UNIVERSITY OF NAIROBI

CONVOLUTIONAL NEURAL NETWORK BASED FALL ARMYWORM DAMAGE DETECTION SYSTEM

BY

COLLINS KARIUKI KARIIRU

THIS RESEARCH PROJECT REPORT WAS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF MASTER OF SCIENCE IN COMPUTATIONAL INTELLIGENCE OF THE UNIVERSITY OF NAIROBI

JULY 2022

DECLARATION

This project report is my original work and has not been presented to any other University for the any award.

SIGNATURE ……… ………………………………………………

DATE: 7th July 2022

Mr. Collins Kariuki Kariiru

P52/38275/2020

THE SUPERVISOR

This project report has been submitted in partial fulfillment of the requirements for the award of the degree of Master of Science in computational intelligence with my approval as the university supervisor.

SIGNATURE DATE: 7th July 2022

Prof Elisha Opiyo.

Department of Computing and Informatics

DEDICATION

This thesis is dedicated to my late father Erastus Kariiru whose passion for science inspired me to pursue this master’s degree.

ACKNOWLEDGEMENTS

I would like to my sincere gratitude to the Almighty God for the gift of life and the opportunity to pursue this degree. His grace has been sufficient all the way. I am very grateful to my supervisor Prof Elisha Opiyo for the guidance and encouragement throughout the project. Many thanks to my family for their immense support along the way. I couldn’t have achieved this without them. I would also want to thank my classmates for their words of encouragement and support. Finally, I thank the University of Nairobi and faculty at large for working around the clock to ensure we received world class education.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

ABBREVIATIONS AND ACRONMYN

FAW - Fall Armyworm

IPM - Integrated Pest Management

FAO - Food and Agriculture Organization

KARLO - Kenya Agricultural & Livestock Research Organization

CABI - Centre for Agriculture and Bioscience International

CV - Computer Vision

CNN - Convolutional Neural Network

YOLO - You Only Look Once

mAP - Mean Average Precision

ABSTRACT

Fall armyworm (Spodoptera frugiperda) is an invasive pest that attacks a wide range of plants (Early et al., 2018) It is especially notorious for attacking one of Africa's most important foods: maize which is a source of livelihood and a staple food for millions of people across the continent. (Day et al., 2017). Current approaches used in fall armyworm monitoring require physical presence of an agricultural expert (agricultural extension officer or plant entomologist) to guide farmers in the identification of fall armyworm damage on maize leaves. Without expert training, farmers could easily confuse FAW attacks with other common maize pests leading to delayed or incorrect intervention measures and can lead to the loss of an entire crop. Meissle et al. (2010). In the recent past, machine learning techniques have been applied in pest detection. (Ebrahimi et al., 2017; Voulodimos et al., 2018). Despite the potential benefits offered by current machine learning approaches in literature, there lacks a CNN based mobile artifact that offers an easy-to-use alternative to classify and localize fall armyworm damage on maize leaves in the natural farm environment. This research compares the performance of two one stage convolutional neural network meta-architectures to develop a FAW damage detection mobile application. Experimental results show impressive performance, with the best performing efficientdet lite model achieving a mean average precision of 85.85% and the best performing yolov4 tiny model achieving a mean average precision of 82.5%.

CHAPTER ONE: INTRODUCTION

1.1 BACKGROUND OF THE STUDY

Fall Armyworm (Spodoptera frugiperda) is an invasive pest that attacks over 350 plant species. It is native to the tropical and sub-tropical regions of the Americas. (Early et al., 2018) but has over the last five years spread to the African continent (initial reports in 2016) and Asia (initial reports in 2018). The spread and devastating effects of the fall armyworm attacks have been felt especially in Africa since it attacks maize plants considered a major staple food and source of livelihood for millions of farmers in the continent (Day et al., 2017). The food and agriculture organization notes that farmers loose between 20 - 40 % of their yields to pests and diseases threating the state of food security FAO. (2020).

Currently FAW monitoring and detection is done through manual based monitoring (farm scouting) and trap-based monitoring (pheromone traps). Like many other manual pest monitoring techniques, these approaches are subjective, delayed and hard to implement at scale.(Dai et al., 2016; Selvaraj et al., 2019; Thenmozhi & Reddy, 2019) (He et al., 2019). Lack of prompt action in case of a FAW invasion can lead to loss of an entire crop yield (Kassie et al., 2020). Researchers have studied different ways computer vision techniques can be applied in the agricultural discipline (Ghadge et al., n.d.; Paul et al., 2020; Tian et al., 2020) Classical image processing and deep learning techniques have been proposed to detect anomalies in plants.(Ensari et al., 2020; Jayswal & Chaudhari, 2020; J. Liu & Wang, 2021; Patil et al., 2020; Rustia et al., 2021; Syarief & Setiawan, 2020). The adoption of deep learning approaches in object classification and detection has greatly improved the computer vision tasks in comparison to traditional image processing techniques (Voulodimos et al., 2018). Existing models lack generalizability when tested in natural environments. This is mainly because images used to train the said models are taken on plain backgrounds in a laboratory setting. (Selvaraj et al., 2019).

The researcher seeks to develop a convolutional neural network-based solution that provides timely and accurate FAW damage detection on maize leaves. The proposed solution will leverage the power of deep neural network and deep transfer learning (C. Tan et al., 2018) to train and deploy CNN models that unique distinguish fall armyworms damage on maize leaves from infestation by other pests. This research project will compare two algorithms based on one-stage CNN meta-architectures (Huang et al., 2017). The better performing model will be deployed on a mobile application for use in the farm. The resulting artifact will provide a vital integrated pest management tool for stakeholders in the maize value chain including farmers, governmental and non-organizational organizations taking keen interest in integrated pest management strategies against the fall armyworm.

