Crop Pest Recognition Using Image Processing

by

Pial Ghosh
19201069
Istiak Ahmed Alin
19201087
Hasan al mahmud Chowdhury
23141069

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
October 2024

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Pial Ghosh

19201069

Istiak Ahmed Alin

19201087

Hasan al mahmud Chowdhury

23141069

Approval

The thesis/project titled “Crop Pest Recognition Using Image Processing” submitted by

1. Pial Ghosh(19201069)
2. Istiak Ahmed Alin(19201087)
3. Hasan al mahmud Chowdhury(23141069)

Of Summer, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October 22, 2024.

Examining Committee:

Supervisor:
(Member)

Md Tanzim Reza
Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

One of the most vital aspects of a human's existence is food. Each food contains several nutrients which help in growth and development of the human body. It also prevents our body from various diseases. Most of the food we consume comes from crops, trees and plants. Pests infestation is the biggest threat for the agriculture sector.It can cause various types of diseases in crops and reduce crop production. As a result, it is necessary to detect pests early and take necessary steps to stop the infestation. For decades, humans used traditional manual techniques to detect pests. However, this technique is very time consuming, laborious and less accurate. With the development of deep learning , pest detection has become easier than the traditional techniques.The aim of this study is to propose a novel model named VGG19-KAN for crop pest detection and compare it with the State-of-the-arts model like Mobilenetv2 and VGG19. We used the IP102 dataset to train the model. We divided the pest images of IP102 into 8 crop types: rice, corn, wheat, beet, alfalfa, vitis, citrus, and mango. This paper highlights the potential of the VGG19-KAN model. For example, we trained VGG19-KAN along with Mobilenetv2 and VGG19 and found that VGG19-KAN performed much better than Mobilenetv2 and VGG19 in Mango class.The training accuracy of VGG19-KAN was 98.07%.

Keywords: Computational power; Human body; food; Pests; Crops; Infestation; Production; Development; Disease

Table of Contents

Chapter 1

Introduction

The population of the world is predicted to increase to 9.7 billion in 2050 and may peak at over 11 billion in 2100. This means that in order to feed the growing population, there has to be a sufficient supply of food while minimizing crop damage [4]. In agriculture, pests have always been a big issue for decades. Because it causes a lot of harm to the crops. So, pest monitoring has become very important.There are millions of species found in the world which is the biggest issue for pest control. As it is hard to identify an insect pest. In the past, people used to implement manual techniques to identify pests which are very time consuming , laborious and have a chance to cause errors to identify [1][2]. If somehow farmers can detect insects before attacking the crop field can help them to minimize the crop damage and increase crop production.Due to development of science and technology , computer and machine learning techniques have become very popular in the agriculture sector [5][4]. Machine learning can give accurate information about pest insects and saves a lot of time which was the biggest disadvantage of manual techniques[2].

1.1 Problem Statement

Pest attacks are one of the biggest issues in the agriculture sector. Pest attacks can cause severe damage and disease to crops and lower the production [7]. While there are many deep learning techniques that exist for pest detection, we need more research to address the issue. As the world population is going to reach 9.7 billion by the year 2050, as a result it will put pressure on livable agriculture to produce adequate food. The agricultural industry has been plagued with one major ongoing problem: pest infection. It reduces the crop yield to an enormous amount and has been one of the major food security concerns. Pests destroy crops and bring losses to farmers. The major drawback with the conventional methodologies of pest detection, based on manual visual inspection, is that they are inefficient, time-consuming, and prone to many errors. It triggers failure in the management of pest outbreaks. These disadvantages are very crucial in large-scale farming, as delays in timely detection of pests may spread widely and cause damage[3].

Out of the nearly a million species of pests in the world, it is extremely cumbersome to identify an infestation and its controls using manually applied methods. Delayed or wrong identification results in massive losses of crops. In respect to the shortcomings of the manual methods of pest detection, there exists a high demand for automated techniques that can better the accuracy, speed, and efficiency of the operation of detection. The recent advances in machine learning and deep learning introduce promising methods by automating pest detection through image processing. Though promising CNN models have come forward, even VGG19 models that handle this problem often lack the precision to handle diverse pest species across multiple crop types. These will underperform in a particular category of crops, hence limiting their practical application in diverse agricultural settings.

Our proposed study, therefore, brings an enhanced deep learning model VGG19-KAN to offset the deficiency within the current methods applied for pest detection. This would increase the accuracy of the detection of pest-attack vulnerable major crops like rice, corn, and wheat. Our research paper will, therefore, present the performance of the VGG19-KAN on the IP102 dataset in comparison with state-of-the-art models like Mobilenet V2 and VGG19 to establish superior performance in the detection of pests on a wide range of crops while reducing time-consuming efforts in managing the pest.

1.2 Research objectives

Bangladesh is an agriculture based country. A large portion of the economy comes from agriculture. Every year many pests such as - Rice yellow stem borer, Dark headed stem borer, green leaf hopper, pink borer, jute white mite, spotted bollworm etc. attack the field of the crops. As a result, the production of the crop is hampered and farmers have to face a huge loss.Pest infestation affects the economy of our country. The main goal of this esearch is to propose a novel architecture named VGG19-KAN to identify the harmful pests and stop the infestation. So, crop production increases. The key points of our research are as follows :

- To collect a secondary dataset for our research purpose.
- To train the VGG19-KAN with the collected dataset.
- To compare the VGG19-KAN model with state of the art algorithms.

Chapter 2

Related Work

The Groundnut Vision Transformer (GNViT) model to detect and classify pests. Vision Transformers (ViTs) over traditional Convolutional Neural Networks (CNNs) is well articulated, laying a solid foundation for this paper [19].The methodology section is comprehensive, detailing the dataset, the GNViT model, and the experimental setup. The use of the IP102 dataset which includes images of various pests affecting groundnut crops, provides a robust base for training and evaluation. The GNViT model uses a pre-trained ViT model for the pest classification. The authors have clearly explained the data reprocessing steps, data augmentation techniques and the model training process.The dataset has 4157 images categorized into four classes (Thrips, Aphids, Armyworms, and Wireworms).Techniques like resizing, contrast enhancement, and Error Level Analysis (ELA) ensure the dataset's integrity and improve model accuracy.Various transformations (rotation, flipping, color adjustments) are applied to increase the dataset's variability and prevent overfitting.The GNViT model adapts the ViT architecture, employing the Adam optimizer and CrossEntropy Loss function, with detailed pseudo-code provided.The results demonstrate the GNViT model's high accuracy (99.52%) and its superiority over other models such as LeNet, AlexNet, GoogleNet, and ResNet. The GNViT model's performance is evaluated using metrics like accuracy, precision, recall and F1-score, with notable improvements in training accuracy from 55.37% to 99.52%.

The paper [14] starts by highlighting the significance of tomato farming in the global agribusiness landscape and the crucial role of early disease detection in preventing yield losses. It discusses the limitations of traditional visual classification techniques and the potential of computer technologies, specifically machine learning and deep learning, to automate disease identification in tomato plants .The study uses two deep learning models, Inception V3 and Vision Transformer (ViT), to classify diseases in tomato leaves. The dataset is derived from the Plant Village dataset, which includes 10,010 images of tomato leaves across 10 different classes (nine disease classes and one healthy class). The methodology is detailed and systematic which involves data preprocessing, model training and evaluation. The use of transfer learning with the Inception V3 model and the self-attention mechanism in ViT is well-explained which provides a clear understanding of the model architectures and their implementation.The results indicate that the ViT outperforms Inception V3 in both accuracy and loss metrics. The ViT achieves a training accuracy of 97.37% and a validation accuracy of 95.76% and compared to Inception V3's training accuracy of 89.24% and validation accuracy of 88.98%.

[13]The paper proposes an Enhanced Vision Transformer Architecture (EViTA) model which is designed for pest identification, segmentation and classification. The model leverages the capabilities of Vision Transformers (ViT) and incorporates a novel dual-layer transformer encoder to handle various segment sizes of pest images. The proposed methodology utilizes three pest datasets : Aphids (IP102 Dataset), Wireworm (IP102 Dataset) and Gram Caterpillar collected from publicly available repositories. The paper employs Moth Flame Optimization for feature extraction and StandardScaler for data normalization to improve the prediction accuracy.The model's performance is evaluated using various metrics like accuracy, precision, sensitivity, specificity, F1 score, Mean Absolute Error (MAE) and Mean Squared Error (MSE), providing a thorough assessment.The results are compared against several state-of-the-art CNN models, demonstrating the superiority of the proposed EViTA model in terms of prediction accuracy.In methodology,The process of segmenting images into tokens and the role of positional encoding should be elaborated on.More details on the MFO algorithm's implementation and how it improves feature selection are needed.The explanation of the EViTA model, including the role of the Extra Arrangement Segment (EAS) and the integration with the MFO which should be clearer.

The paper [16] addresses the critical issue of food security and crop yield by focusing on the timely and accurate identification of plant diseases. It proposes a novel approach combining Vision Transformer (ViT) for high identification accuracy and GPipe for enhanced running speed. The experiments utilize the PlantVillage dataset, and results show that ViT achieves a 93% recognition accuracy, with pipeline parallelism significantly reducing memory requirements. This research offers a low-cost and efficient plant disease identification tool beneficial for agriculture.The ViT model is chosen for its accuracy. It processes image data by dividing it into patches, mapping these to vectors, and using a Transformer encoder with self-attention mechanisms to capture image relationships. The MLP Head classifies the processed data. In terms of GPipe, This scalable training method partitions the model across multiple GPUs, allowing pipeline parallelism. This approach improves training efficiency and reduces GPU memory consumption.The study uses three sets of control experiments with different numbers of GPUs to test the impact of pipeline parallelism on model performance.Images from the PlantVillage dataset are preprocessed by resizing, normalizing, and onverting pixel values to ensure uniform input for the models.ViT achieves a 93% accuracy on the color dataset, outperforming ResNet, which achieves 84%.ViT's performance drops on gray images but remains relatively high on segmented images. ResNet's performance is more stable across all datasets.Using GPipe significantly increases throughput and reduces memory usage per GPU, enhancing efficiency.

This paper[6] addresses the pressing need for precise identification and classification of plant diseases and insect pests, which is critical for the advancement of precision and smart agriculture. The proposed solution leverages the Vision Transformer (ViT) neural network, a model initially designed for natural language processing but adapted here for image recognition tasks.The paper employs the Vision Transformer, a cutting-edge architecture known for its superior performance in image recognition tasks, particularly when dealing with large datasets.

[11]In this paper, the performance of KAN and MLP networks is compared on various types of irregular and noisy functions.To ensure fairness, the number of parameters and training sample sizes are controlled.The results shows that KAN does not always outperform MLP but sometimes it does exceed KAN's performance.

[20] In this paper,the authors introduce Wav-KAN, a neural network architecture that integrates wavelet functions into the KAN framework to improve interpretability,robustness and performance.In their paper, this wavelet-based approach enables faster training speed and enhanced accuracy. In the result, Wav-KAN's potential as a powerful tool for creating interpretable and high performance neural networks with applications across various domains[17].

