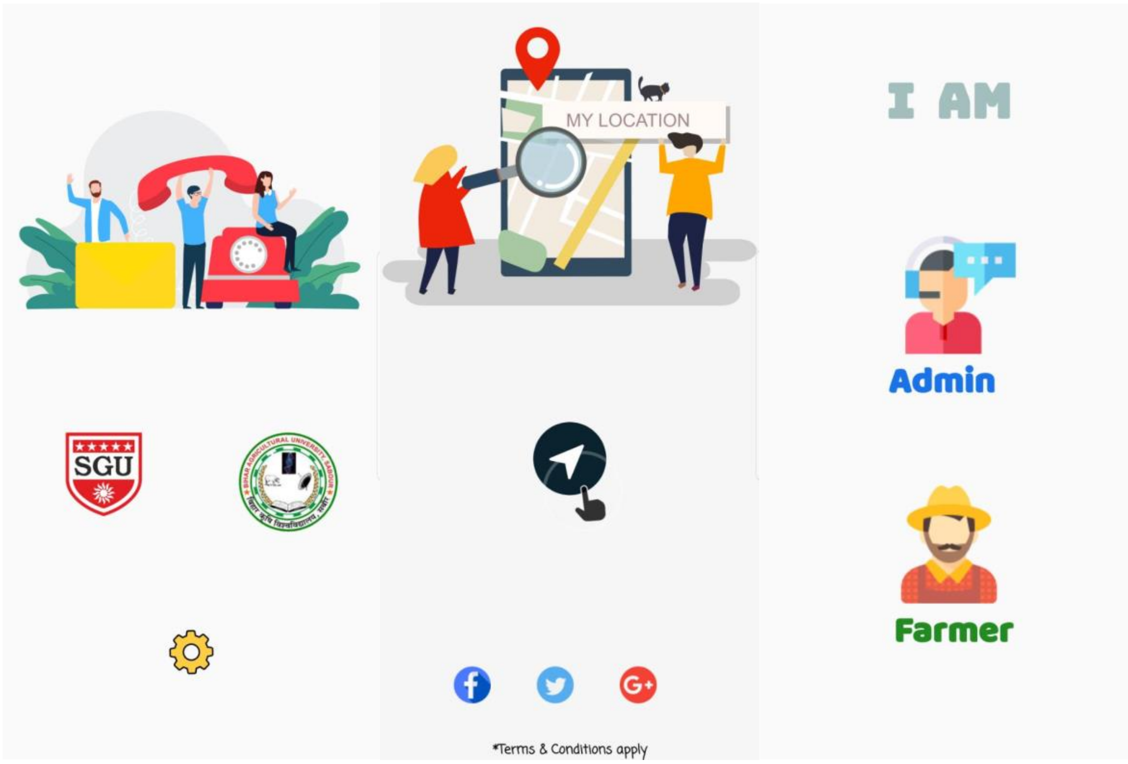
Snapshots e to h show walkthrough of website. Snapshot e shows portfolio page, f shows the dashboard where all stats are displayed. Snapshot g shows image viewing page where all images are listed and h shows image upload section.

```
Train for 42 steps, validate for 11 steps
Epoch 1/100
41/42 [============================>.] - ETA: 3s - loss: 5.1696 - accuracy: 0.7320
Epoch 00001: val_loss improved from inf to 1.48401, saving model to Rice Pests.h5
42/42 [==============================] - 196s 5s/step - loss: 5.0796 - accuracy: 0.7346 - val_loss: 1.4840 -
8974
Epoch 2/100
41/42 [============================>.] - ETA: 3s - loss: 1.0323 - accuracy: 0.9178
Epoch 00002: val_loss improved from 1.48401 to 1.09546, saving model to Rice Pests.h5
42/42 [==============================] - 164s 4s/step - loss: 1.0246 - accuracy: 0.9175 - val_loss: 1.0955 -
9032
Epoch 3/100
41/42 [============================>.] - ETA: 3s - loss: 0.4197 - accuracy: 0.9470
Epoch 00003: val_loss did not improve from 1.09546
42/42 [==============================] - 161s 4s/step - loss: 0.4150 - accuracy: 0.9475 - val_loss: 1.4868 -
9326
Epoch 4/100
41/42 [============================>.] - ETA: 3s - loss: 0.4111 - accuracy: 0.9585
Epoch 00004: val_loss did not improve from 1.09546
```