1.2 PROBLEM STATEMENT

Pest and diseases negatively affect crop growth process and the resulting yield that is harvested (Cerda et al., 2017). Current approaches used in FAW monitoring require physical presence of an agricultural extension officer or plant pathologist (farm scouting and inspecting pheromone traps). In addition to that farmer use some empirical knowledge they have acquired along the way to identify presence of FAW in their farms. Though effective in some cases, more often than not, it is subjective and prone to errors (Thenmozhi & Reddy, 2019). Without expert training, farmers confuse FAW attacks with other common maize pests such as Cotton bollworm (Helicoverpa armigera) and Southern armyworm (Spodoptera eridania). Untimely and Incorrect identification of a pest leads to delayed corrective measures and can lead to loss of an entire crop (Meissle et al., 2010).

Previous literature relating to pest identification in computer vision has concentrated more on pest classification and estimating population on pheromone traps. (Thenmozhi & Reddy, 2019) (Wang et al., 2015) proposed a CNN based system to classify 40 classes of insect species found in the Xie1, Xie2 and National Bureau of Agricultural Insect Resources (NBAIR) datasets. (Chiwamba et al., 2018) proposed the use of CNN system in embedded Raspberry Pi in counting the number of FAW moths on a pheromone trap so as to estimate its population. Other techniques employed in pest identification are based on combining traditional image processing techniques and classical machine learning such as histogram of oriented gradient (HOG) or Scale invariant feature transform and Support Vector Machines. While these methods show impressive results in pest classification(Fuentes et al., 2017) they cannot be generalized in identification of other pests. The researcher seeks to extend the current body of knowledge on the application of CNN based algorithms in pest detection. In this case the application will target FAW damage on maize leaves.

The researcher seeks to investigate the application of one stage convolutional neural networks in monitoring FAW attacks. This by extension will improve the existing FAW monitoring techniques.

1.3 THE OBJECTIVES

1.3.1. GENERAL OBJECTIVE

Develop a mobile based fall armyworm (FAW) damage detection system using one stage convolutional neural network meta-architectures.

1.3.2. SPECIFIC OBJECTIVES

1. Collect field data of images of the damage done by fall armyworm.
2. Compare the performance of YOLO v4 tiny and EfficientDet lite CNN meta - architectures.
3. Develop an android mobile application that detects FAW damage on maize leaves using one stage convolutional neural network meta-architectures.

1.4 RESEARCH QUESTIONS

1. Can YOLOv4 tiny or EfficientDet lite CNN algorithms be used to develop an accurate FAW damage detection model?
2. How does the performance of YOLOv4 tiny and EfficientDet lite CNN meta-architectures compare in detecting FAW damage on maize leaves?
3. Can the aforementioned model be integrated into a mobile application for FAW damage detection?

1.4 SIGNIFICANCE OF THE STUDY

The research proposes an accurate and timely solution for identifying FAW invasion on maize leaves. FAW is a serious threat to an already weak food security situation on Africa. Since it was first reported in the continent in 2016, yearly maize losses due to FAW attacks are estimated at 9.4% - 66% (Baudron et al., 2019; Day et al., 2017; Kumela et al., 2019). By using the proposed system in FAW damage detection, different stakeholders will benefit. First the society at large will enjoy increased food security and improve environmental conservation. Early detection of FAW helps farmers take corrective actions to protects their plants from further destruction (Fuentes et al., 2017). Early detection also ensures that correct amount and type of pesticides are used, saving the environment from hazardous effects due to excessive use of pesticides (He et al., 2019). The research also adds on the efforts of non-governmental organizations such as Food and Agriculture Organization and Centre for Agriculture and Bioscience International researching on integrated pest management solutions for dealing with the fall armyworm and other pests (Khatri et al., 2020). The research also provides a robust and accurate tool for farmers that helps them reduce economic losses by detecting FAW invasions early and taking the necessary steps in safeguarding their crops. Finally, the research benefits computer vision researchers interested in deep neural network methodologies in pest detection.

1.5 RESEARCH CONTRIBUTION

The research contribution is the proposed approach of using one stage CNN meta-architecture mainly focused on mobile devices. The application provides an end to end platform that takes images as an input, performs object detection on the images and output an image with bounding boxes and the confidence scores highlighting areas attacked by FAW. The resulting artifact can be installed on mobile devices and used in the field. This provides a step forward in incorporating vision-based application in integrated pest management. The artifact works on devices with diverse camera resolutions and in complex environments such as as varying backgrounds, natural lighting, camera orientation, illumination etc.

1.6 SCOPE, ASSUMPTIONS AND LIMITATIONS OF THE STUDY

The study's scope is limited to detecting FAW attack on maize leaves. Other crops attacked by the FAW pest are not considered. The images used to train and evaluate the models were taken in Kirinyaga county, Kenya so the application might not generalize well in areas with different environmental conditions. The study will use one stage convolutional neural network-based approach to detect presence of fall armyworm on maize leaf images taken in the natural environment and output an image with bounding box and confidence score around regions where fall armyworm damage is identified. Future versions and configurations of the meta-architectures might show varying results from those presented in this paper.

The researcher utilizes commercial and open-source software packages and libraries in performing this study and is therefore limited to the capabilities of said tools. The researcher will utilize more than one software package/library when deemed necessary to overcome this limitation.

1.7 ORGANIZATION OF THE RESEARCH THESIS

This thesis includes five chapters. The first chapter gives an introduction of the research, the problem statement, research objectives and the research contributions. The second chapter gives relevant literature review related to the FAW, computer vision techniques and algorithms the researcher seeks to explore. Related works in this research will also be highlighted. The third chapter covers the research methodology used to train the models and the approach followed in developing the mobile application. The fourth chapter presents the results and discussion after training and deploying the CNN models on the mobile application. Chapter five covers the Conclusion and recommendations for future research.