The study [15] addresses the challenge of early plant disease detection using an improved Vision Transformer network, TrIncNet. It replaces the computation expensive MLP module of traditional ViTs with a more efficient Inception module which improves feature extraction and reduces the number of trainable parameters. The model's architecture has skip connections to eradicate the vanishing gradient problem. TrIncNet's performance was validated using the PlantVillage and Maize disease datasets, where it outperformed other CNN architectures and ViTs, achieving higher accuracy and efficiency. As a result,TrIncNet is suitable for integration with IoT devices for real-time plant disease detection.The TrIncNet model achieved the highest validation accuracy (97.0%) and lowest validation loss (0.035) on the Maize dataset. It also demonstrated superior results in accuracy, precision, recall, and F1-score. TrIncNet required significantly fewer trainable weight parameters (6.95 million) compared to the ViT model (7.12 million).On the PlantVillage dataset, TrIncNet attained the highest validation accuracy (99.95%) and lowest validation loss (0.02). It outperformed other models with 99.93% accuracy, 99.92% precision, 99.91% recall, and 99.91% F1-score.

The paper [8] focuses on the PlantCLEF2022 challenge, where the goal is to identify plant species from a large-scale dataset comprising millions of images and 80,000 classes. Given the limited number of images per class (average of 36), this task is treated as a few-shot image classification. Instead of using the traditional convolutional neural networks (CNNs), the authors employ a self-supervised Vision Transformer (ViT) model. This approach secured first place in the challenge, achieving a Macro Averaged Mean Reciprocal Rank (MA-MRR) of 0.62692. The self-supervised ViT offers advantages over CNNs, including no inductive biases and better feature extraction for downstream tasks.The PlantCLEF2022 challenge provides a platform to address these issues using a dataset of 80,000 classes and millions of images.The PlantCLEF2022 dataset is divided into training and testing sets. The training dataset consists of web images (1.1 million images, 57,000 classes) and trusted images (2,885,052 images, 80,000 classes). The trusted dataset is preferred for training due to higher annotation quality. The testing dataset includes 55,306 images from 26,868 observations. The evaluation metric used is the MA-MRR, which accounts for multiple chances to recognize plant species, making it suitable for observation-level classification.The results demonstrate the effectiveness of the self-supervised ViT approach, which achieved first place in the challenge. The MA-MRR score of 0.62692 was significantly higher than the second (0.60792) and third place (0.51092). Extending the training by twenty epochs further improved the score to 0.64079. The pretrained model also showed promising results in plant disease recognition tasks on four public datasets, outperforming the ImageNet pretrained model.

In the novel hybrid model called ”PlantXViT” that combines CNN and ViT for plant disease detection. The model is designed to increase the local feature extraction capabilities of CNNs and the global feature extraction capabilities of Vision Transformers to improve the accuracy. This model is very efficient in plant disease identification. PlantXViT is a lightweight model which is suitable for IoT-based smart agriculture services. The model trained on five publicly available datasets where it showed impressive performance compared to state-of-the-art methods.The combination of CNN and ViT allows efficient extraction of both local and global features, which is important for accurate plant disease detection.With only 850,500 trainable parameters, PlantXViT is very much optimized to use in IoT devices which makes it practical for real-world applications in agriculture sector.The model achieves impressive accuracy getting 93.55% on the Apple dataset, 92.59% on the Maize dataset, and 98.33% on the Rice dataset.The use of gradient-weighted class activation maps (Grad-CAM) and Local Interpretable Model Agnostic Explanation (LIME) enhances the model’s predictions, which is important to trust AI decisions [20].

This paper [3] explains about a method for detecting plant leaf diseases by combining a hybrid segmentation algorithm and CNN. This method integrates hue, saturation, intensity (HSI) and LAB color spaces for effective segmentation of disease symptoms from complex backgrounds. This approach leverages the compatibility of these color spaces with human vision to enhance segmentation accuracy. Following segmentation, a modified CNN inspired by AlexNet. But optimized for computational efficiency, classifies the segmented images. The method was tested on a diverse dataset of around 1000 images which achieved a validation accuracy of 91.33% which is 15.51% higher than conventional methods. This significant improvement underscores the method’s potential in practical agricultural applications and contributes to better management and prevention of plant diseases and ultimately supporting food security. Future research could expand the dataset and refine the algorithm for real-time application and include more comparative analyses with other advanced methods.

This thesis [10]addresses the global agricultural challenge of crop pest detection by proposing an advanced deep learning model. The proposed ResNet-50-PCSA model combines a parallel attention mechanism with residual blocks to enhance accuracy and real-time performance. Expensive experiments showcase its effectiveness ,achieving a remarkable accuracy of 98.17% on crop pest image and demonstrating adaptability to rice leaf diseases. ResNet-50 is chosen for feature extraction and data augmentation techniques are employed for robust training. The ResNet-50-PCSA model utilizes a bottleneck-PCSA model within ResNet-50 to refine features which a carefully curated dataset comprises 10 common crop pests and contributes to the model and its accuracy is 98.17%. Comparative experiments with SENet and CBAM highlight the efficacy of the parallel attention mechanism. The model's adaptability is tested on a public dataset of rice leaf diseases, achieving a high accuracy of 99.35%.Comparative experiments with other CNN models showcase the ResNet-50-PCSA[12] model's versatility beyond pest recognition.

Crop pest recognition in the field introduces a pioneering method,MCapsNet which is aiming to overcome the challenges posed by traditional methods and enhance crop protection. By incorporating a modified Capsule Network and an attention mechanism, the proposed approach demonstrates superior performance compared to traditional convolutional neural networks(CNNs) and CapsNet. The attention mechanism proves effective in capturing crucial classification features and MCapsNet exhibits success in classifying diverse insect types in field crops.By applying Modified Capsule Network (MCapsNet) with an attention mechanism for crop pest recognition in field images.MCapsNet leverages deep learning capabilities, specifically Capsule Networks to automatically extract invariant features from diverse pest images which points out mechanism enhances feature extraction by focusing on relevant contextual relationships and the LeakyReLU activation function accelerates model convergence. Through experiments on a dataset comprising images of common crop pests, MCapsNet demonstrates superior accuracy and recall compared to other CNN models. The key contributions lie in the effective integration of Capsule Networks, attention mechanisms and LeakyReLU activation for robust crop pest recognition which offers a promising solution for practical implementation in agriculture to enhance crop protection [9].

The paper discusses a framework designed to connect gaps between the symbolism of science and the connectionism of AI in integrating KANs into scientific discovery. This approach allows for a two-way flow: while KANs identify relevant features, unravel modular structures, and therefore deduce symbolic formulae; it can embed scientific knowledge into the KANs and interpret from them. So far, pykan, a Python library, has implemented several key functionalities including MultKAN: KANs with multiplication nodes, kanpiler: a tool to compile symbolic formulas to KANs; and tree converter: to convert KANs or neural networks to tree graphs. It is apparent that the framework shows the capability of KANs in discovering a range of physical laws, which includes but is not limited to conserved quantities, Lagrangians, symmetries, and constitutive laws[17].

Chapter 3

Methodology

The dataset first needed to preprocess before loading into the model. The images were splitted into training, testing and validation. The training, testing and validation images were divided into 8 crop types. Then we preprocessed the image using data augmentation. After preprocessing the images, we loaded them into the VGG19-KAN along with VGG19 and Mobilenet v2 for training.Then we compared the models with each other.

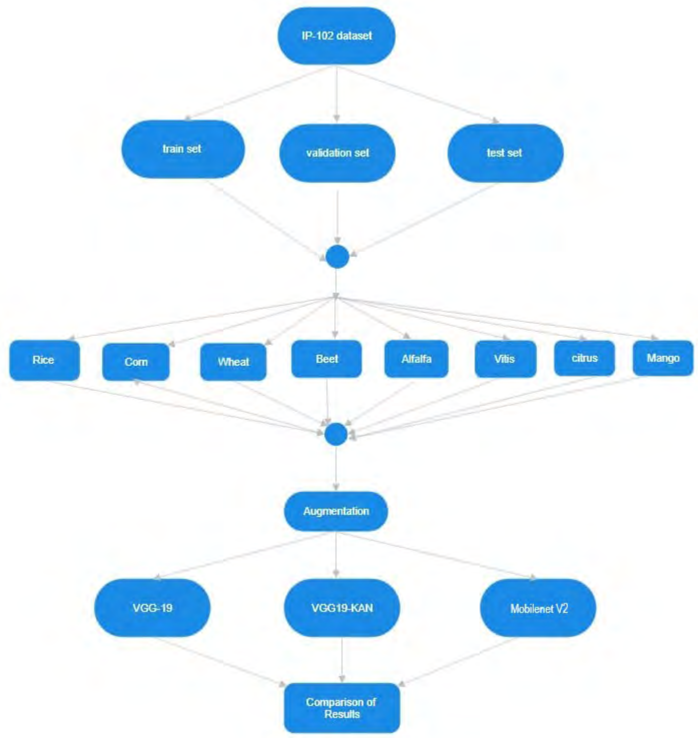

Figure 3.1: Methology-flowchart

3.1 Dataset

We used the IP102 dataset which is one of the largest dataset available for insect classification. This dataset has 75,222 images belonging to 102 classes and an average size of 737 per class. The dataset has a split of 6:1:3. There are super classes which are Rice, Corn, Wheat, Beet, Alfalfa, Vitis, Citrus, Mango.

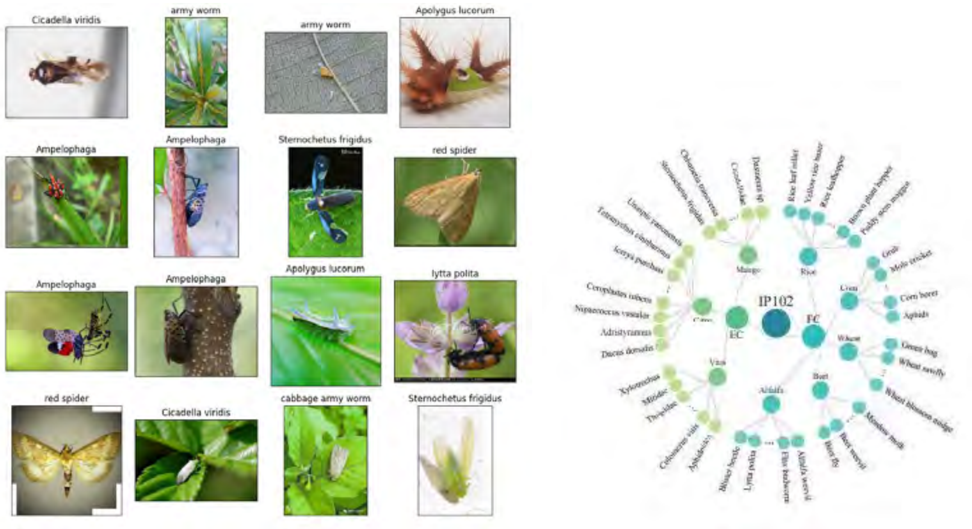

Figure 3.2: Dataset

3.2 Data preprocessing

Different image augmentations were performed on the training, validation, and test datasets during data preprocessing. For the training dataset, images were resized to 224x224 pixels. To provide a further boost, random horizontal flips and rotations of up to 90 degrees introduced more improvements in order to increase the model's generalization. Color jitters were applied that randomly changed brightness and contrast by ±0.2, thus emulating several types of changes in lighting. In particular, it trains the model to pay attention to different parts of the image by performing a RandomResizedCrop that extracts a random portion of the image and resizes it to 224x224. Moreover, it removes small random areas of an image-this is called Random Erasing. As a result, this method introduces occlusion and hence is robust. Finally, all these images are translated into tensors for further processing in PyTorch.