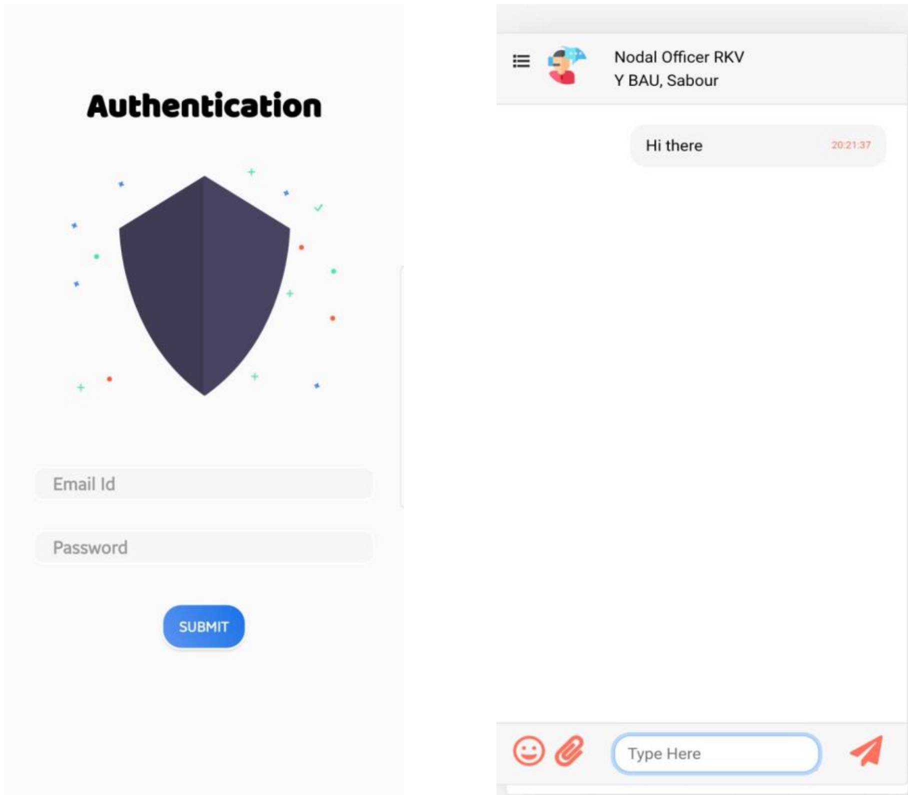
```
9707
Epoch 98/100
41/42 [============================>.] - ETA: 3s - loss: 0.0318 - accuracy: 0.9962
Epoch 00098: val_loss did not improve from 0.78489

Epoch 00098: ReduceLROnPlateau reducing learning rate to 2.147484093416499e-25.
42/42 [==============================] - 161s 4s/step - loss: 0.0310 - accuracy: 0.9963 - val_loss: 0.7863 -
9707
Epoch 99/100
41/42 [============================>.] - ETA: 3s - loss: 0.0357 - accuracy: 0.9977
Epoch 00099: val_loss did not improve from 0.78489
42/42 [==============================] - 160s 4s/step - loss: 0.0348 - accuracy: 0.9978 - val_loss: 0.7863 -
9707
Epoch 100/100
41/42 [============================>.] - ETA: 3s - loss: 0.0153 - accuracy: 0.9954
Epoch 00100: val_loss did not improve from 0.78489
42/42 [==============================] - 160s 4s/step - loss: 0.0149 - accuracy: 0.9955 - val_loss: 0.7863 -
9707
```

Snapshot i. CNN training

Above snapshots show CNN training process.



Snapshot j. Kisaansaathi Application



Snapshot l. Authentication and Chat page (Kisaansaathi)

Snapshots j and l represent Kisansathi application. j introduces to application and l shows the login and chat interface.

CHAPTER 7
TESTING

7.1 Testing Introduction

Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of ending software bugs. Software testing can also be stated as the process of validating and verifying that a software program or application or product:

1. Meets the business and technical requirements that guided its design and development;
2. Works as expected; and
3. Can be implemented with the same characteristics Software testing, depending on the testing method employed, can be implemented at any time in the development process.

However, most of the test effort occurs after the requirements have been defined and the coding process has been completed. As such, the methodology of the test is governed by the software development methodology adopted. Different software development models will focus the test effort at different points in the development process. Newer development models, such as Agile, often employ test driven development and place an increased portion of the testing in the hands of the developer, before it reaches a formal team of testers. In a more traditional model, most of the test execution occurs after the requirements have been defined and the coding process has been completed.