CHAPTER TWO: LITERATURE REVIEW

2.1 MAIZE FARMING IN KENYA

Maize remains one of the most important sources of food in Kenya. It doubles up as a staple food (65% of staple food calories) (Mohajan, 2014) and a source of income to thousands of small- and large-scale farmers cultivating it around the country. Research conducted by (Edoh Ognakossan et al., 2016) highlights that the maize growing regions in Kenya are subdivided into six main agroecological zones i.e., the lowland tropical (LT) zone, dry mid-altitude (DM) zone, highland tropics (HT) zone, moist transitional (MT) zone, moist mid-altitude (MM) zone and finally the dry transitional (DT) zone. It is worth noting that the maize yield / area of cultivation is highly skewed between the six zones. The moist transitional (MT) and highland tropics (HT) zones produce the highest yield (2.5 tons per ha) accounting for 50% of total maize yield countrywide followed by the moist mid-altitude (MM) zone producing (1.5 tons per ha), accounting for 20% of total maize yield and finally low- land tropics (LT), dry mid-altitude (DM) and dry transitional (DT) zones (1 ton per ha) accounting for the remaining 30%.

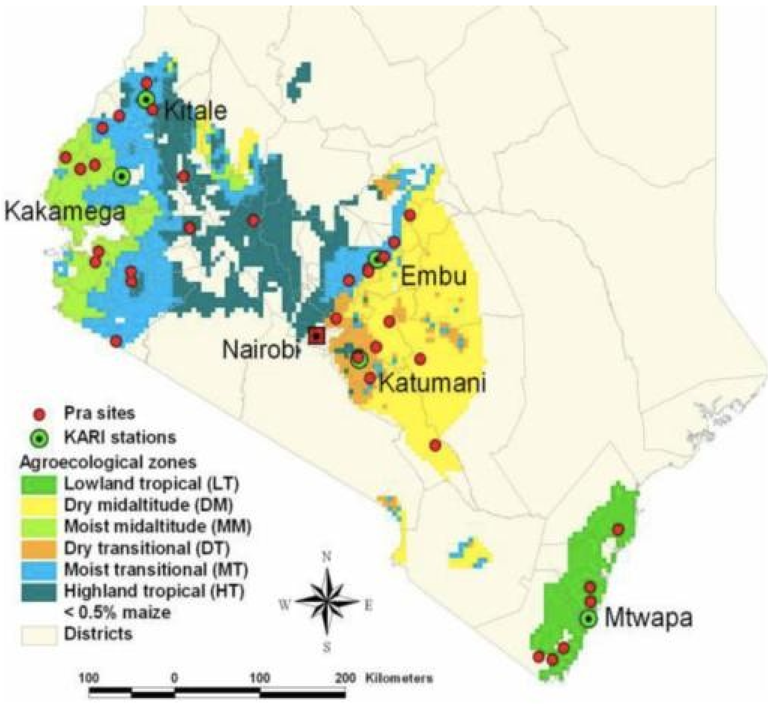


Figure 1: Agroecological Zones of Kenya

SOURCE: (Groote et al., 2011)

2.2 COMMON MAIZE PESTS IN KENYA

According to KARLO, the most common maize pests in Kenya are African Maize Stalk borers (Busseola fusca), Maize Leafhoppers (Cicadulina spp), Maize Aphids (Rhopalosiphum maidis), Bollworms (Helicoverpa armigera) and Cutworms (Noctuidae). In this section we will discuss the pests and the common signs of their attacks.

- **African Maize Stalk borers (Busseola fusca) –** They are common and destructive insect pests that attack maize plants between 3 and 5 weeks old. The moth lays eggs on the leaves of the youngest unfolded leaves which mature into caterpillars that spread to nearby crops.
- **Maize Leafhoppers (Cicadulina spp) –** This is another common maize pest that hops from one plant to another causing spread of the maize streak virus. Maize leafhoppers have two distinct black spots between their eyes and are slender in size. This pest is especially dangerous since it can lead to total yield loss unless intervention measures are taken in a timely manner.
- **Maize Aphids (Rhopalosiphum maidis) –** Aphids attacks are characterized by sooty mold on the leaves of young maize plants. The pests soft bodied with two long antenna and are bluish-green in color.
- **Bollworms (Helicoverpa armigera) –** This pest feeds on a variety of plant parts including the leaves, flower and fruit. Damage on maize leaves leads to stunted growth.
- **Cutworms (Noctuidae) –** Cutworms are caterpillars that damage the maize seedlings before germinating above the ground level

a) African Maize Stalk borers **b) Maize Leafhoppers**

Figure 2:African Maize Stalk borers & Maize Leafhoppers

c) Maize Aphids d) Bollworms

Figure 3:Maize Aphids & Bollworms

e) Cutworms

Figure 4:Cutworms

2.3. OVERVIEW OF FAW IN KENYA.

Initial reports of FAW in Africa were in Central and West Africa in 2016, later spreading to sub-Saharan Africa in the same year (Day et al., 2017). FAW infestation spread quickly within the continent due to its ability to lay large number of eggs (1500 per female moth), travel over long distances (100 km per night) and its preference for maize plants which are widely cultivated. Based on a survey conducted in Ghana and Zambia by (Day et al., 2017), Fall Armyworm could cause a yield loss of between 8.3 and 20.6 million tons of maize yearly valued at between $ 2.4 billion and $ 6.2 billion.(Kumela et al., 2019). The estimated the maize yield losses caused by FAW in Kenya in 2018 stood at about 47%. The Food and Agriculture Organization defines pest yield loss as a percentage difference between attainable and actual yield because of pest attacks.(Oerke, 2006). Crop yield losses can be computed using direct methods, indirect methods, expert opinions, farmer estimates and through community surveys (De Groote et al., 2020). A survey conducted by (De Groote et al., 2020) shows that in 2017 and 2018 maize yield losses amounted to 37% and 33% of total yield equivalent to 1 million tons of maize in each of the two years. The destructive nature of FAW has led to increased interest and support by the government of Kenya through the Technical Cooperation Project (TCP), Food and Agriculture Organization and the International Maize and Wheat Improvement Centre. (Padhee & Prasanna, 2019).