Super-Class		Class	Train	Val	Test	IR
FC	Rice	14	5,043	843	2,531	6.4
	Corn	13	8,404	1,399	4,212	27.9
	Wheat	9	2,048	340	1,030	5.2
	Beet	8	2,649	441	1,330	15.4
	Alfalfa	13	6,230	1,037	3,123	10.7
EC	Vitis	16	10,525	1,752	5,274	74.8
	Citrus	19	4,356	725	2,192	17.6
	Mango	10	5,840	971	2,927	61.7

Figure 3.3: Dataset

These transformations are lighter for both the validation set and the test set to avoid adding more variance during the process of evaluation. The preprocessing here is the same as in training: resizing images to 224x224 pixels and then transforming them into tensors. This augmented training set provides better generalization, while the consistent preprocessing here ensures a reliable model evaluation.

3.3 Proposed model

3.3.1 KAN(Kolmogorov Arnold Network)

Kolmogorov Arnold Network was proposed by the author of the paper [18]. They claimed that it is a potential alternative to MLP(Multi layer Perceptron). KAN has a strong mathematical foundation similar to the MLP. MLP is based on the universal approximation theorem. KAN is based on the Kolmogorov Arnold theorem. The key difference between KAN and MLP is that MLP have learnable weights on edges and fixed activation function on neurons. On the other hand, KAN has a learnable activation function on edges instead of neurons.

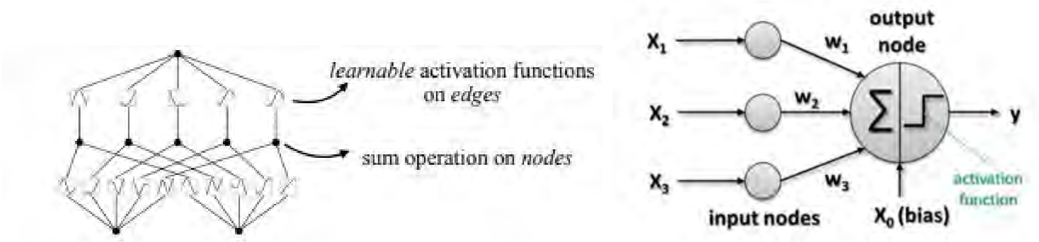

Figure 3.4: KAN & MLP

Table 3.1: Comparison between KAN and MLP

KAN	MLP
It is based on the Kolmogorov Arnold theorem	It is is based on the universal approximation theorem
It has a learnable activation function on edges instead of neurons	It has learnable weights on edges and fixed activation function on neurons

3.3.2 VGG19-KAN

The VGG19KAN model extends the pre-trained VGG19 architecture by appending custom KANLinear layers in place of the fully connected layers of the original VGG19. This model initializes a pre-trained VGG19 that makes use of all the convolutional layers as feature extractors, thus it allows the model to tap into the pre-trained weights for hierarchical feature extraction.

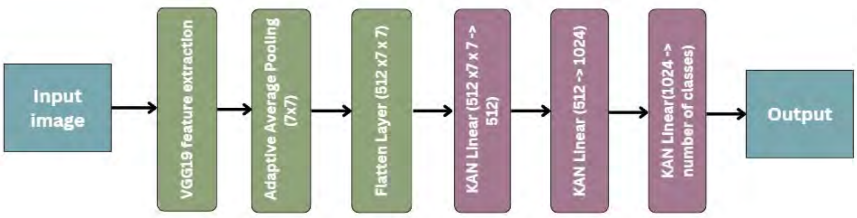

Figure 3.5: Architecture of VGG19-KAN

The first few layers capture low-level features, such as edges or textures, while the rest of the layers result in more abstract features that are typically higher-level. Convolutional layers are placed inside the features module. The model feeds the input through the VGG19 convolutional layers followed by an adaptive average pooling layer. This layer reduces the spatial dimensions of the feature maps to a fixed size of 7x7, preparing them for the fully connected layers and ensuring consistency regardless of the size of an input image. The model in this paper uses three custom KANLinear layers in the place of typical entirely connected layers. This reduces the pooled output coming from the pooling layer, a 25,088-dimensional vector, to 512 dimensions. The second KANLinear layer expands this output to 1024 dimensions. The final KANLinear maps this to the number of classes in the output, for classification. Where as the conventional fully connected layers would offer better interpolation capability through the B-spline interpolation capability of the KANLinear layers, it may be able to learn more complex non-linear relationships better. At the forward pass, the input image is processed through the VGG19 convolutional layers followed by pooling and flattening. Further, the resulting output is passed through the three KANLinear layers in succession to bring out class scores useful for classification purposes. This hybrid is designed to improve the results by combining the adaptive transformability abilities of KANLinear layers with the strengths that VGG19 possesses concerning feature extraction for tasks involving delicate relationships between the features.First few layers capture the low-level features; for instance, edges or textures, and the rest of the layers result in more abstract features, which normally are higher-level. Convolutional layers go inside features module. The input is fed through VGG19 convolutional layers, followed by an adaptive average pooling layer.

3.3.3 Mobilenet V2

MobileNetV2 is a convolutional neural network architecture designed specifically for mobile and edge devices which offers a balance between performance and efficiency. Unlike traditional residual blocks that use a series of convolutionals followed by a shortcut connection MobileNetV2 uses inverted residuals where the shortcut connection is placed between twin bottleneck layers. In this architecture, ReLU6 activation function is used in most layers, providing better performance for mobile and edge devices by being more resistant to low precision computation.

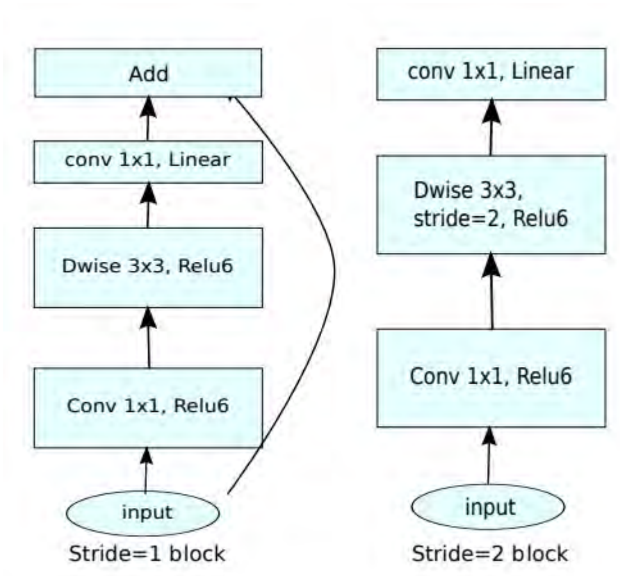

Figure 3.6: Architecture of CNN

In terms of using this architecture, we can see that the design with shortcuts connecting thin bottleneck layers helps in reducing the computational load while preserving the accuracy. Likewise,using a linear layer at the end of each epoch block helps in retaining more information and deducing the loss of representational power. For that reason, MobileNetV2 achieves high efficiency by extensively using depthwise separable convolutions and reducing the number of operations required.

3.3.4 VGG19

VGG19 was a deep CNN architecture that was made up of 19 layers, three fully connected layers, 16 convolutional layers, and one final softmax layer for classification. The small 3x3 convolutional filters with a stride of 1 consecutively cracked. Max-pooling layers reduced the spatial dimensions after every two or more convolutional layers. Whereas in the initial layers the number of filters in convolutional blocks is 64, in the deepest parts it is 512. Following all the convolutional blocks, output is compressed and feeds into two fully connected layers composed of 4096 neurones each. Finally, a third fully connected layer is added in order to output the class probabilities. The network follows each convolution and fully connected layer with ReLU activations and uses the softmax function at the end to ensure multi-class classification. VGG19 is quite effective in image recognition tasks and has an architecture with a lot of uniformity. It is simple in nature. VGG19 is a deep CNN architecture consisting of 19 layers; among these, three are fully connected, sixteen are convolutional layers, and the last one is for classification-softmax. It is a network that uses very small 3×3 convolutional layers stacked consecutively with a stride of 1.Max-pooling layers usually follow two or more convolutional layers, aiming at reducing the dimensions of the input data volume. The number of filters in the convolution blocks constantly increases from 64 in the first layers to 512 in the deepest. After the convolutional blocks, the output is flattened and fed into two fully connected layers comprising 4096 neurones each, while the third fully connected layer will serve to produce class probabilities.It shall then be noticed that the network will use ReLU after every convolution and fully-connected layer for activation, while a softmax function is deployed in the end for multi-class classification. The famous VGG19 is superior to others in image recognition; besides, it is easy to use because of its simple architecture and harmonious structure.

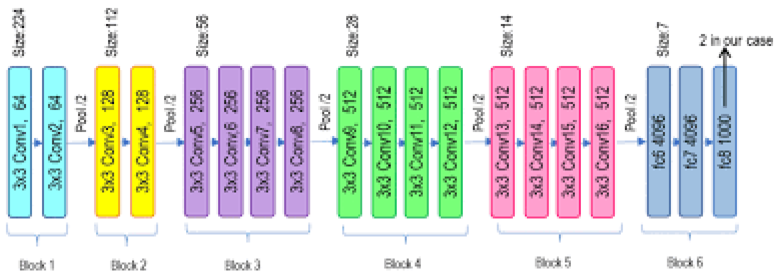

Figure 3.7: Architecture of VGG19

Chapter 4

Results and Discussion

After training the VGG19, CNN and proposed model VGG19-KAN, we determined the training and validation accuracy. Also, we determined F1, precision and recall of the models. The learning rate was 0.0001 and weight decay was 0.0001. The number of epochs was 80. We used Intel Xeon processor and Nvidia Tesla P100 Gpu with 29 GB ram to train these models. We used necessary python libraries like pytorch to train our model.

4.1 Training and validation accuracy of Rice class

There are a total of 14 classes in the Rice super class. There are 5,043 images in training and 843 in validation. After training the models with Rice super class. The training accuracy of VGG19-KAN was 92.33% , VGG19 got 90.03% and Mobilenetv2 got 91.71%. The validation accuracy of VGG19-KAN was 66.63% and VGG19 was 64.77% and Mobilenetv2 was 63.23%. Among the models, VGG19-KAN achieved highest accuracy in both training and validation.

