"E-KETHA": Enriching Rice Farmer's Quality of Life through a Mobile Application

S.M.W. Jayasundara
Department of Software Engineering
Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

H.H.W.M.B.S. Paranagama
Department of Software Engineering
Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

G.W.D.A. Gunarathne
Department of Information Technolgy
Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

P.Y.D. Jayasinghe
Department of Software Engineering
Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

K.M.U. Ranthilina
Department of Software Engineering
Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

L.A.A.U. Gunasinghe
Department of Information Technolgy
Sri Lanka Institute of Information Technology, Malabe, Sri Lanka

ABSTRACT
When it comes to Asian countries, rice is the most common type of food that is consumed daily. Due to that rice farmers face a huge amount of stress to supply according to the massive demand. This is happening while they are farming in poor conditions such as, amongst diseases and pests that harm rice crops with the inclusion of weeds that plague the field. They also have difficulties finding the correct fertilizers and the amount that is needed for the crops to grow properly. Another issue discovered, was that some rice plants are underdeveloped, and farmers lack the understanding when it comes to proper treatment. These topics were chosen according to a multitude of statistics including losses due to all insects, losses due to all diseases, losses due to all weeds, potential production harvested, and total potential production lost before harvest, being found respectively at 34.4%, 9.9%, 10.8%, 44.9%, and 55.1%. The main objective of this research is to provide solutions to the above-mentioned issues faced by the farmers. Four CNN models were compared in previously mentioned four areas to provide solutions. The used models were 2 customized CNNs that were best fit for pest and fertilizer management, a customized AlexNet for growth management, and a customized ResNet model for weed management. To map the weed, u-net, FCN architecture was used and calibrated, with it providing 94.88% training accuracy and 64.39% validation accuracy. Hight measuring and area calculation was implemented and then finetuned using a custom-made python operation. The aim is to develop a mobile application that will help farmers solve these problems using the chosen algorithms. The application will use image processing to analyze crops to find solutions stored in a cloud database. Then machine learning and deep learning will be used to recommend appropriate solutions.

General Terms
Machine learning, image processing, deep learning, CNN

Keywords
Farming, rice crops, agriculture,harvest

1. INTRODUCTION
Rice farming, which has been cultivated by humans since 3000-2500 BC has been one of the main staple sources of sustenance. This is due to multiple beneficial qualities that it provides such as being rich in carbohydrates, fiber, selenium, and even vitamin B. Since rice mainly grows in hot and humid climates, Asia is the current mass producer of rice in the world. In particular, Chinese, Indian, and Sri Lankan people tend to eat rice on a daily basis. Considering Asian countries, due to rice being high in demand, local farmers struggle massively to meet the said demand. This can be to the point of even having to import rice from overseas as shown in the graph below.

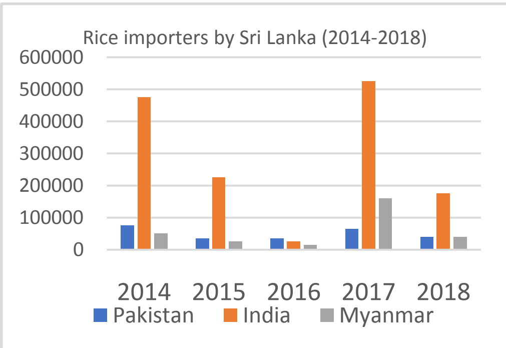

Fig 1: Rice importers by Sri Lanka (2014-2018), by main importing countries

There is a multitude of reasons on why rice farming can be slowed down when it comes to local and foreign farming. However in particular they affect farmers in developing countries more due to a lack of proper technology and knowledge. The four most important reasons as to why it is slowed down are,

- Pests and disease – The many pests and diseases that could harm rice crops.
- Weeds – The weeds that absorb nutrients from the soil.
- Fertilizer misuse – The improper use of fertilizers that harms rice crops.

- Growth problems – Issues when it comes to the growth of rice crops.

2. LITERATURE REVIEW

2.1 Crop farmers mobile application

Help the farmer with summary information about crops, fruits, and vegetables. Climatic and Soil Requirements, Avocado, Banana, Beans, Carrot, Jackfruit, Cucumber, Garlic, Irish Potato, Lettuce, Sorghum, Watermelon, Onion, Bell Peppers and Peppers, Pineapple and Eggplant sour info explaining how to grow them. The app also describes the most common causes of pests and diseases, symptoms, how they spread, with prevention and control measures. Where possible, the app will advise on suitable farming methods to control crop pests. This app can be used as a guide for new farmers, or anyone involved in farming around the world. Teach new farming techniques/methods to avoid attacking your crops. It also provides information on best practices to follow in order to improve farmers' performance in growing these crops[1].