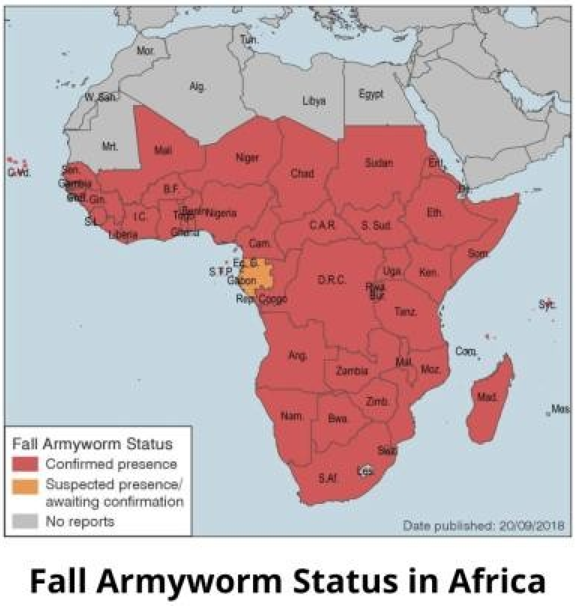


Figure 5:Fall Armyworm Status in Africa

Source: Rwamushana (2018)

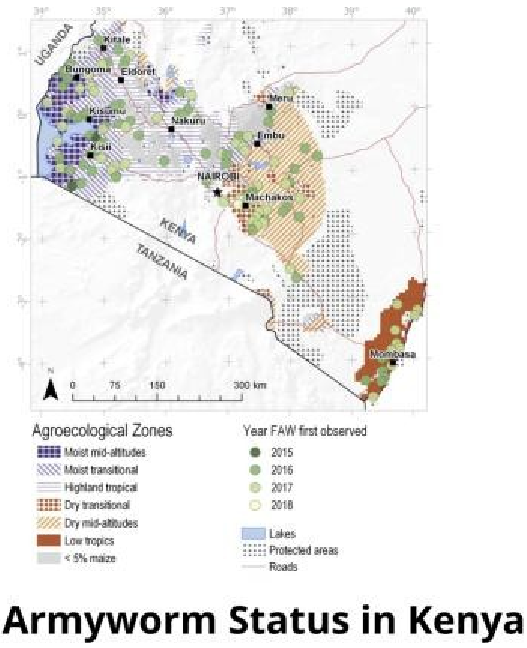


Figure 6:Fall Armyworm Status in Kenya

Source: (De Groote et al., 2020)

2.4. FAW LIFE CYCLE

Understanding the FAW life cycle is the basis for developing an effective Integrated Pest Management System against this invasive pest. (Padhee & Prasanna, 2019) clearly documents the FAW life cycle which takes between 30 and 90 days depending on the weather conditions. One of the reasons that makes FAW dangerous to the wide variety of crops it attacks is its high reproduction rate; estimated at 1500 eggs per female moth (Padhee & Prasanna, 2019). In addition, FAW insects can travel over long distances; about 500 kilometers during its lifetime. The FAW pest goes through four main stages namely eggs, caterpillar/ larvae, pupae, and adult (moth). The first stage is the egg stage which lasts between 2 and 3 days. The larval stage takes between 14 and 30 days over six instars. FAW are most destructive during the 3rd and 4th instars of the larval stage causing extensive defoliation. The pupae stage lasts between 8 and 30 days depending on the weather conditions. The final phase is the adult stage that lasts approximately 10 days.

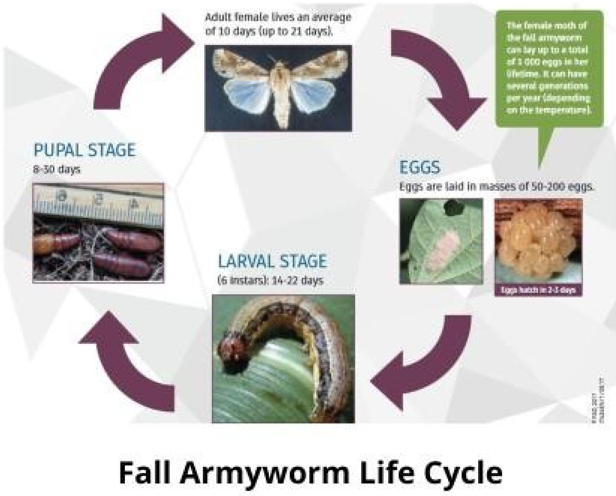


Figure 7:Fall Armyworm Life Cycle

Source: FAO

2.5. FALL ARMYWORM MONITORING TECHNIQUES.

There are two main techniques currently used to assess whether a farm is infested with fall armyworm. These techniques are manual monitoring/ scouting and trap-based monitoring/ pheromone traps.

2.5.1 MANUAL FAW MONITORING

Manual technique requires the farmer or extension officer to physically scout the farm. Scouting process is done by randomly selecting five points in the farm examining ten maize plants at each point, inspecting two or three emerging leaves from the funnel for signs of FAW eggs, small caterpillars, minor pane leaf damage or frass, taking count of the number of plants in each batch of ten with the FAW attack symptoms mentioned and recording the information. The information is encoded using 0 and 1 where 1 representing FAW infected plants and 0 otherwise.

2.5.2 TRAP BASED FAW MONITORING

The trap-based approach uses pheromone trap that attracts male FAW pests. The procedure of setting up the trap include:

- Hang the trap at the edge of the field ensuring that it is about 30 cm above the tallest maize plants to avoid blockage of its entrance by plant leaves.
- Place the pheromone lure (5 per trap) in the basket compartment on top of the trap and the insecticidal strip (10 per trap) that kills trapped insects.

- Replace the strips monthly and pheromone lure every two months.