Table 4.1: Comparison between VGG19, MobileNetV2 and VGG19-KAN of rice class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	92.33%	66.63%	0.65	0.65	0.66
VGG19	90.03%	64.77%	0.65	0.64	0.65
Mobilenetv2	91.71%	63.23%	0.63	0.63	0.64

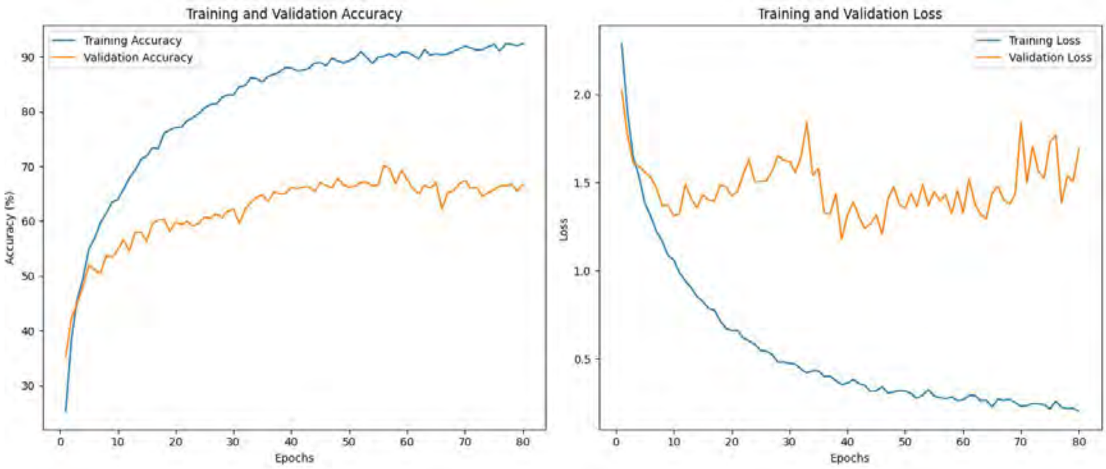

Figure 4.1: Training and Validation accuracy graph of VGG19-KAN

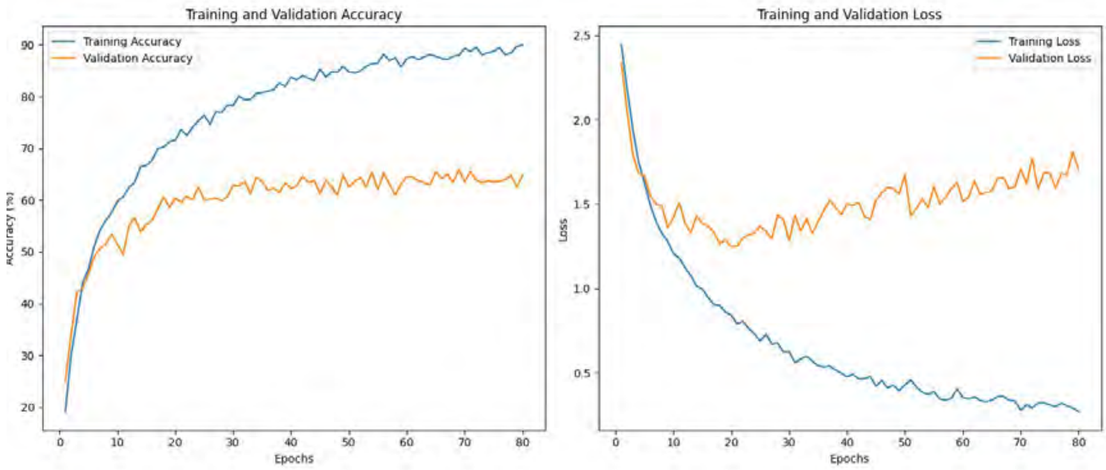

Figure 4.2: Training and Validation accuracy graph of VGG19

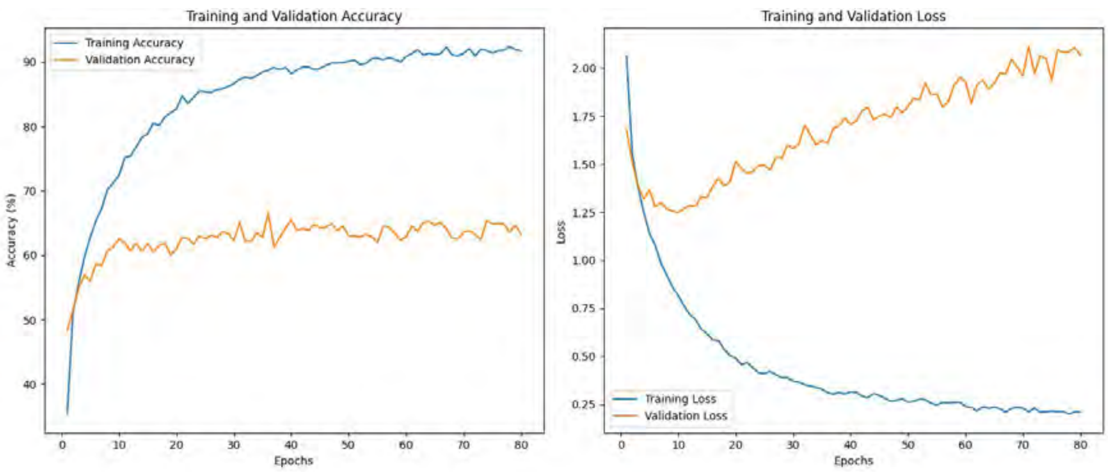

Figure 4.3: Training and Validation accuracy graph of MobileNetV2

4.2 Training and validation accuracy of Corn class

There are a total of 13 classes in the Corn super class. There are 8404 images in training and 1399 in validation. After training the models with Crop super class. The training accuracy of VGG19-KAN was 96.31% ,VGG19 got 95.53% and Mobilenetv2 got 94.43%. The validation accuracy of VGG19-KAN was 83.35% and VGG19 was 80.20% and Mobilenetv2 was 79.56%.Here, The training and validation accuracy of VGG19-KAN is higher than other models.

Table 4.2: Comparison between VGG19, MobileNetV2 and VGG19-KAN of corn class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	96.31%	83.35%	0.74	0.73	0.76
VGG19	95.53%	80.20%	0.70	0.70	0.70
Mobilenetv2	94.43%	79.56%	0.69	0.69	0.69

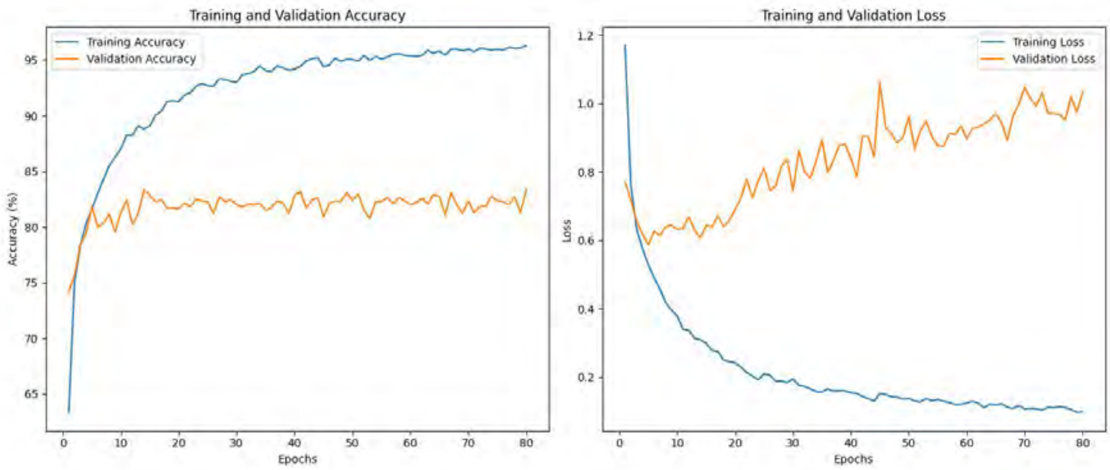

Figure 4.4: Training and Validation accuracy graph of VGG19-KAN

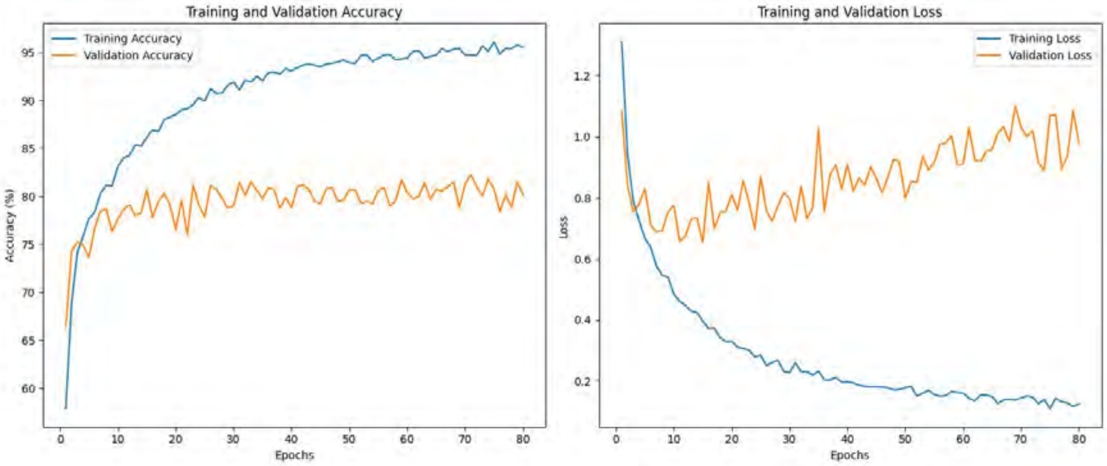

Figure 4.5: Training and Validation accuracy graph of VGG19

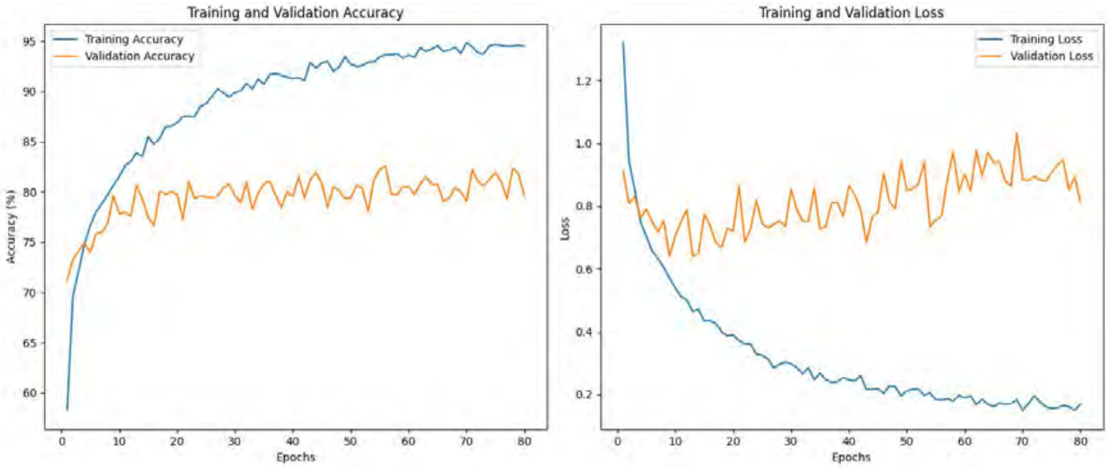

Figure 4.6: Training and Validation accuracy graph of MobileNetV2

4.3 Training and validation accuracy of Wheat class

There are a total of 13 classes in the Wheat super class. There are 2,408 images in training and 340 in validation. After training the models with Wheat super class. The training accuracy of VGG19-KAN was 90.82% ,VGG19 got 92.82% and Mobilenetv2 got 90.62%. The validation accuracy of VGG19-KAN was 67.06% and VGG19 was 67.94% and Mobilenetv2 was 64.41%. Here the validation accuracy is quite similar in case of vgg19-KAN and vgg19 model.