4.1 Testing strategies

7.2.1 Black Box Testing

Black box testing methods focus on the functional requirements in the software. That is, Black box testing enables us to derive sets of input conditions that will fully exercise. All Functional requirements of the program Black box testing attempts to find errors in the Following categories:

- Incorrect or missing function
- Interface errors
- Errors in data structure or external job access
- Performance errors
- Initialization and termination errors.

In the proposed application with the help of this technique, we do not use the code to determine a test suite; rather, knowing the problem that we're trying to solve, we come up with four types of test data:

1. Easy-to-compute data
2. Typical data
3. Boundary / extreme data
4. Bogus data

7.2.2 White Box Testing

White box testing is a set case design method that uses the control structure of the procedural design to derive test cases. Using White box testing methods, we can derive test cases that:

- Guarantee that all independent paths within a module have been exercised
- at least once.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.

- Exercise internal data structures to ensure their validity.

In the proposed application the white box testing is done by the developer implemented the code, the implemented code is studied by the coder, determines all legal (valid and invalid) AND illegal inputs and verifies the outputs against the expected outcomes, which is also determined by studying the implementation code.

7.3 Test plan

Test Plan Identifier: Flip Invariant It is used to identify test plan uniquely.

7.3.1. Purpose of the Test Plan Document

- The main purpose of this document is to fit a particular project as needs. It documents and tracks the necessary information required to effectively define the approach to be used in the testing of the project as product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager, project team, and testing team.

7.3.2. Objective of Test Panning

- To find as many defects as possible and get them _x.

7.3.3Items to be Tested OR Not to be Tested

- Describe the items/features/functions to be tested that are within the scope of this test plan. Include a description of how they will be tested, when, by whom, and to what quality standards. Also include a description of those items agreed not to be tested.

- **Items to be tested**
  1. Overall functionality of the application and website
  2. User Interface of the application and website

- **Not to be Testes**
    1. Performance of the application and website

7.3.4. Test Approach

(a) Describe the overall testing approach to be used to test the project as product. Provide an outline of any planned tests. There are many approaches like:

i. Black Box Testing

ii.White Box Testing.

Here we used Black Box Testing approach. In Black Box Testing we just give input to the system and check its output without checking how system processes it.

7.3.5. Test Pass OR Test Fail Criteria

- When actual and expected results are same then test will be passed. When actual andexpected results are different then test will be failed.

7.3.6. Test Entry OR Exit Criteria

Describe the entry and exit criteria used to start testing and determinewhen to stop testing.

- Entry criteria: As soon as we have requirement we can start testing.
- Exit criteria: When bug rate fall below certain level we can stop testing.

7.3.7. Test Suspension OR Resumption Criteria

(a) Describe the suspension criteria that may be used to suspend all or portions of testing.

Also describe the resumption criteria that may be used to resume testing.

2. Suspension criteria: if there is large change in application like change in requirements, we can suspend work for some time.
3. Resumption criteria: after resolving the respective problem we can resume work.

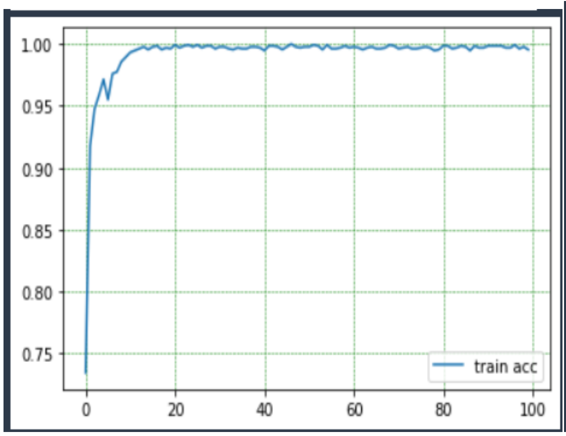
7.4 Test Cases

Test ID	Objective	Pre requisite	Steps	Input Data	Expected Output	Actual Output	Status
1.	Get location on Agrikanti	Location Access	1. open application 2. give location access		Location toast message	Location toast message	PASS
2.	Upload Crop Image	Camera Access	1.Select crop type 2. Click on camera icon 3. Click and select image or upload from gallery	Imag e	Upload successful toast message	Upload successful toast message	PASS
3.	Upload Images from website	Admin access to website	1. Go to upload images tab 2. Select crop type and browse images 3. Choose images	Imag e	Upload Successful Message	Upload Successful Message	PASS

Table 7.4. Test Cases of the System

CHAPTER 8
RESULT AND ANALYSIS

We have trained model for 100 epochs by applying all above factors. After training the model we get the accuracy up to 96% and loss nearly Equal to 0.78. We have generated two files .h5 file and .tflite file which are used in deployment of web application and android application. It contains trained weights which will be used for prediction. From the graph we can specify that there is an exponential increase in accuracy and exponential decrease in loss. For first 15 epochs there is fluctuation in graph. After 20 epoch there is no any sudden change has recorded.