Figure 8: Fall Armyworm Pheromone Traps

2.6 COMPUTER VISION IN AGRICULTURE

Computer vision is an interdisciplinary scientific field that seeks to give computers the ability to visually perceive the real world and gain high level understanding or digital images and videos (Sonka et al., 2014). There are various ways computer vision techniques have been implemented in the agricultural field. (Patrício & Rieder, 2018) explores how computer vision and advanced artificial intelligence techniques are combined to form robust precision agriculture methods in disease detection, grain quality checking and phenotyping in maize, rice, wheat, soybeans, and barley plants. (Arakeri, 2016) proposes an automated tomato fruit grading system based on computer vision techniques. The proposed system achieved an accuracy of 96.47% on the task. (Chiu et al., 2020) proposes the use of computer vision based aerial drones to take high quality images consisting of both RGB and Near- infrared channels used in plant anomaly detection and semantic segmentation of agricultural patterns. (Zhao et al., 2020) proposes a ground level mapping and navigation system based on computer vision algorithm - (Mesh Simultaneous Localization and Mapping algorithm, Mesh-SLAM) and Internet of Things (IoT), to generate a 3D farm map. This will be implemented as a multi-agent system consisting of ground level robots taking the images, edge node system coordinating the robots and cloud-based system for general management and deep computing

2.7 ONE & TWO STAGE CNN META-ARCHITECTURES

Convolutional neural networks are deep neural network that performs convolution operations on images to extract complex patterns that uniquely identify objects in the images. In addition to the convolution layers, CNN also have pooling and non-linearity layers (Yu et al., 2014). Deep learning models can be trained to perform classification, detection, or segmentation tasks (Howard et al., 2018; Tuggener et al., 2018). Convolutional neural networks that perform object detection can be further subdivided into one stage and two stage architectures (J. Liu & Wang, 2021). Both CNN meta- architectures are used to classify and localize regions of interest in the image. Examples of one stage detectors are You Only Look Once (YOLO) algorithm (J.-W. Chen et al., 2021) Single Shot Multi-Box Detector (SSD) (W. Liu et al., 2016) and EfficientDet (M. Tan et al., 2020) while those under the two stage meta-architecture are Faster Region Based Convolutional Neural Networks (Faster RCNN) (X. Chen & Gupta, 2017), Region-based Fully Convolutional Network (R-FCN) (Dai et al., 2016).

2.7.1. ONE STAGE META-ARCHITECTURE

One stage CNN meta-architecture algorithms combine the region proposal and bounding box regression phases into one stage approached as a simple regression process. The detectors assign class probabilities and bounding box coordinates together hence reducing the computational complexity and the resulting detection time (M. Tan et al., 2020). One stage detector can be used for mobile and embedded systems devices with low computational capacity. (Qin et al., 2019).

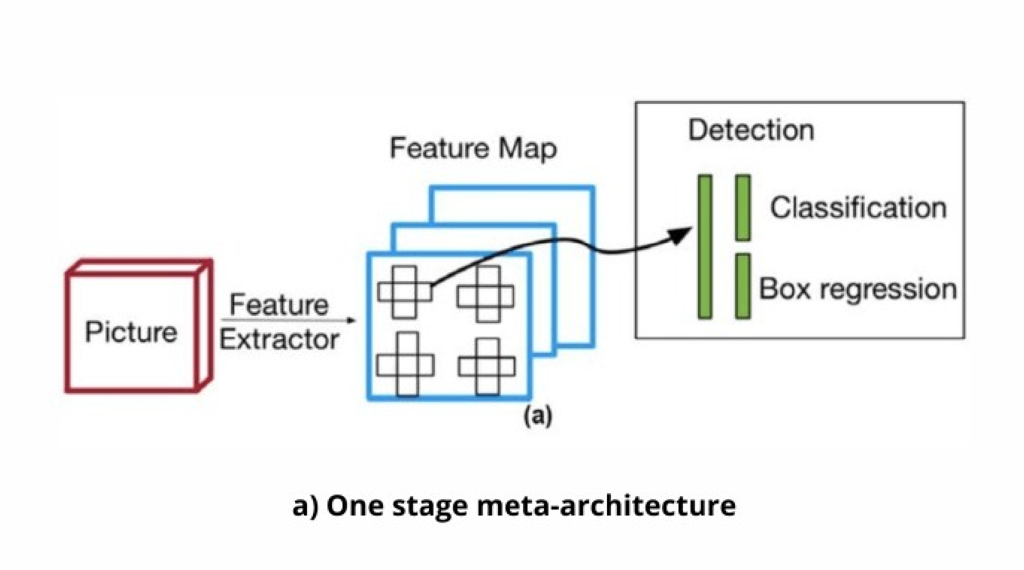


Figure 9: One Stage Meta-architecture

Source: (Kemajou et al., 2019)

2.7.2. TWO STAGE META-ARCHITECTURE

The first phase of the two stage CNN meta-architectures comprises of the Region Proposal Network which uses features from the backbone / feature extractor to identify regions of interest in the image. Regions of interest are identified by comparing the intersection over union between object proposals and annotated ground truth. Object proposals that achieve a score equal or higher than the predetermined threshold is then considered for the second phase. In the second phase of the two-stage detection network, the regions of interest (ROI) previously identified are classified and corresponding bounding boxes assigned. Two stage detectors have higher accuracy than one stage detectors but take longer due to the computational complexity associated with each stage. (Soviany & Ionescu, 2018)