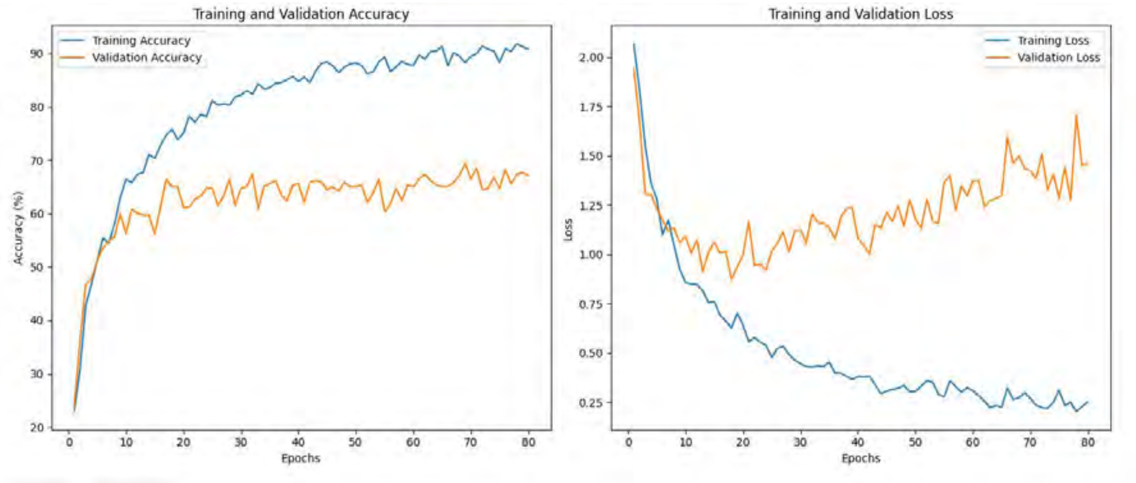

Figure 4.7: Training and Validation accuracy graph of VGG19-KAN

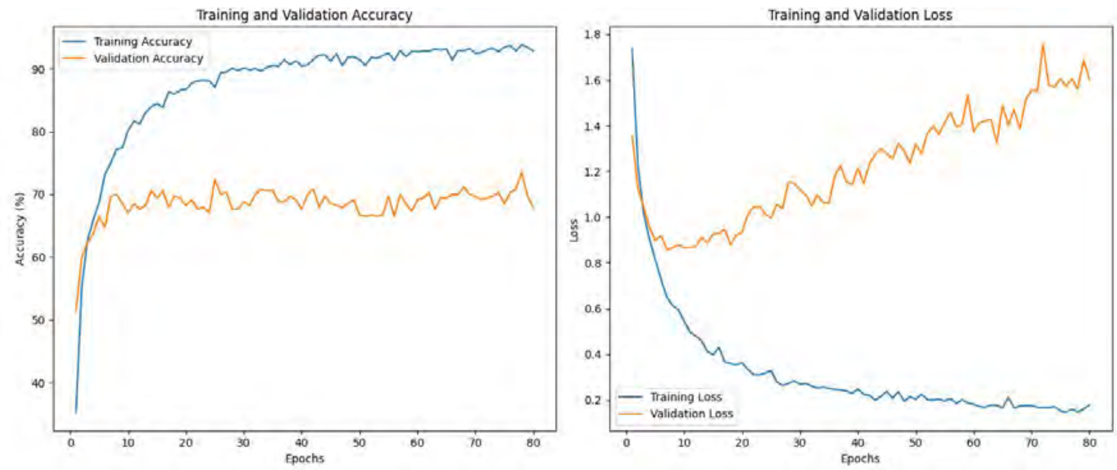

Figure 4.8: Training and Validation accuracy graph of VGG19

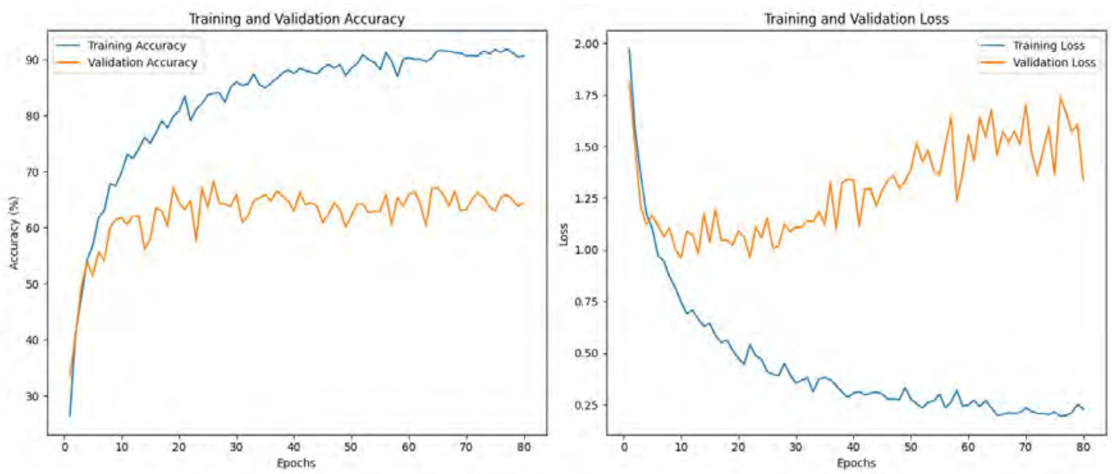

Figure 4.9: Training and Validation accuracy graph of MobileNetV2

Table 4.3: Comparison between VGG19, MobileNetV2 and VGG19-KAN of Wheat class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	90.82%	67.06%	0.70	0.69	0.73
VGG19	92.82%	67.94%	0.70	0.69	0.73
Mobilenetv2	90.62%	64.41%	0.66	0.65	0.67

4.4 Training and validation accuracy of Beet class

There are a total of 8 classes in the Beet super class. There are 2,649 images in training and 441 in validation. After training the models with Beet super class. The training accuracy of VGG19-KAN was 94.71% ,VGG19 got 96.60% and Mobilenetv2 got 94.90%. The validation accuracy of VGG19-KAN was 78.23% and VGG19 was 78.91% and Mobilenetv2 was 74.83%.Here the validation Accuracy of VGG19-KAN and VGG-19 is quite similar but greater than Mobilenetv2.

Table 4.4: Comparison between VGG19, MobileNetV2 and VGG19-KAN of Beet class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	94.71%	78.23%	0.73	0.73	0.75
VGG19	96.60%	78.91%	0.74	0.73	0.77
Mobilenetv2	94.90%	74.83%	0.70	0.68	0.72