2.2 Pest identification using image processing using neural network

This study is done by Johnny L. Miranda et al. with the goal of classifying pests in crops. Pest infestation in rice production is a challenging task for crop technicians and farmers. A pest infestation can cause serious losses and also affect the income of farmers. Decisions for pest predictions can be made by estimating the density of farmers. Existing detection techniques for these species involve the use of various traps to detect their presence. In this study, an identification system was developed for the automatic detection of field insect pests. Continuous monitoring by a wireless camera for video recording is done by catching the insect with a sticky trap. Various imaging techniques are used to identify and extract the captured insect. A neural network was used to identify the extracted insect pests. The new automated detection system developed in this study provides reliable detection[2].

2.3 Weed Classification for Site-Specific Weed Management Using Automated Stereo Computer-Vision

MojtabaDadashzadeh et al. completed this study with the intention of classifying weeds in a specific site using a stereo vision system to distinguish rice plants and weeds. This system is further augmented using an artificial neural network and two other metaheuristic algorithms, them being particle swarm optimization (PSO) and the bee algorithm (BA). With stereo videos being recorded of the site beforehand and decomposed into singular frames, rice plants were extracted using color, shape, and even texture. Then the previously mentioned metaheuristic algorithms were used to optimize the neural network and classify the weed detected as well. According to the K-nearest neighbors (KNN) classifier, this reached 88.74% and 87.96% for right and left channels without accounting for arithmetic or the geometric means as the basis and with it 92.02% and 90.7% respectively[3].

2.4 A nutrient recommendation system for soil fertilization based on evolutionary computation

This research has been conducted to predict the fertilizers for different crops and give nutrient recommendations by analyzing the crop fertility and yield production. However, this application is limited to selected fertilizers (Nitrogen (N), Phosphorus (P), and Potassium (K)). This recommendation is done by using an improved genetic algorithm (IGA) which will use time-series sensor data and recommends of various crop settings. By analyzing the way that fertilizer works, the application will be able to instruct farmers to get the maximum yield output [4].

2.5 Rice crop height measurement using a digital image processing

This is a plant height identification method currently in operation at Thailand. It detects the height of the plant and shows the height to the user. But it does not use a mobile application.

Here is an automatic image processing method to identify the user based on the photos taken by a digital camera mounted on a field server, including a marker bar used to describe the height of the rice plant. Height can be assessed by analyzing the uploaded image obtained by the user. Digital image processing for analysis uses four steps to automatically measure rice crop height. Therefore, it is possible to get the height of the rice tree [5].

Table 1. Comparing existing applications and our application features

	2.1	2.2	2.3	2.4	2.5	E-Ketha
Detect diseases	Yes	No	No	No	No	Yes
Detect pests	No	Yes	No	No	No	Yes
Detect weeds	No	No	Yes	No	No	Yes
Provide guidance manage fertilizer	No	No	No	Yes	No	Yes
Detect growth	No	No	No	No	Yes	Yes

3. RESEARCH OBJECTIVES

The main objective of this research project is to help farmers with their paddy fields and make life easier for them. The farmers will be receiving proper guidance and techniques so that producing a steady abundant yield of crops to match the great demand of consumers.The Sub objectives are as follows,

3.1 Detection of pests and diseases using image processing and finding solutions by applying machine learning

Users will have the ability to take a picture of a diseased or a pest-ridden crop to identify the type of disease or pest. After identifying the pest or disease type the application will present the most suitable solutions to treat the crops.

3.2 Detection of weeds using image processing and finding solutions by applying machine learning

Users will have the ability to take a picture of weeds in the paddy field aerially or weed plant itself to identify the hotspots or the type. Then the application will present the most suitable solutions to remove the weeds without having to harm the rice crops.

3.3 Provide fertilization solutions according to the size of the paddy field and the fertilizer type using image processing, then after providing the instructions by applying machine learning

Users can use the application to track how much of a paddy area is required to fertilize and by taking an image of the fertilizer that is being used to fertilize. Then the application will help to identify the best utilization methods with detailed instructions including the amount and dosage of fertilization that could be used to aid their growth.

3.4 Rice crop growth identification using image processing and giving solutions to debilitated crops by applying machine learning

Users will first need to enter the distance to the rice plant. Then the user has to take a picture of the plant. Finally, the application will provide solutions for the deficient crops.

4. METHODOLOGY

The methodology followed in this particular research is mentioned below.