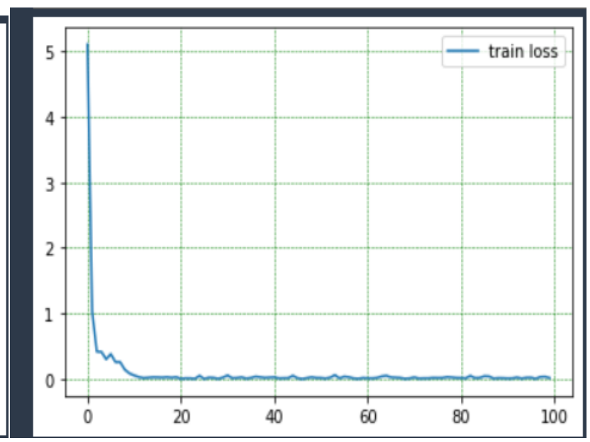
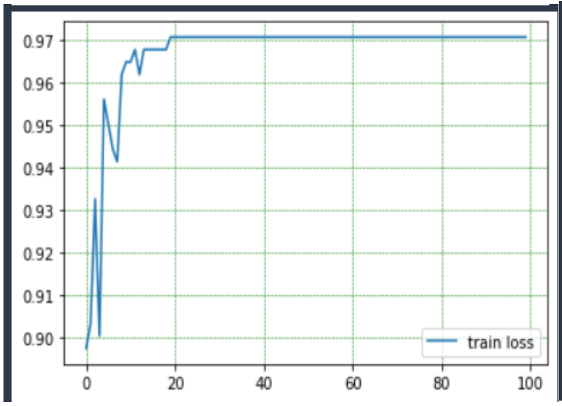


Fig. 8 (a) train accuracy vs number of epochs

Fig. 8 (b) train loss vs number of epochs



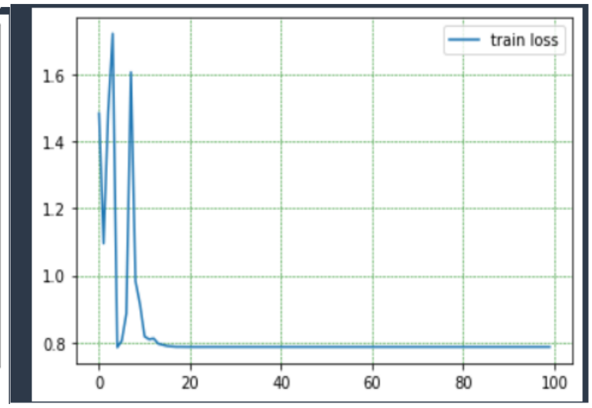


Fig. 8 (c) validation accuracy vs number of epochs

Fig. 8 (d) validation loss vs number of epochs

There are many optimizers available for deep learning but it depends on deep learning model which optimizer is good. Since, we have Adamax, Adam, AdamW and Stochastic Gradient Descent (SGD). Adam is combination of Stochastic Gradient Descent and RMSProp with momentum. Adam evaluates learning rate of every parameter separately. Adamax is next version of Adam optimizer and it calculates gradients from the first two moments. We have used Adam optimizer in our compiling process.

CHAPTER 9

CONCLUSION

Agriculture is the main resource of food and money for farmers. Now-a-days disease occurring on plants has increased which has reduced the productivity of the plants from 10-95 percent. Manually identifying the disease is the difficult and time-consuming task. Instead of manually identifying disease image processing and machine learning technique can be used to identify the disease which is less time consuming, cost effective and more accurate the manual identification technique.

Instead, Machine Learning techniques can be implemented using Convolutional Neural Network (CNN) applied to analyzing visual imagery. In addition to that pre-trained models can also be used for detection of plant/crop diseases and pests. Fine-tuning the models will improve their accuracy and will be easy for us to handle.

Data collection problem is also solved as the developed application will be used directly by farmers on field for uploading data. Along with this, Agricultural experts will sort the images accordingly. And the images will be stored on cloud for easier accessibility.