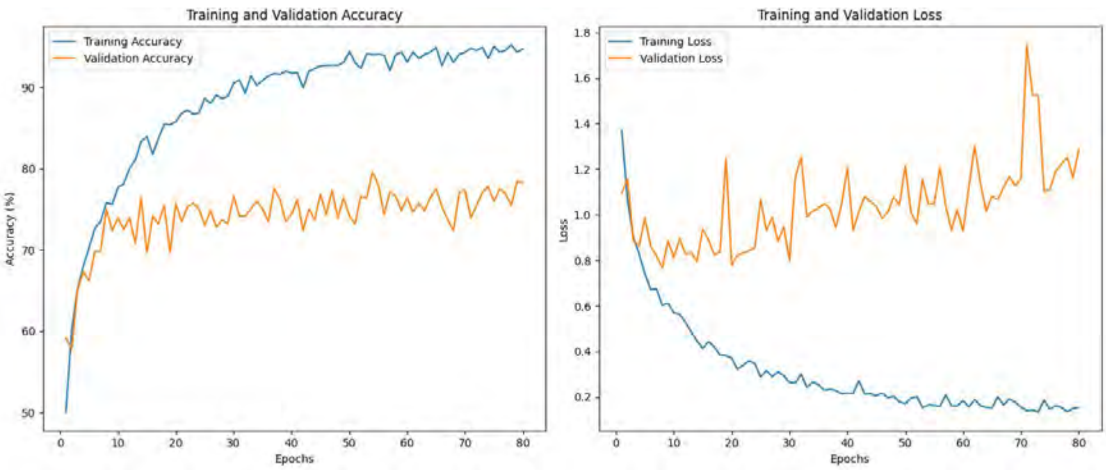

Figure 4.10: Training and Validation accuracy graph of VGG19-KAN

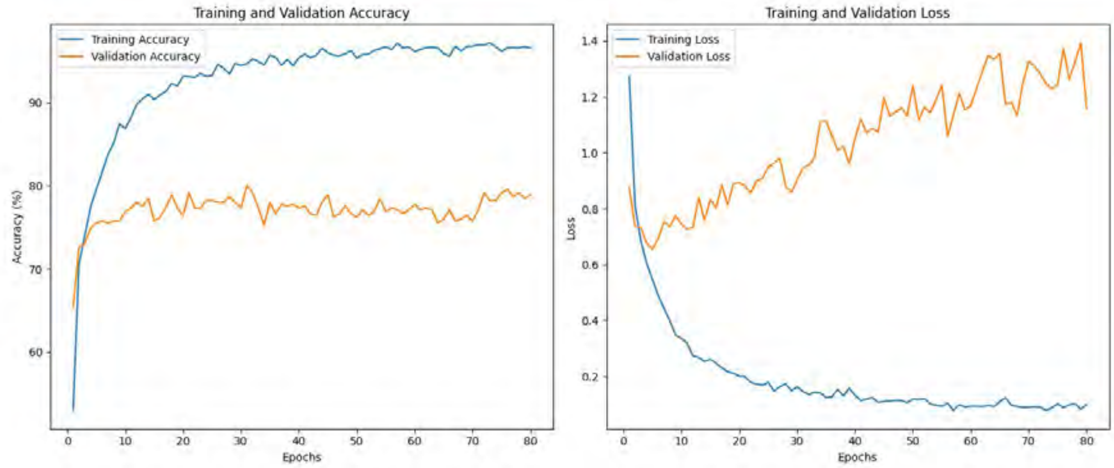

Figure 4.11: Training and Validation accuracy graph of VGG19

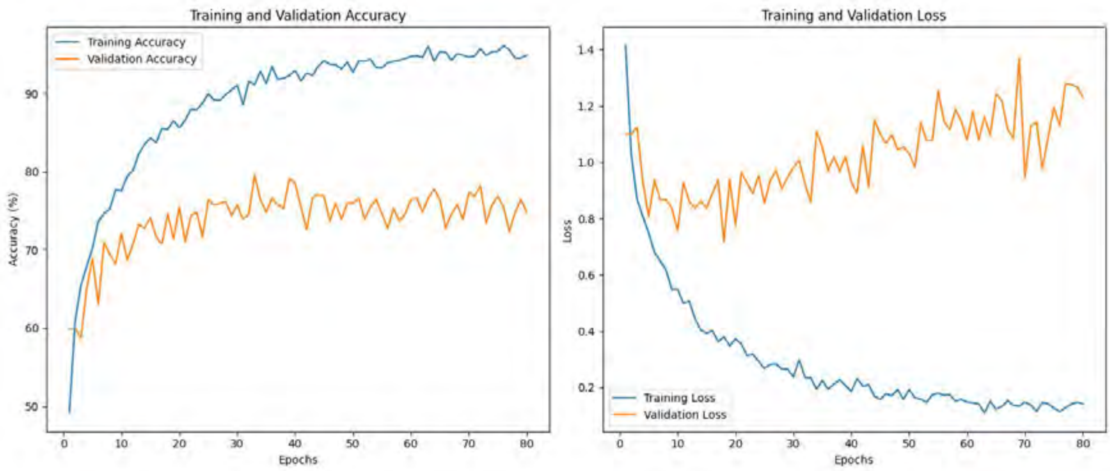

Figure 4.12: Training and Validation accuracy graph of MobileNetv2

4.5 Training and validation accuracy of Alfalfa class

There are a total of 13 classes in the Alfalfa super class. There are 6230 images in training and 1037 in validation. After training the models with Alfalfa super class. The training accuracy of VGG19-KAN was 96.07% ,VGG19 got 94.25% and Mobilenetv2 93.45%. The validation accuracy of VGG19-KAN was 73.10% and VGG19 was 71.94% and Mobilenetv2 was 71.65%. The training accuracy and the validation accuracy VGG19-KAN is higher than other models.

Table 4.5: Comparison between VGG19, MobileNetV2 and VGG19-KAN of Alfalfa class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	96.07%	73.10%	0.70	0.68	0.73
VGG19	94.25%	71.94%	0.70	0.70	0.70
Mobilenetv2	93.45%	71.65%	0.69	0.67	0.72

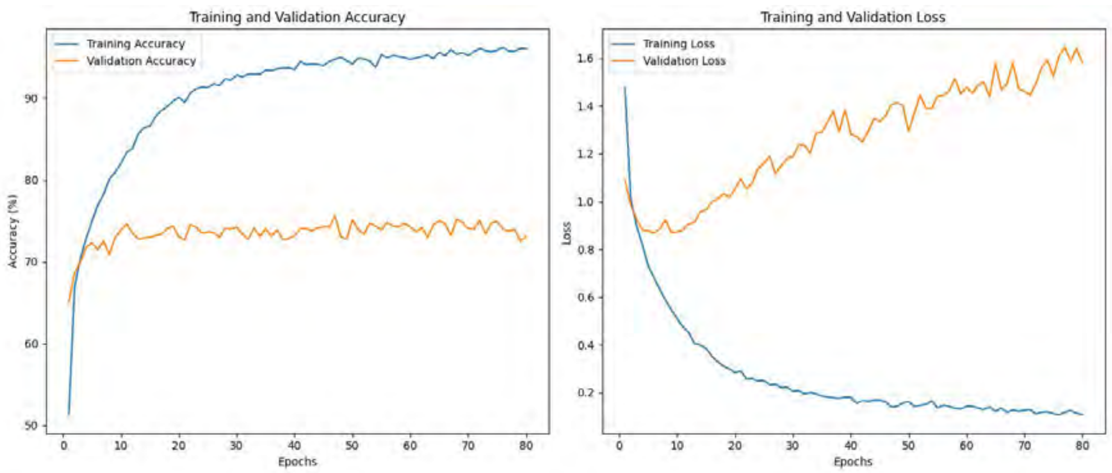

Figure 4.13: Training and Validation accuracy graph of VGG19-KAN

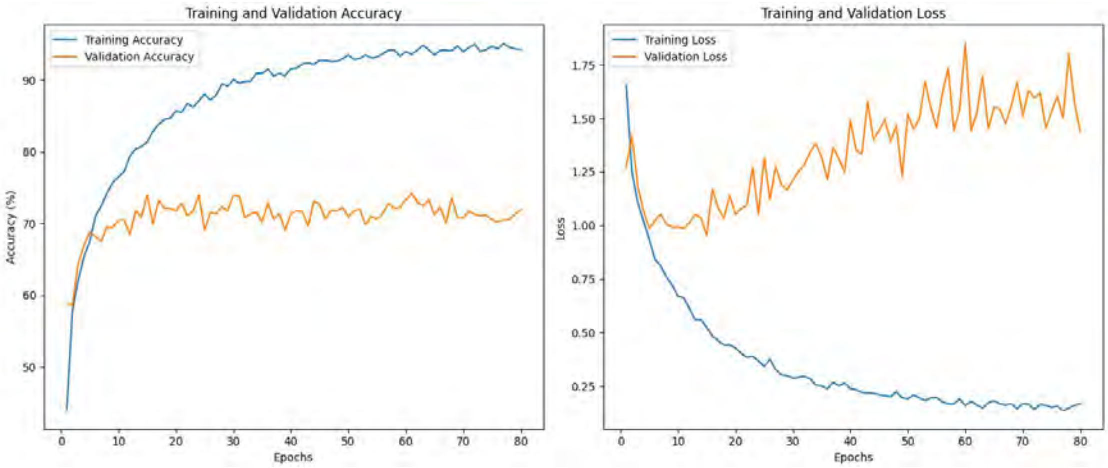

Figure 4.14: Training and Validation accuracy graph of VGG19

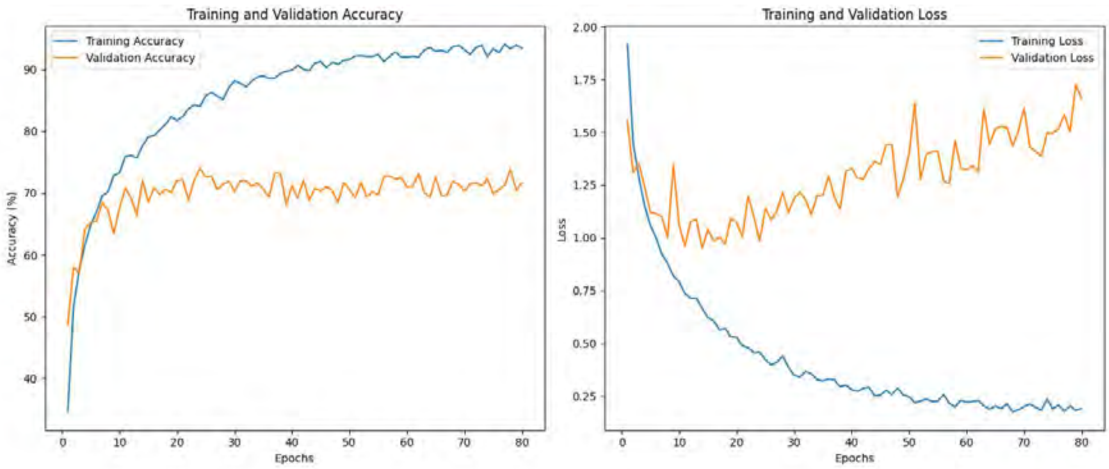

Figure 4.15: Training and Validation accuracy graph of MobileNetV2

4.6 Training and validation accuracy of Vitis class

There are a total of 16 classes in the Vitis super class. There are 6230 images in training and 1037 in validation. After training the models with Vitis super class. The training accuracy of VGG19-KAN was 98.45% ,VGG19 got 97.72% and Mobilenetv2 got 96.31%. The validation accuracy of VGG19-KAN was 88.07% and VGG19 was 87.04% and Mobilenetv2 was 86.64%. The training accuracy and the validation accuracy VGG19-KAN is higher than other models.

Table 4.6: Comparison between VGG19, MobileNetV2 and VGG19-KAN of Vitis class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	98.45%	88.07%	0.79	0.75	0.85
VGG19	97.72%	87.04%	0.75	0.71	0.84
Mobilenetv2	96.31%	86.64%	0.76	0.75	0.78