4.1 Data acquisition

Kaggle dataset containing 4081 images split into 4 different classes and they are "Blast","Browspot", "Tungro" and "Bacterialblight" [7] were used for detection of diseases and providing solutions. Kaggle dataset containing 4781 images split into 4 different classes and they are "Caterpillar","Trombidium","Caterpillar",and "white grubs" were used for detection of pests and providing solutions. Deep weeds TensorFlow dataset containing 17,509 images split into 8 different weed species has been used for the weed identification model training with them being "Black River", "Charters Towers", "Cluden", "Douglas", "Hervey Range", "Kelso", "McKinlay" and "Paluma" were used for detection of weeds and providing solutions. The dataset used for the weed mapping component contains UAV (Unmanned Aerial Vehicle) imagery including ground truth images corresponding to the UAV imagery. The labels used for the training are "Weed","Rice", "Sand" and "Other" were used for weed mapping from UAV images. A custom-made dataset has been created for this component which contains 3800 images split into 2 different classes with them being “rice” and “non-rice” used for rice crop growth identification and providing solutions. A custom-made dataset has been created for this component which contains 785 images split into 4 different fertilizer types has been used for the model training and they are "Muriate of Potash (MOP)", "Triple Super Phosphate (TSP) ", "Urea" and "Zinc Sulphate" used for providing fertilization solutions according to the size of the paddy field and the fertilizer type

4.2 Pre-processing

When it comes to pre-processing all the models that are described below went through the same process. Which includes shuffling, resizing, rescaling, flipping horizontally and vertically. Finally, normalization was performed according to the mean and standard deviation calculated for the datasets.

4.3 Models discussion

4.3.1 Detection of diseases and providing solutions

Customized CNN was used as the main model for disease identification. CNN was chosen due to it being one of the most basic deep learning models which can take input images and have them differentiated. TensorFlow is primarily used here, this is due to the highly flexible array of tools, libraries, and community resources contained in it. Then Keras is used to provide abstractions and building blocks for the development of machine learning code. For plotting the outputs maplotlib was used due to its simple but detailed GUI. 80% and 20% split was made for the training and testing set. The data model starts with layers, they are as follows

- 3 input layers with maxPooling2d
- 3 output layers,
- 1 flatten layer
- 2 Dense layer

Tried different Optimizers with the model in order to get the maximum results.

4.3.2 Detection of pests and providing solutions

For pest identification, AlexNetwas used as the model with modifications being done in order to best fit the dataset. The reason why AlexNet was chosen is due to its relatively short training time compared to other deep learning models. This is because it allows multi-GPU usage thus making use of multiple GPUs if they are present to detect pests and provide solutions. Hyperparameter tuning[6] was performed for the parameters of batch size, learning rate, and epochs. Due to there being research showing that higher values for learning rate and batch size do not always provide Higher results, a lower number was chosen initially with it gradually going higher. As for the epochs, a brute force method was used to see which would be best. The data was shuffled, resized, and rescaled in order to perform preprocessing. An image size of 256*256 due to it fitting with the model parameter.

4.3.3 Detection of weeds and providing solutions

For the purpose of Weed identification, the ResNet (Residual Network)[8] model was used with modifications made to improve the model. This model was chosen in order to answer the issue of vanishing or exploding gradient which is a nuisance in deep neural networks that have a large number of layers. What is meant by vanishing or exploding gradient is the gradient becoming zero or becoming a large number with the increase of layers thus providing a high error rate on both training and test datasets. How ResNet archives this is by using the concept of residual blocks which utilize the technique of skip connections. These skip connections connect activation layers to oncoming layers by skipping the layers in the middle of them. How it decides to skip is by seeing if the next layer is damaging the performance. In particular ResNet50 model is used here due to the reasoning of giving the best results as well as 224x224 pixel size images being used as the dataset. The 50 after the model’s name is the number of layers in the model as such ResNet50 contains 50 layers. Torch is used as the main framework, providing the resnet library as well as its lr_scheduler which gives the optimal learning rate. Torch is used as it provides flexibility and speed for deep neural network implementation such as in this case. In order to preprocess, the mean and the standard deviation were calculated for the normalization. Size of the image was changed to 224 as it was the best fit for the model. Some random rotation and flipping were introduced to augment the images. This was done for both test and train portions of the dataset in the 80% and 20% split respectively. As for the valid set, a further 10% was taken from the train set. 32 batch size was chosen according to previous research findings after providing the best level of predictions with shuffling enabled. In the model, there is an initial 7x7 convolutional layer including batch normalization. This batch normalization allows for the re-scaling and re-centering, thus making the training process significantly faster. ReLU has been chosen as the activation function due to it not activating all the neurons at the same time, thus keeping the operational costs at a manageable level. Downsampling max pooling is used in a similar vein. For the basic block that comes after the initial layers, 3x3 convolutional layers have been given, and lastly for the bottleneck block 1x1,3x3 along with 1x1 convolutional layer. As for the training, a learning rate of 0.001 has been chosen after it was found by the lr_scheduler. Adam was chosen as the optimization algorithm for this learning rate finder as it is considered the best when it comes to the dataset being vast. It does so by observing the loss acquired after going through learning rates in a given range. 10 epochs were used as they provided the best accuracy. Top-k accuracy is used to predict the label for the prediction as the classes can look very similar to each other and in this dataset, some classes look very similar to each other. Hyperparameter tuning[6] was performed for the parameters of batch size, learning rate, and epochs. Due to there being research showing that higher values for learning rate and batch size do not always provide Higher results, a lower number was chosen initially with it gradually going higher. As for the epochs, a brute force method was used to see which would be best.The data was shuffled, resized, and rescaled in order to perform preprocessing. An image size of 256*256 due to it fitting with the model parameter.