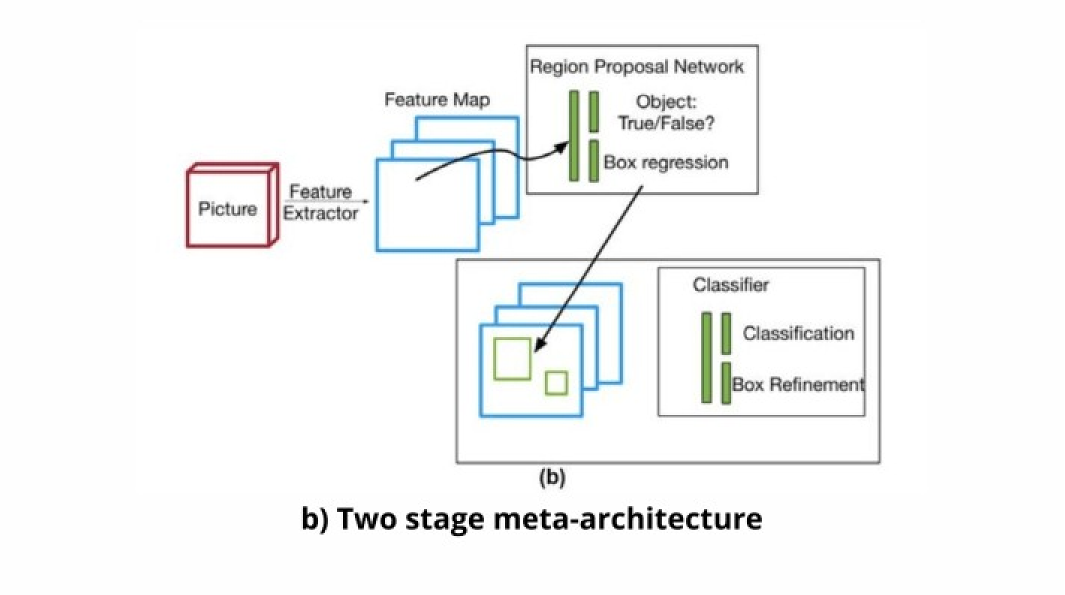


Figure 10: Two Stage Meta-architecture

Source: (Kemajou et al., 2019)

This research explored the performance of two algorithms, from the one-stage meta-architecture. Since the model will be deployed on a mobile phone, the researcher chose YOLOv4 tiny and EfficientDet lite which are optimized for this task. (Nguyen et al., 2020). The two algorithms are discussed below.

2.7.3 YOLOv4 TINY AND EFFICIENTDET LITE CNN ALGORITHMS

2.7.3.1. EFFICIENTDET LITE

EfficientDet lite are mobile friendly object detection algorithms based one stage CNN meta-architecture. Efficentdet framework has been optimized to minimize this limitation. EfficientDet uses BiFPN to optimize the backbone. In addition, it utilizes compound scaling techniques to uniformly scale the width, depth, and resolution its three main components i.e., the backbone, feature networks and classification/ bounding box prediction network concurrently. (Nguyen et al., 2020)

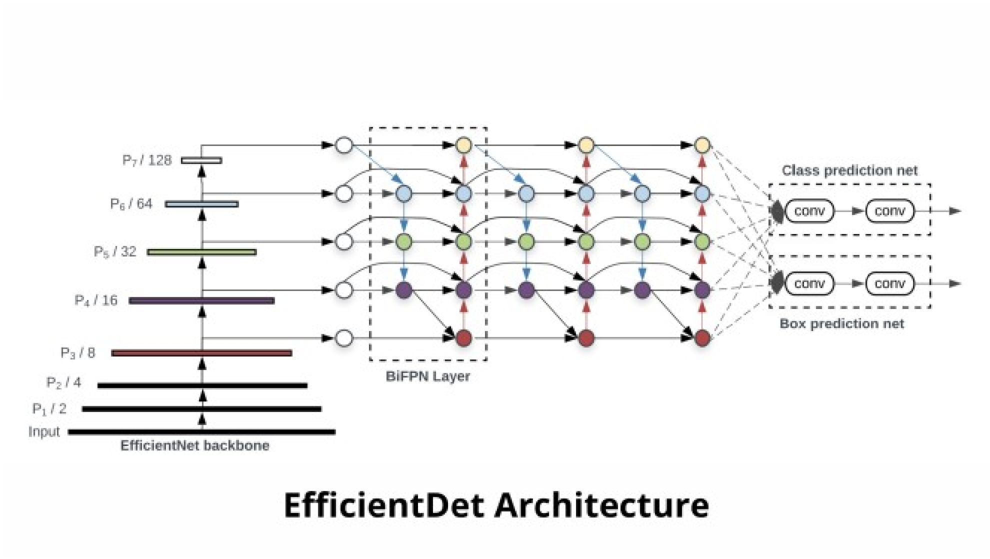


Figure 11: EfficientDet Architecture

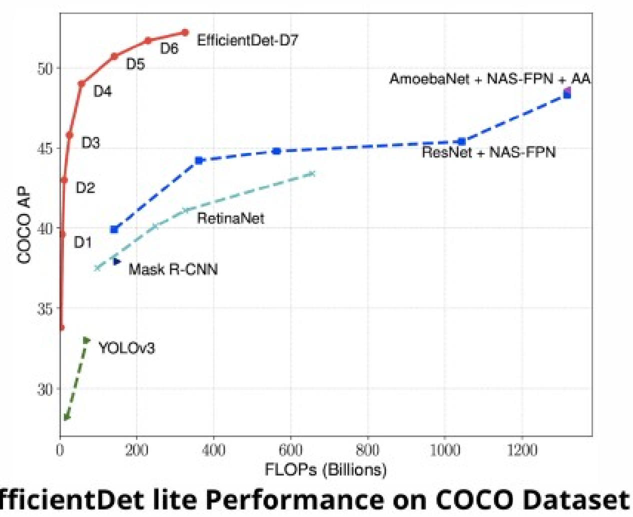


Figure 12: EfficientDet Performance on COCO Dataset

2.7.3.2. YOLOv4 TINY

YOLOv4 tiny, which is a compressed version of YOLOv4, is a state of the art one-stage object detection algorithm. Since this research aims at coming up with a model that is optimized for mobile devices, YOLOv4 tiny was a perfect candidate. With fewer parameters in its architecture, the training and inference time is greatly reduced, an advantage for mobile and edge devices which have limited computational power capabilities. It consists of two YOLO heads and 29 pretrained convolutional layers. Although its accuracy is less than its parent algorithm (YOLOv4), the inference time is better. The average precision of the model is 22%, 40.2% AP50 as compared to 43.5%, 65.7% AP50 in YOLOv4. (Jiang et al., 2020)