Along with this, the Kisaansaathi application will help farmers by allocating a select number of scientists for a select number of farmers in a particular area. The farmers will be able to chat with those scientists regarding all their crop problems and will be able to solve it.

This project helps in understanding the creation of a System that will do image processing and identify the plant diseases using machine learning approach.

References:

1. Manisha Bhange, H.A.Hingoliwala (2015) “Smart Farming: Pomegranate Disease Detection Using Image Processing”. Procedia Computer Science 58 (2015) p.280 – 288.

2. Aravind Krishnaswamy Rangarajan, Raja Purushothaman, Aniirudh Ramesh (2018) “Tomato crop disease classification using pre-trained deep learning algorithm”. Procedia Computer Science 133 (2018) p.1040–1047.
3. Kamlesh Golhani, Siva K. Balasundram, Ganesan Vadamalai, Biswajeet Pradhan (2018) “A review of neural networks in plant disease detection using hyperspectral data”. INFORMATION PROCESSING IN AGRICULTURE 5 (2018) p.354–371

4. Karen Lucero Roldán-Serratoa, J.A.S. Escalante-Estradab, M.T.Rodríguez-Gonzálezb (2018) “Automatic pest detection on bean and potato crops by applying neural classifiers”. Engineering in agriculture, environment and food (2018) p.245-255.

5. M.P. Bange, S.A. Deutscher, D. Larsen, D. Linsley, S. Whiteside (2004) “A handheld decision support system to facilitate improved insect pest management in Australian cotton systems”. Computers and Electronics in Agriculture 43 (2004) 131–147

6. Lin Jiao, Shifeng Dong, Shengyu Zhang, Chengjun Xie, Hongqiang Wang (2020) “AF-RCNN: An anchor-free convolutional neural network for multi-categories agricultural pest detection”. Computers and Electronics in Agriculture 174 (2020) 105522.

7. Preetha Rajan, Radhakrishnan B (2016) “A Survey on Different Image Processing Techniques for Pest Identification and Plant Disease Detection”. IJCSN International Journal of Computer Science and Network, Volume 5, Issue 1, February 2016 p.137-141.

8. Yu Sun, Xuanxin Liu, Mingshuai Yuan, Lili Ren, Jianxin Wang, Zhibo Chen (2018) “Automatic in-trap pest detection using deep learning for pheromone-based Dendroctonus valens monitoring”. Biosystems Engineering 176 (2018) 140-150.

9. Yun Hwan Kim, Seong Joon Yoo, Yeong Hyeon Gu, Jin Hee Lim, Dongil Han, Sung WookBaik (2014) “Crop Pests Prediction Method using Regression and Machine Learning Technology: Survey”. IERI Procedia 6 (2014) 52 – 56.

10. Yanfen Li, Hanxiang Wang, L. Minh Dang, Abolghasem Sadeghi-Niaraki, Hyeonjoon Moon (2020) “Crop pest recognition in natural scenes using convolutional neural networks”. Computers and Electronics in Agriculture 169 (2020) 105174.

11. A.M. Rajeshwari (2020) “Fuzzy Decision Support System for Recommendation Crop Cultivation Based on Soil Type”. Researchgate Conference Paper.

12. Isabel M. del Aguila, Joaquın Canadas, Samuel Tunez (2015) “Decision making models embedded into a webbased tool for assessing pest infestation risk”. Biosystems Engineering 133 (2015) 102-115.

13. Yong-Wei Bao, Ming-Xuan Yu, Wei Wu (2011) Design and Implementation of Database for a webGIS-based Rice Diseases and Pests System. Procedia Environmental Sciences 10 (2011) 535 – 540.

14. Vijai Singh, A.K. Misra (2017) “Detection of plant leaf diseases using image segmentation and soft computing techniques”. Information Processing in Agriculture 4 (2017) 41-49.

15. Yan Li, Chunlei Xia, Jangmyung Lee (2015) “Detection of small-sized insect pest in greenhouses based on multifractal analysis”. Optik - Int. J. Light Electron Opt. (2015).

16. Everton Castel˜ao Tetila, Bruno Brandoli Machado, Gilberto Astolfi, Nícolas Alessandro de Souza Belete, Willian Paraguassu Amorim, Antonia Railda Roel, Hemerson Pistori (2020) “Detection and classification of soybean pests using deep learning with UAV images”. Computers and Electronics in Agriculture 179 (2020) 105836.