4.3.4 Weed mapping from uav images

The custom modification of U-net[9], which use a fully convolutional network is used for the semantic segmentation task of mapping the weeds from the paddy field. U-net is specifically chosen due to its high performance when it comes to segmentations that has few labels such as the case in this dataset. TensorflowKeras is used as the main framework due to it providing all the necessary abstractions as well as building blocks needed for the implementation of the u-net model. The model itself is made up of 23 convolutional layers split into contraction, bottleneck, and expansion sections. Contraction includes 3x3 convolutional layers with 2x2 max pooling. Expansion layer has 3x3 CNN layers with 2x2 convolution layer. Finally, the expansion section has multiple blocks that contain 3x3 CNN layers with 2x2 upsampling layers. After some modifications and changes, the default model was proven to give the best output. The conv2D layers in the model use the “ReLu” activation function and “he_normal” kernel initializer. The former is due to the reason mentioned in the previous part and the former is due to it being better coupled together with “ReLu”. Hyperparameter tuning [6] was performed for the parameters of batch size, learning rate, and epochs. Due to there being research showing that higher values for learning rate and batch size do not always provide Higher results, a lower number was chosen initially with it gradually going higher. As for the epochs, a brute force method was used to see which would be best.

4.3.5 Provide fertilization solutions according to the size of the paddy field and the fertilizer type

TensorflowKeras model is used for creating this model. Customized CNN was used as the main model for fertilizer identification. CNN was chosen due to it being one of the most basic deep learning models which can take input images and have them differentiated. The. 80% and 20% split was made for the training and testing set. As for the train, test, and validation datasets, a batch size of 32, a target size of 256*256 due to the resolutions of the preprocessed images and categorical class mode since there are multiple classes. The layers of the model have been modified accordingly in order to get maximum accuracy.

- 4 - Convolution2D layers with ‘relu’ activation function
- 4 - pooling layers
- 4 - MaxPooling2D layers
- 4 - Dropout layers
- 1 - Flatten layer (to get output in the set of numbers)
- 1 -Dense layer with ‘SoftMax’ activation function (to change the output into a probability)

As for the Optimizer ‘Adam’ was used due to the problem being large and containing lots of data and parameters. ‘categorical_crossentropy’ is used as a loss function because the dataset contains more than 2 classes. In order to calculate the area of the paddy field, a Mobile device’s GPS has been used. The application was developed so that a user can easily calculate any paddy field part that they want to fertilize. User has to ping the 4 corner locations of the area that is required for fertilization. Then the application will get the latitude and longitude of each location, and calculate the area of the paddy field. Hyperparameter tuning [6] was performed for the parameters of batch size, learning rate, and epochs. Due to there being research showing that higher values for learning rate and batch size do not always provide Higher results, a lower number was chosen initially with it gradually going higher. As for the epochs, a brute force method was used to see which would be best. The data was shuffled, resized, and rescaled in order to perform preprocessing. An image size of 256*256 due to it fitting with the model parameter.

4.3.6 Rice crop growth identification and providing solutions

This is done in order to identify whether the plant is a rice plant or not. For this a custom AlexNetmodel[10] with Keras TensorFlow is used for the specification of this dataset and its special quality which is speed. Another strength this algorithm possesses is nonlinearity which is provided by Rectified Linear Unit (ReLU). This also adds to its already impressive speed. The layers of the model have been modified accordingly in order to get the maximum accuracy.

- 3 - Convolution2D layers with ‘relu’ activation function
- 3 – Max pooling layers
- 2 – Fully connected hidden layers
- 1 – Fully connected output layers
- 1 – Flatten layer (to get output in set of numbers)
- 2 – Dropout layers
- 3 – Dense layer with ‘SoftMax’ activation function

The images were resized to 227*227 which is the required input size of the AlexNet model. In order to measure the height of the rice plant, a python code has been implemented that has the capability to measure the height when the distance to the plant has been inputted. 80% and 20% split was made for the training and testing set. Hyperparameter tuning [6] was performed for the parameters of batch size, learning rate, and epochs. Due to there being research showing that higher values for learning rate and batch size do not always provide Higher results, a lower number was chosen initially with it gradually going higher. As for the epochs, a brute force method was used to see which would be best. Imutils package is for the resizing, image translation, rotation, resizing, skeletonization, or blur amount detection and scipy.spatial.distance, a package for measuring the distance of the object.