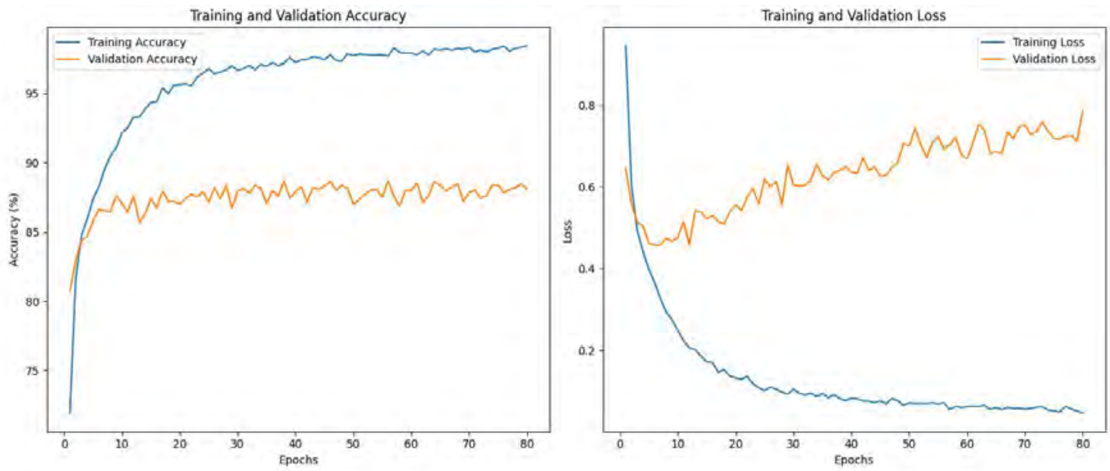

Figure 4.16: Training and Validation accuracy graph of VGG19-KAN

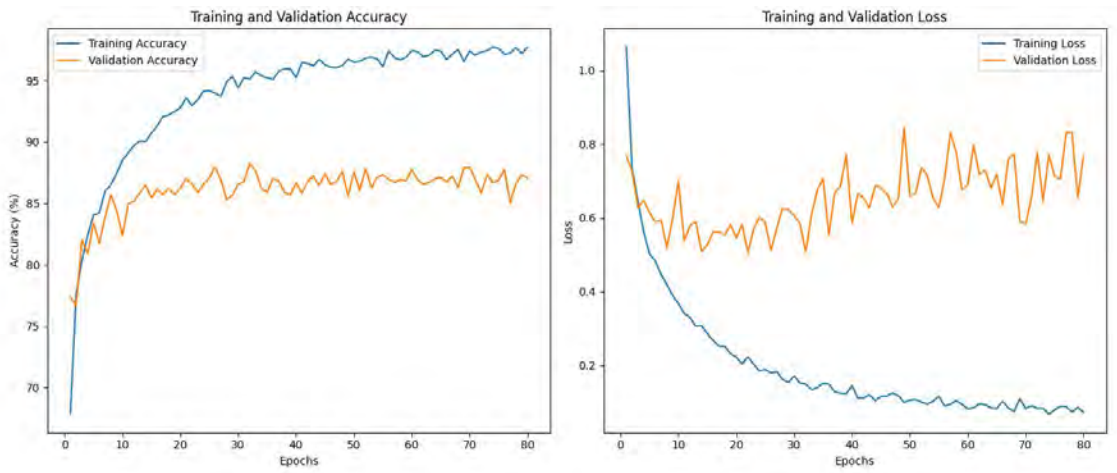

Figure 4.17: Training and Validation accuracy graph of VGG19

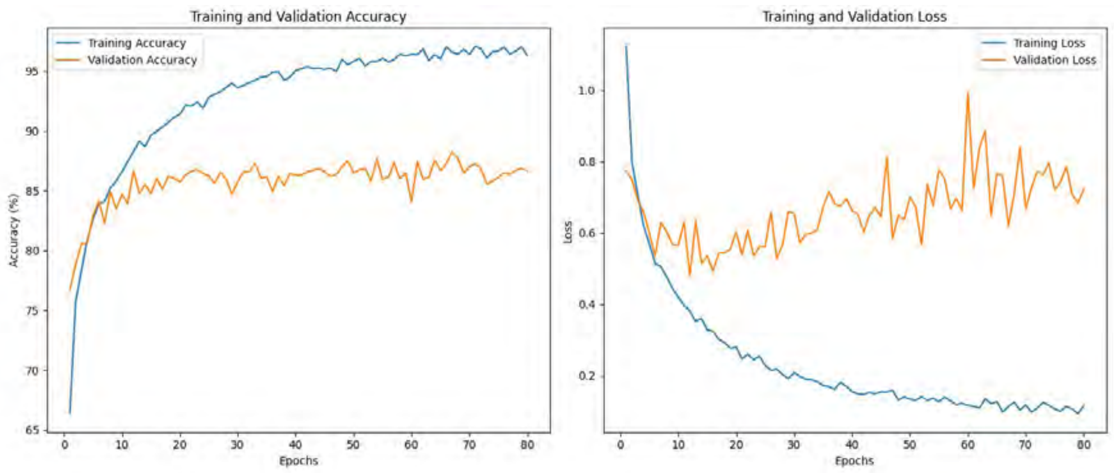

Figure 4.18: Training and Validation accuracy graph of MobileNetV2

4.7 Training and validation accuracy of Citrus class

There are a total of 19 classes in the Citrus super class. There are 4356 images in training and 725 in validation. After training the models with Rice super class. The training accuracy of VGG19-KAN was 95.32% ,VGG19 got 88.96% and CNN got 61.16%. The validation accuracy of VGG19-KAN was 80.83% and VGG19 was 64.97% and CNN was 56.97%. The training accuracy and the validation accuracy of VGG19-KAN is higher than other models.

Table 4.7: Comparison between VGG19, MobileNetV2 and VGG19-KAN of Citrus class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	98.03%	83.17%	0.78	0.76	0.79
VGG19	95.71%	81.66%	0.76	0.75	0.78
Mobilenetv2	95.52%	80.28%	0.73	0.73	0.78

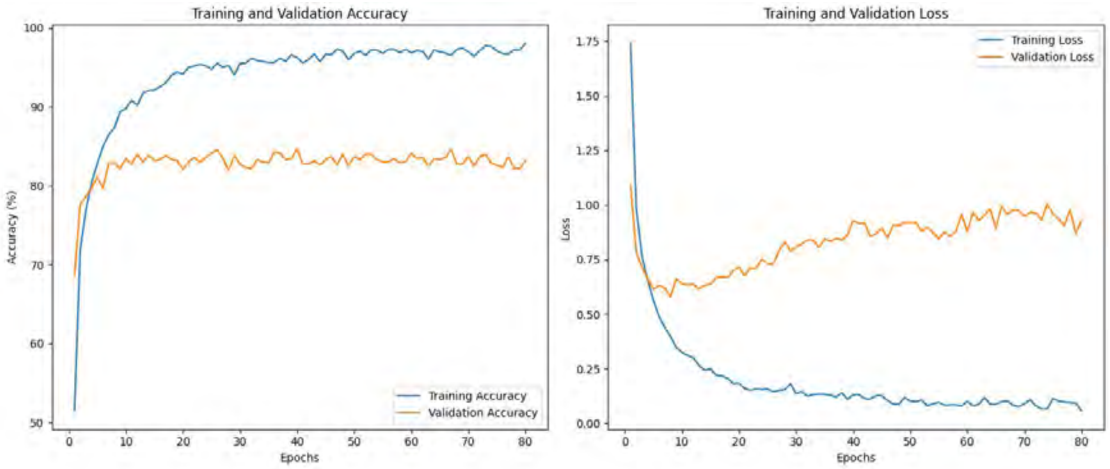

Figure 4.19: Training and Validation accuracy graph of VGG19-KAN

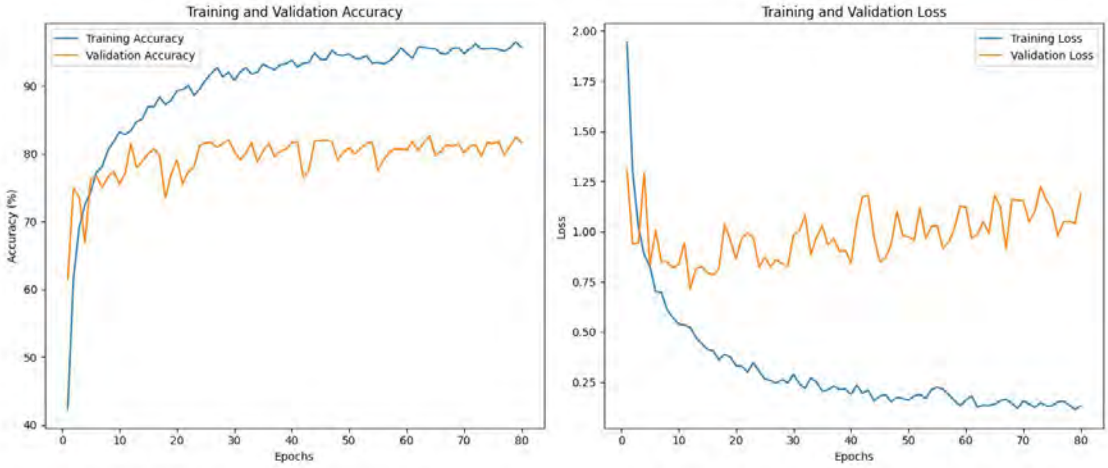

Figure 4.20: Training and Validation accuracy graph of VGG19

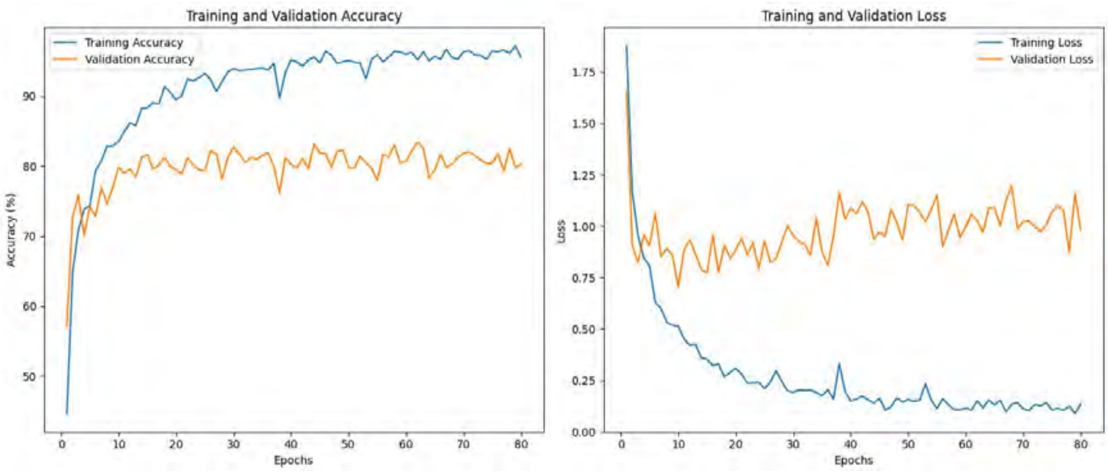

Figure 4.21: Training and Validation accuracy graph of MobileNetV2

4.8 Training and validation accuracy of Mango class

There are a total of 10 classes in the Mango super class. There are 5840 images in training and 971 in validation. After training the models with Mango super class. The training accuracy of VGG19-KAN was 98.07% ,VGG19 got 97.28% and Mobilenetv2 got 98.12%. The validation accuracy of VGG19-KAN was 92.07% and VGG19 was 90.94% and Mobilenetv2 89.70%. The training accuracy and the validation accuracy VGG19-KAN is higher than other models.