17. Wayne Goodridge, Margaret Bernard, René Jordan, Reanne Rampersad (2017) “Intelligent diagnosis of diseases in plants using a hybrid Multi-Criteria decision making technique”. Computers and Electronics in Agriculture 133 (2017) 80-87.

18. Fangyuan Wang, Rujing Wanga, Chengjun Xiea, Po Yangc, Liu Liu (2020) “Fusing multi-scale context-aware information representation for automatic in-field pest detection and recognition”. Computers and Electronics in Agriculture 169 (2020) 105222.

19. Chowdhury R. Rahman, Preetom S. Arko, Mohammed E. Ali, Mohammad A. Iqbal Khan, Sajid H. Apon, Farzana Nowrin, Abu Wasif (2020) “Identification and recognition of rice diseases and pests using convolutional neural networks”. Biosystems Engineering 194 (2020) 112-120.

20. Saurabh Sindhu, Divya Sindhu (2019) “Image Processing Technology Application for Early Detection and Classification of Plant Diseases”. Article in INTERNATIONAL JOURNAL OF COMPUTER SCIENCES AND ENGINEERING · May 2019.

21. Jeremy P.M. Whish, Neville I. Herrmann, Neil A. White, Andrew D. Moore, Darren J. Kriticos (2014) “Integrating pest population models with biophysical crop models to better represent the farming system”. Article in press, Environmental Modelling and Software (2014) 1-8.

22. A.K. Tripathy, J. Adinarayana, K. Vijayalakshmi, S.N. Merchant, U.B. Desai, S. Ninomiya, M. Hirafuji, T. Kiura (2014) “Knowledge discovery and Leaf Spot dynamics of groundnut crop through wireless sensor network and data mining techniques”. Computers and Electronics in Agriculture 107 (2014) 104-114.

23. Jingcheng Zhang, Yanbo Huang, Ruiliang Pu, Pablo Gonzalez-Moreno, Lin Yuan, Kaihua Wu, Wenjiang Huang (2019) “Monitoring plant diseases and pests through remote sensing technology: A review”. Computers and Electronics in Agriculture 165 (2019) 104943.

24. Peng Chen, Qingxin Xiao, Jun Zhang, Chengjun Xie, Bing Wang (2020) “Occurrence prediction of cotton pests and diseases by bidirectional long short-term memory networks with climate and atmosphere circulation”. Computers and Electronics in Agriculture 176 (2020) 105612.

25. Trond Rafoss, Knut Sælid, Arild Sletten, Lars Fredrik Gyland, Liv Engravslia (2010) “Open geospatial technology standards and their potential in plant pest risk

management—GPS-enabled mobile phones utilising open geospatial technology standards Web Feature Service Transactions support the fighting of fire blight in Norway". Computers and Electronics in Agriculture 74 (2010) 336-340.

26. Oluwafemi Tairu (2018) "Plant AI—Plant Disease Detection using Convolutional Neural Network".
DOI 26/07/2020.

27. Burhanudin Syamsuri, Gede Putra Kusuma (2019) "Plant Disease Classification using Lite Pretrained Deep Convolutional Neural Network on Android Mobile Device". Researchgate publication 339326576 DOI December 2019.

28. B Nithya Ramesh, Vishwas BM, Banu Shankar (2018) "Plant Leaf Disease Detection Using Advanced Image Processing and Neural network". International Journal of Recent Trends in Engineering & Research (IJRTER) Volume 04, Issue 04; April- 2018 [ISSN: 2455-1457].

29. Lawrence C. Ngugi, Moataz Abelwahab, Mohammed Abo-Zahhad () "Recent advances in image processing techniques for automated leaf pest and disease recognition – A review".

30. Alvin R. Malicdem, Proceso L. Fernandez (2020) "Rice Blast Disease Forecasting for Northern Philippines". Article in press.

31. M.G. Hill, P.G. Connolly, P. Reutemann, D. Fletcher (2014) "The use of data mining to assist crop protection decisions on kiwifruit in New Zealand". Computers and Electronics in Agriculture 108 (2014) 250–257.

32. Cheng-Long Chuang, En-Cheng Yang, Chwan-Lu Tseng, Chia-Pang Chen, Gi-Shih Lien, Joe-Air Jiang (2014) "Toward anticipating pest responses to fruit farms: Revealing factors influencing the population dynamics of the Oriental Fruit Fly via automatic field monitoring". Computers and Electronics in Agriculture 109 (2014) 148–161.