4.4 Result and discussion

4.4.1 Detection of pests and providing solutions

Adam is used as the optimizer since it gave the best outputs and it was fit for datasets such as this.

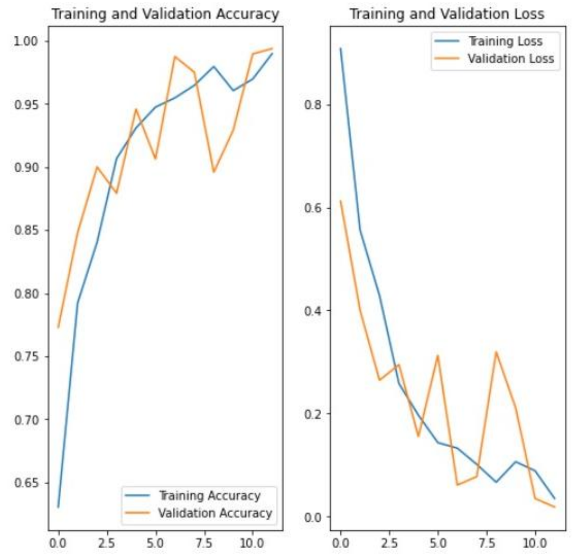

Fig 2: Test accuracy

The above figure represents training and validation accuracy with the loss. The y-axis depicts the accuracy for the first graph and the loss for the second graph. The x-axis depicts the number of epochs. As it is clear from the graphs with a large number of epochs, loss and accuracy fluctuation gets higher. It is apparent that a large number of epochs does not negatively affect accuracy or loss.98.98% training accuracy and 97.71% test accuracy were able to be reached using this model as shown in the examples and predictions. 20 epochs and 32 batch size are the hyperparameters used in this model.

Table 2. Model accuracy

Epoch	loss	accuracy	Val_loss	Val_accuracy
08/12	0.1013	0.9647	0.0775	0.9750
09/12	0.0666	0.9766	0.3196	0.8958
10/12	0.1064	0.9605	0.2108	0.9292
11/12	0.0888	0.9696	0.0351	0.9896
12/12	0.0351	0.9898	0.0186	0.9937

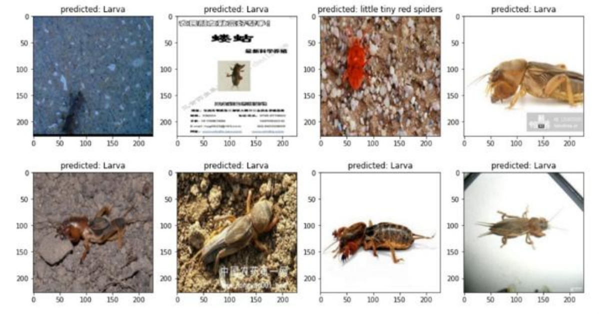

Fig 3: Pests prediction output

4.4.2 Detection of diseases and providing solutions

Table 3. Model accuracy

Epoch	Loss	accuracy	Val_loss	Val_accuracy
11/15	0.3942	0.8649	0.3157	0.8678
12/15	0.3645	0.8610	0.2362	0.9040
13/15	0.4016	0.8421	0.2662	0.8963
14/15	0.3234	0.8775	0.2476	0.9071
15/15	0.2875	0.8837	0.1775	0.9288

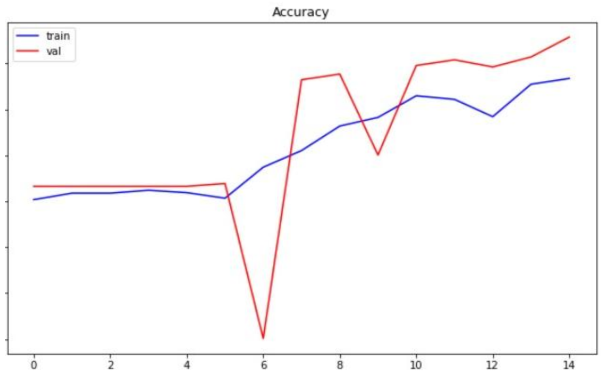

Fig 4: Training accuracy chart

The above figure represents training and validation accuracy. The y-axis depicts the accuracy and the x-axis depicts the number of epochs. Even though validation accuracy drops near the middle it quickly recovers and runs parallelly to the training accuracy. It is evident that the most favorable number of epochs is close to 15.88.37% training accuracy and 92.88% test accuracy were able to be reached using this model as shown in the examples above with the predictions for the test data shown below. 15 epochs, 0.1 learning rate, and 32 batch size are the hyperparameters used in this model.Adam is used as the optimizer since it gave the best outputs and it was fit for datasets such as this.