Table 4.8: Comparison between VGG19, MobileNetV2 and VGG19-KAN of Mango class

Model	Training Accuracy	Validation Accuracy	F1	Recall	Precision
VGG19-KAN	98.07%	92.07%	0.85	0.82	0.88
VGG19	97.28%	90.94%	0.82	0.80	0.86
Mobilenetv2	98.12%	89.70%	0.82	0.78	0.87

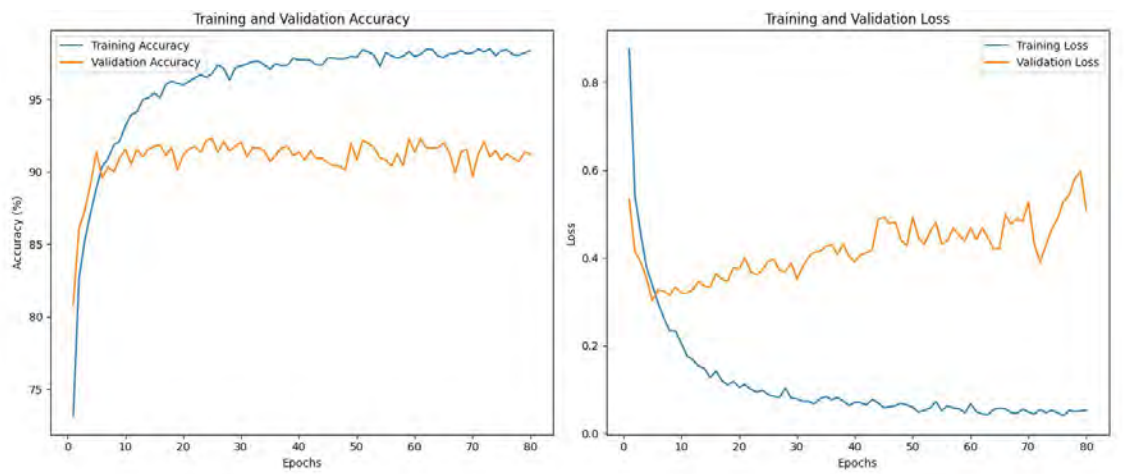

Figure 4.22: Training and Validation accuracy graph of VGG19-KAN

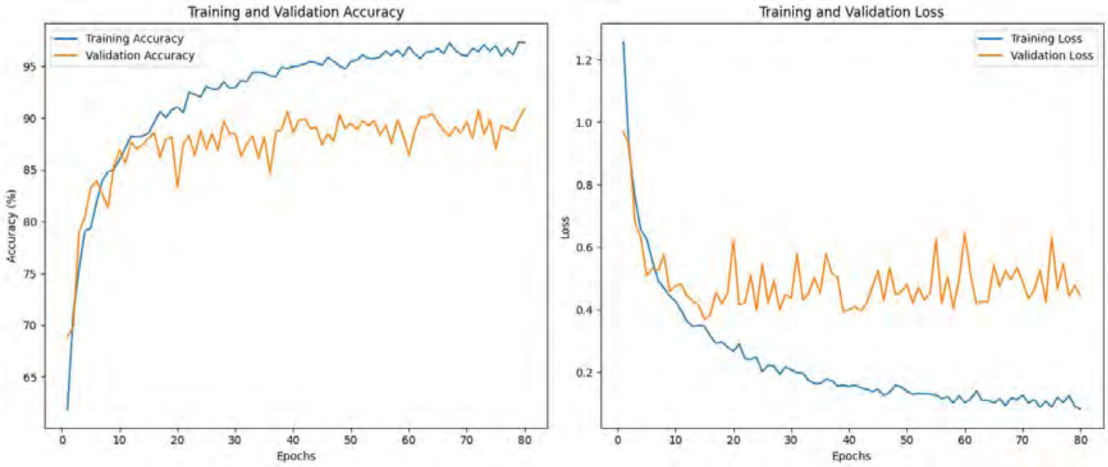

Figure 4.23: Training and Validation accuracy graph of VGG19

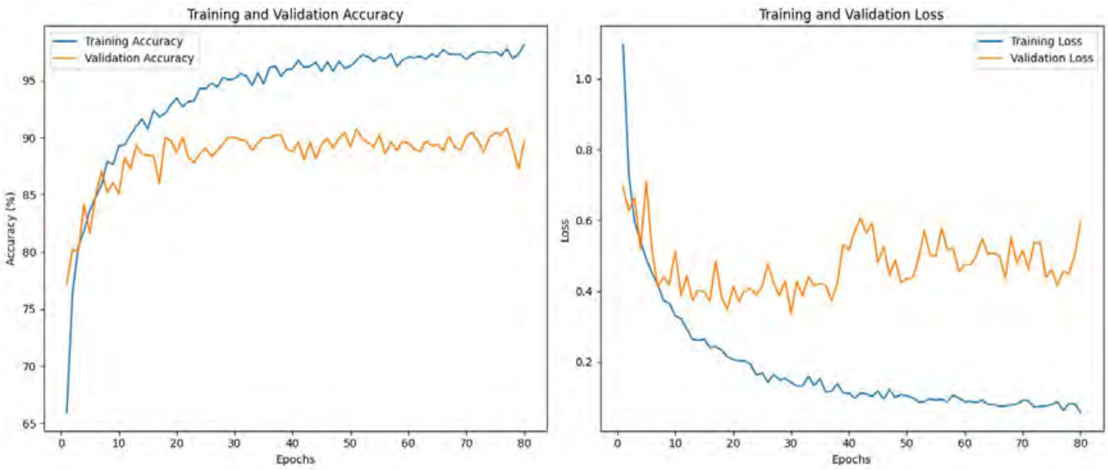

Figure 4.24: Training and Validation accuracy graph of MobileNetV2

4.9 Result analysis

VGG19-KAN, VGG19, and Mobilenet V2 were evaluated by training accuracy, validation accuracy, f1, recall, and precision metrics for each crop class. In Rice class, VGG19-KAN achieved the highest accuracy (92.33%) and validation accuracy (66.63%). F1 (0.65) was similar to the VGG19 but recall and precision were higher than other models. In the corn class, the VGG19-KAN model once again demonstrated superior performance in terms of training accuracy (96.31%) and validation accuracy (83.35%). The f1 (0.74), recall (0.73), and precision (0.76) were also pretty impressive. In wheat class, the validation accuracy of VGG19-KAN and VGG19 was pretty similar but VGG19 was a little bit higher than VGG19-KAN. In beet class, the validation accuracy of VGG19-KAN and VGG19 were also pretty similar but VGG19 was also a little bit higher than VGG19-KAN. In Alfalfa, Vitis, Citrus, and Mango classes, VGG19-KAN consistently outperformed VGG19 and Mobilenet V2, with the highest training accuracy (92.07%) and validation accuracy (97.28%) in the mango class. Also, f1, precision, and recall were higher than other models. After doing analysis between VGG19-KAN, VGG19 and Mobilenet V2 on 8 crop classes, we can conclude that VGG19-KAN performed better than VGG19 and MobilenetV2.

Chapter 5

Conclusion

According to FAO, every year pests cause a lot of damage to crops [13]. Therefore, machine learning techniques are the best option for farmers to reduce their losses. In our paper we are going to use various proposed models which will be going to identify the pest and give its information as an output.We will be using IP102 dataset to train our proposed model and get our required output.The main purpose of this research is to create a new agricultural pest classification model to identify insect pest before it destroying any crops.Thus, saving our agriculture sector from destruction.

5.1 Future work

The research work we are proposing to detect crop pests using a novel model named VGG19-KAN. This model is relatively new and there is less work with this model. Specially, No one didn't implement in the agriculture sector. The future work that we can work one:

- We will try to make the model more efficient so that less powerful devices can run this model.
- We want to explore the KAN network and refine it so that it can perform better in pest classification.
- We will experiment this KAN network with many CNN models like RESNET50, VGG16 to see that how much does it perform against CNN models using Fully connected network.

Bibliography

[1] D. Xia, P. Chen, B. Wang, J. Zhang, and C. Xie, “Insect detection and classification based on an improved convolutional neural network,” *Sensors*, vol. 18, no. 12, p. 4169, 2018.

[2] L. Liu, R. Wang, C. Xie, *et al.*, “Pestnet: An end-to-end deep learning approach for large-scale multi-class pest detection and classification,” *Ieee Access*, vol. 7, pp. 45 301–45 312, 2019.

[3] Y. Nanehkaran, D. Zhang, J. Chen, Y. Tian, and N. Al-Nabhan, “Recognition of plant leaf diseases based on computer vision,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–18, 2020.

[4] J. Chen, W. Chen, A. Zeb, D. Zhang, and Y. A. Nanehkaran, “Crop pest recognition using attention-embedded lightweight network under field conditions,” *Applied entomology and zoology*, pp. 1–16, 2021.

[5] T. Kasinathan and S. R. Uyyala, “Machine learning ensemble with image processing for pest identification and classification in field crops,” *Neural Computing and Applications*, vol. 33, pp. 7491–7504, 2021.

[6] H. Li, S. Li, J. Yu, Y. Han, and A. Dong, “Plant disease and insect pest identification based on vision transformer,” in *International conference on internet of things and machine learning (IoTML 2021)*, SPIE, vol. 12174, 2022, pp. 194–201.

[7] N. Ullah, J. A. Khan, L. A. Alharbi, A. Raza, W. Khan, and I. Ahmad, “An efficient approach for crops pests recognition and classification based on novel deeppestnet deep learning model,” *IEEE Access*, vol. 10, pp. 73 019–73 032, 2022.

[8] M. Xu, S. Yoon, Y. Jeong, J. Lee, and D. S. Park, “Transfer learning with self-supervised vision transformer for large-scale plant identification.,” in *CLEF (Working Notes)*, 2022, pp. 2238–2252.

[9] S. Zhang, R. Jing, and X. Shi, “Crop pest recognition based on a modified capsule network,” *Systems Science & Control Engineering*, vol. 10, no. 1, pp. 552–561, 2022.

[10] S. Zhao, J. Liu, Z. Bai, C. Hu, and Y. Jin, “Crop pest recognition in real agricultural environment using convolutional neural networks by a parallel attention mechanism,” *Frontiers in Plant Science*, vol. 13, p. 839 572, 2022.

[11] M. De Silva and D. Brown, “Plant disease detection using vision transformers on multispectral natural environment images,” in *2023 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD)*, IEEE, 2023, pp. 1–6.

[12] P. Gole, P. Bedi, S. Marwaha, M. A. Haque, and C. K. Deb, "Trincnet: A lightweight vision transformer network for identification of plant diseases," *Frontiers in Plant Science*, vol. 14, p. 1 221 557, 2023.

[13] P. Venkatasaichandrakanth and M. Iyapparaja, "Pest detection and classification in peanut crops using cnn, mfo, and evita algorithms," *IEEE Access*, 2023.

[14] U. Barman, P. Sarma, M. Rahman, *et al.*, "Vit-smartagri: Vision transformer and smartphone-based plant disease detection for smart agriculture," *Agronomy*, vol. 14, no. 2, p. 327, 2024.

[15] Z. Bozorgasl and H. Chen, "Wav-kan: Wavelet kolmogorov-arnold networks," *arXiv preprint arXiv:2405.12832*, 2024.

[16] H. Liu, "Plant disease recognition based on parallel vision transformer," *Applied and Computational Engineering*, vol. 41, pp. 176–181, Feb. 2024. DOI: 10.54254/2755-2721/41/20230739.

[17] Z. Liu, P. Ma, Y. Wang, W. Matusik, and M. Tegmark, "Kan 2.0: Kolmogorov-arnold networks meet science," *arXiv preprint arXiv:2408.10205*, 2024.

[18] Z. Liu, Y. Wang, S. Vaidya, *et al.*, "Kan: Kolmogorov-arnold networks," *arXiv preprint arXiv:2404.19756*, 2024.

[19] P. Venkatasaichandrakanth and M. Iyapparaja, "Gnvit-an enhanced image-based groundnut pest classification using vision transformer (vit) model," *Plos one*, vol. 19, no. 3, e0301174, 2024.

[20] H. Thakkar, A. Pingle, S. Kulkarni, R. Saraf, and R. V. Kulkarni, "Disease and pest detection in crops using computer vision: A comprehensive study,"