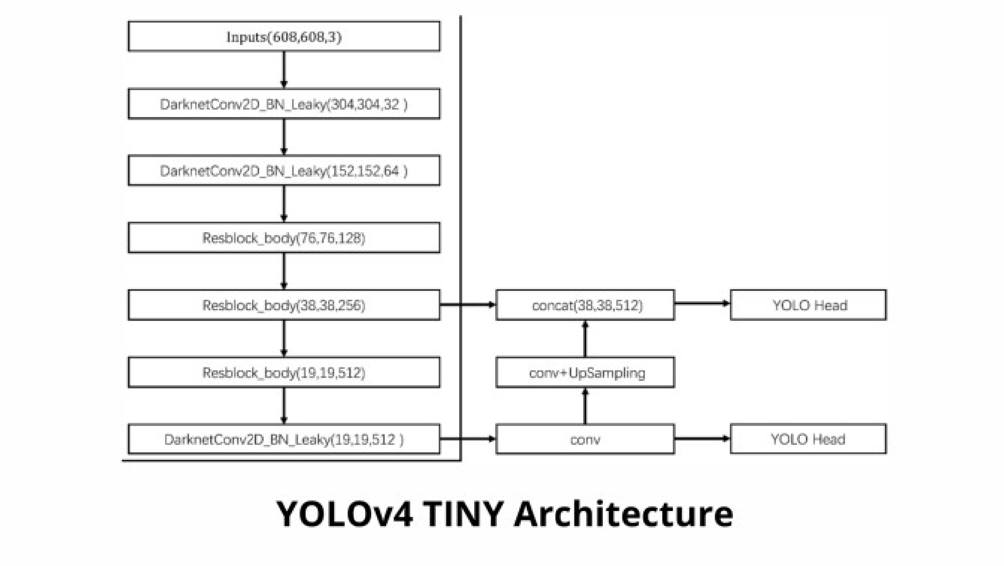


Figure 13: YOLOv4 Architecture

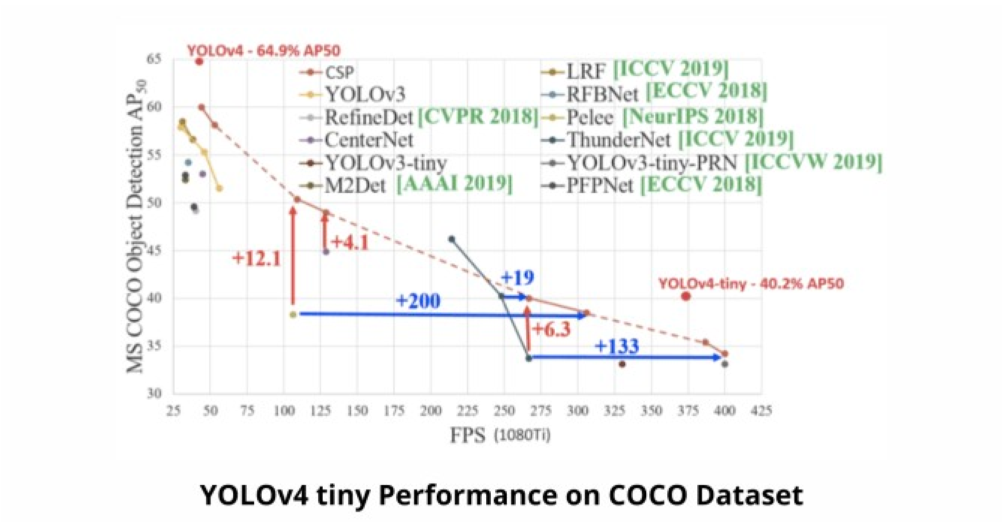


Figure 14: Yolov4 tiny Performance on COCO Dataset

2.8 RELATED WORK

Researchers in the field of computer vision have attempted to develop machine learning based pest detection systems for different pests and following diverse approaches. This section discusses some of the previous research done on the subject.

(Ebrahimi et al., 2017) proposed the use of support vector machines with different kernel functions to detect and classify thrips pests in strawberry plants as an alternative solution for traditional manual insect identification methods. The proposed system would be used for real time pest management in greenhouses. The pest identification process would be conducted by an agricultural robot with a mounted camera. The images would be sent to a web server where the model developed using support vector machine would conduct the predictions. The research yielded impressive results with the best detector having a percentage error of less than 2.25%. This approach however required the researchers to design the SVM based on hand crafted features such as Hue, Saturation, and color indexes. The approach is less reliable and adaptable when generalizing it to detection of another pest using machine vision Voulodimos et al. (2018)

(Ateya, 2018) proposed an internet of things approach for detecting FAW pupa in the soil. The research used a combination of DH11 sensors to collect temperature and humidity parameters and machine learning model to predict presence of FAW pupa in the soil. The model used forward propagation artificial neural networks to combine data from the sensors and feedback from fall armyworm to monitor and alert uses of possible FAW pupa and initiate FAW corrective measures. The system achieved an accuracy of 82.06%. Though the researcher used artificial intelligence approach in monitoring the same pest discussed in this research, the research proposed does not follow a vision-based approach in the prediction of FAW in maize.

(Rustia et al., 2021) proposed the use of convolutional neural networks in classification and detection of 4 pests on pheromone traps in different green houses. The proposed approach aimed at solving the tedious task of estimating pest population by manually counting the number of pests on a pheromone trap. To achieve this, the research proposed the use of a wireless imaging device on an embedded system to count the number of pests of interest on the sticky traps. It is worth noting that the research used separate networks for classification and detection arguing that doing so would ease incorporation of additional pests in the future. The research achieved an F1 score of 92%. Though this research adopted a convolutional neural network approach on real natural environment, it did not detect or classify pests on plant leaves in near real time. (Sun et al., 2018) also adopted a deep learning approach for detecting and classifying pests on pheromone traps.

2.8.1 SUMMARY OF GAPS IN LITERATURE

Related work	Year	Author	Findings	Gaps
Vision-based pest detection based on SVM classification method	2017	(Ebrahimi et al., 2017)	-SVM based image processing used to detect thrips in strawberry plants - Dataset collected in natural environment - Images used in pest classification collected using an automated agricultural robot	- Uses Hand crafted features making it hard to generalize
Fall army-worm prediction model on the maize crop in Kenya: an internet of things based approach	2018	(Ateya, 2018)	- FAW detection at pupa stage -FAW model prediction results send on SMS and system dashboard	- ANN predicts presence of FAW based on IOT sensors rather than vision-based approach
Automatic greenhouse insect pest detection and recognition based on a cascaded deep learning classification method.	2021	(Rustia et al., 2021)	- CNN based approach in detection insects on pheromone traps -Sticky traps have a standardized color which provides a uniform background	- Static background and camera on embedded wireless imaging device.