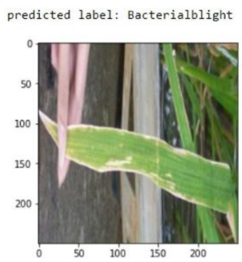

Fig 5: Diseases prediction output

4.4.3 Identification of weeds and providing solutions

Table 4. Resnet Model accuracy

Epoch	Train loss	Train accuracy	Val_loss	Val_accuracy
06	0.122	96.04%	0.166	95.03%
07	0.064	97.86%	0.199	96.30%
08	0.042	98.53%	0.095	96.73%
09	0.021	99.35%	0.082	97.16%
10	0.019	99.49%	0.088	97.09%

- Test Loss: 0.698
- Test Accuracy: 85.62%

99.49% training accuracy and 85.62% test accuracy were able to be reached using this particular model as shown in the examples above with the predictions for the test data shown below. As for the hyperparameters for this model, 10 epochs, and 32 batch size were chosen as this gives the best accuracies, while the number of classes was 9 due to the 8 weed species and another for negative samples. The learning rate was then chosen to be 0.001 as the learning rate finder function gave that amount as the number with the lowest error rate. Adam optimizer was also used here when it comes to model compilation due to the previously mentioned reasons with categorical cross entropy as this has multiple classes. This was done for both test and train portions of the dataset in the 20% and 80% split respectively. As for the valid set, a further 0.1 was taken from the train set. 32 batch size was chosen and according to research after providing the best level of predictions with shuffling enabled.

Table 5. U-Net Model accuracy

Epoch	Loss	accuracy	Val_loss	Val_accuracy
16/20	0.1409	0.9458	2.7952	0.6428
17/20	0.1406	0.9461	2.7834	0.6446
18/20	0.1373	0.9465	2.6865	0.6497
19/20	0.1338	0.9479	2.8509	0.6445
20/20	0.1298	0.9488	3.0347	0.6439

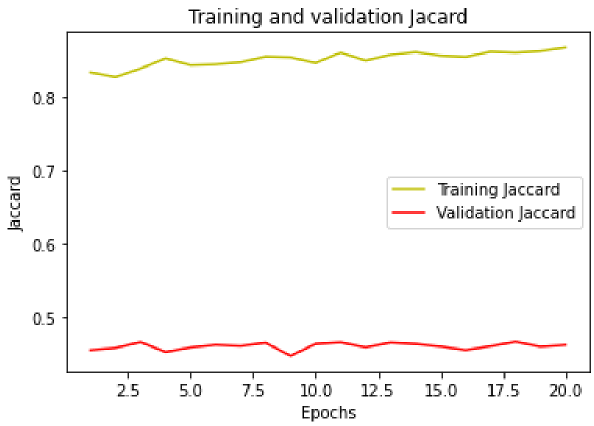

Fig 6: Jaccard accuracy chart

The above figure represents training and validation jaccard accuracy. The y-axis depicts the accuracy and the x-axis depicts the number of epochs. Although validation jaccard remains relatively the same value, the training jaccard accuracy only gets higher with the later epochs. Seeing as the training accuracy only gets higher with a large number of epochs it is apparent that nearly 20 epochs would be ideal for this situation.

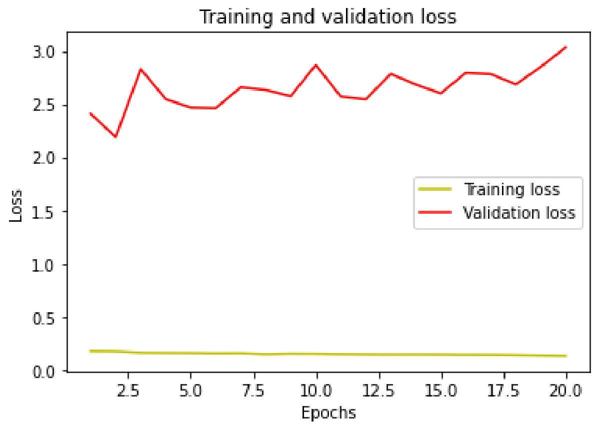

Fig 7: Training and Validation loss chart

The above figure represents training and validation loss. The y-axis depicts the loss and the x-axis depicts the number of epochs. The validation loss changes on a small scale but it is apparent that the training loss is only steadily reduced. Similarly to the jaccard accuracy chart near the 20th epoch is the optimal zone. A training accuracy of 94.88% and a validation accuracy of 64.39% was able to be reached after 20 epochs and 32 batch size. These numbers were chosen after performing hyperparameter tuning in the methodology.

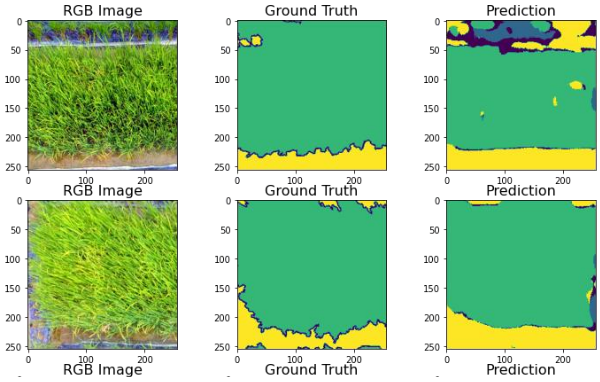

Fig 8: Weed mapping output

4.4.4 Provide fertilization solutions according to the size of paddy field and the fertilizer type

Table 6. Model accuracy

Epoch	loss	accuracy	Val_loss	Val_accu racy
86/90	0.2031	0.8649	0.2297	0.8438
87/90	0.1995	0.8906	0.3161	0.8125
88/90	0.2211	0.8750	0.2790	0.8438
89/90	0.2056	0.9062	0.2989	0.8438
90/90	0.2444	0.8919	0.3199	0.8750

- Loss: 0.2241/Accuracy: 94.20%

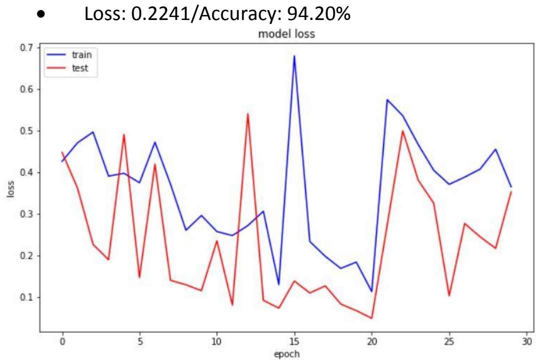

Fig 9: Training accuracy chart

The above figure represents training and validation accuracy. The y-axis depicts the accuracy and the x-axis depicts the number of epochs. In this graph, both the training and validation oscillate very frequently but near the end, they both merge and provide a steady result. 95.23% training accuracy and 94.20% test accuracy were able to be reached using this model as shown in the examples above with the predictions for the test data shown below. Finally, in order to get the maximum test and the training accuracy hyperparameters were tuned accordingly, Batch size – 32,Epoch– 90Maximum accuracy was achieved, according to the previously mentioned configurations

Fig 10: Fertilizer type prediction output

4.4.5 Rice crop growth identification and providing solutions

Table 7. AlexNet model accuracy

Epoch	loss	accuracy	Val_loss	Val_accu racy
16/20	0.3699	0.8561	0.1020	0.9786
17/20	0.3874	0.8615	0.2762	0.9329
18/20	0.4067	0.8510	0.2438	0.9486
19/20	0.4546	0.8284	0.2161	0.9529
20/20	0.3642	0.8639	0.3518	0.9600

- Loss: 0.3518
- Accuracy:96.00%

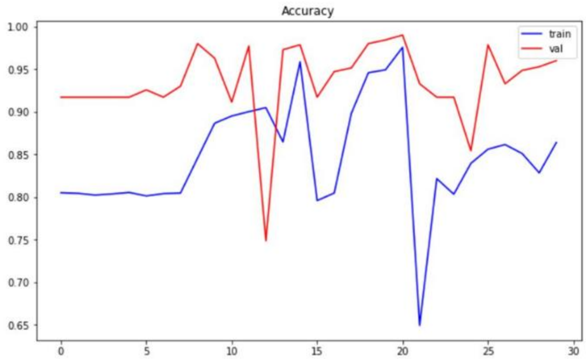

Fig 11: Training accuracy chart

The above figure represents training and validation accuracy. The y-axis depicts the accuracy, and the x-axis depicts the number of epochs. This graph shows a huge drop midway through the number of epochs for both lines. But it recuperates later and gives a better result.

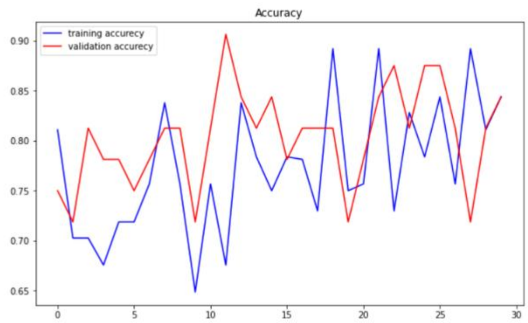

Fig 12: Training loss chart

The above figure represents training and validation loss. The y-axis depicts the loss and the x-axis depicts the number of epochs. In a manner, to the previous graph, this graph also suffers in the middle. Considering both figures 11 and 12 it is apparent that the optimal number of epochs is close to 20. 86.39% training accuracy and 96.00% test accuracy were able to be reached using this model as shown in the examples above with the predictions for the test data shown below. 20 epochs, 0.1 learning rate, and 128 batch size are the hyperparameters that worked best and had the lowest error rate.

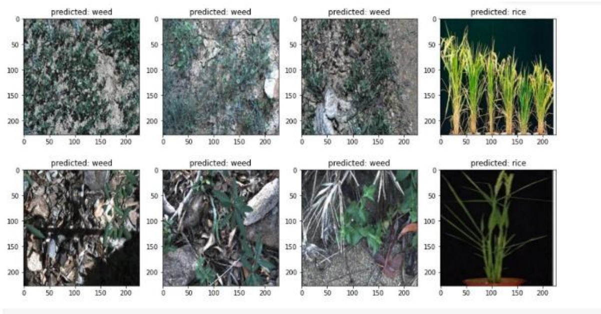

Fig 13: Rice identification output

These are some examples of measured heights using the custom python code.

Fig 14: Height measurement

5. CONCLUSION

This research paper was performed in order to provide rice farmers with solutions to the four major issues that they are currently facing which include pests, disease, weeds, fertilizers, and growth defects. In this research four CNN models are compared and contrasted in order to identify which one of them is best suited when it comes to rice and paddy farm datasets. Considering the outputs provided by four models which are used for image classification, the resnet50(modal 02) model performed best with it providing 99.43% for training accuracy and 97.04% for validation accuracy. Some additional research has also been done for the purpose of creating approaches for weed mapping, rice plant height measurement, and area calculation of the paddy fields. In order to map the weed u-net, FCN architecture was used and calibrated, with it providing 94.88% training accuracy and 64.39% validation accuracy. Hight measuring and area calculation was implemented and finetuned using a custom-made python operation.

6. REFERENCES

[1] Johnny L. Miranda, B. Gerardo and Bartolome T. Tanguilig, “Pest Identification using Image Processing Techniques in Detecting Image Pattern through Neural Network,” International Conference on Advances in Computer and Electronics Technology - ACET 2014, Aug. 43 - 48, 2014.

[2] Bivatec Ltd, “Crop Farmers App - Apps on Google Play.” Play.google.com,play.google.com/store/apps/details?id=com.bivatec.cropfarmersguide&hl=en_US.

[3] MojtabaDadashzadeh, Yousef Abbaspour-Gilandeh, T. MesriGundoshmian and Sajad Sabzi, “Weed Classification for Site-Specific Weed Management Using an Automated Stereo Computer-Vision Machine-Learning System in Rice Fields.” Plants, vol. 9, no. 5, 1 May 2020, p. 559

[4] Usman Ahmed, Jerry Chun-WeiLin, Gautam Srivastava, and YoucefDjenouri, “A nutrient recommendation system for soil fertilization based on evolutionary computation,” Computers and Electronics in Agriculture, vol. 189, no. 106407, p. 106407, 2021

[5] TanakornSritarapipat, PreesanRakwatin and TeerasitKasetkasem, "Automatic Rice Crop Height Measurement Using a Field Server and Digital Image Processing," Sensors , vol. 14, pp. 900-926, 2014

[6] I. IbrahemKandel and Mauro Castelli, “The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset,” ICT Express, vol. 6, no. 4, pp. 312–315, 2020

[7] T. Gayathri Devi, A. Srinivasan, S. Sudha, and D. Narasimhan, “Web enabled paddy disease detection using Compressed Sensing,” Mathematical Biosciences and Engineering, vol. 16, no. 6, pp. 7719–7733, 2019

[8] Mohammad SadeghEbrahimi and HosseinKarkehAbadi, “Study of residual networks for image recognition,” Intelligent Computing, Proceedings of the 2021 Computing Conference, Volume 2, 2021, pp. 754–763

[9] Olaf Ronneberger, Philipp Fischer and Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science(), vol 9351

[10] Y. Lu, “Image classification algorithm based on improved AlexNet in cloud computing environment,” in 2020 IEEE International Conference on Industrial Application of Artificial Intelligence (IAAI), 2020, pp. 250–253.