2.9 CONCEPTUAL MODEL

Figure 14 contains the main components of the conceptual model. The device camera will be used to take pictures of the maize leaf. The image will be analyzed to determine if it contains any signs of FAW. The image is then input into the CNN model where image preprocessing and model inference occurs. In case the leaf has manifestations of fall armyworm attack the model outputs the image, bounding boxes around the infested regions as well as confidence scores for each object detection. If not, the CNN model returns an image with on bounding boxes or labels. The feedback generator generates the view that will be displayed on the mobile application user interface.

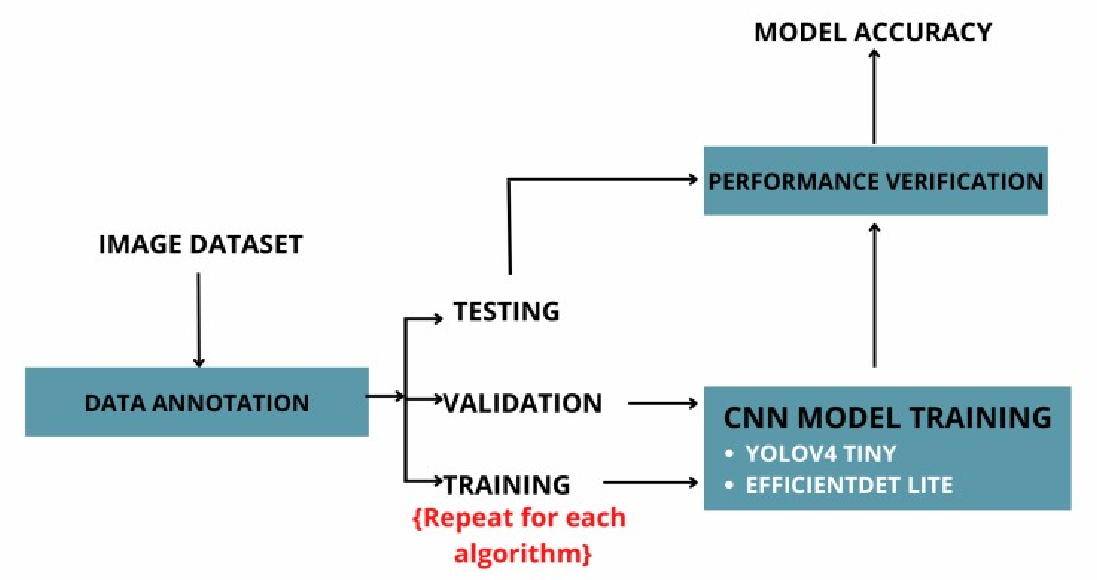


Figure 15: CNN Model Training Conceptual Model

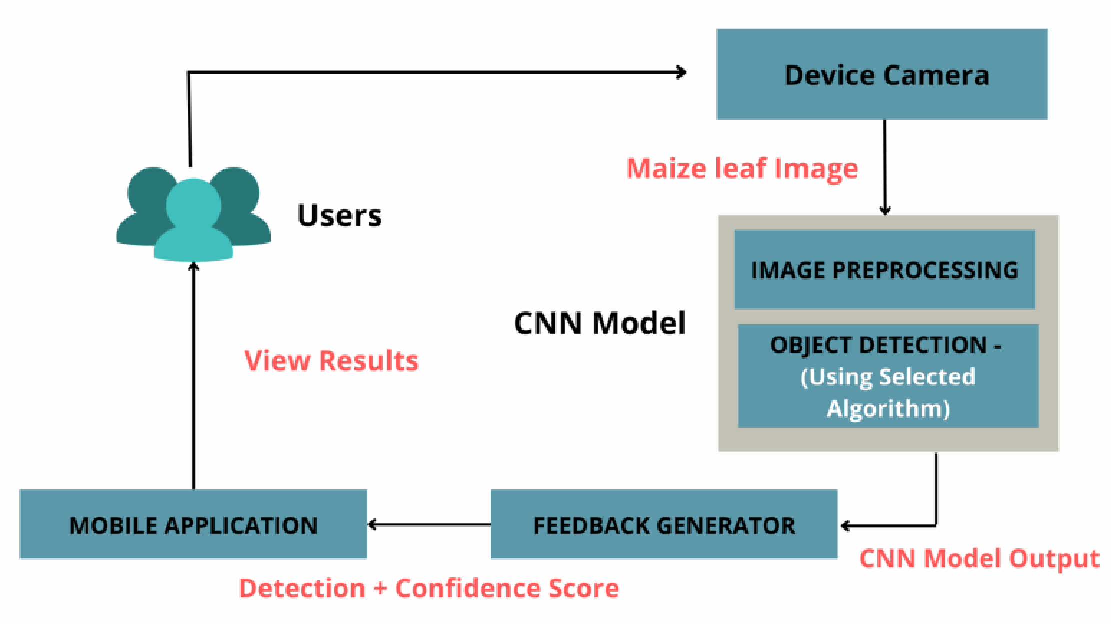


Figure 16: CNN System Conceptual Model (with selected algorithm)

CHAPTER THREE: RESEARCH METHODOLOGY

3.1 INTRODUCTION.

This chapter highlights the research methodology employed in developing a CNN-based FAW damage detection system. It describes the methods used in data collection, CNN model training, and testing as well as design, development, and deployment on the mobile application.

3.2 STUDY SET UP

This research followed a quantitative methods approach. To understand the nature of FAW attacks on maize plants, the researcher interacted with an entomologist from KARLO Embu branch between the 1st and the 10th of October 2021. The researcher gave an overview of the research aim how he planned to address it. In addition to understanding the nature of FAW attacks, the entomologist accompanied the researcher during data collection. This was critical to ensure the images captured had signs of FAW damage. This also informed the data annotation exercise that was conducted later where FAW damage was categorized as being in the early stages or late stages depending on the extent it was damaged. High quality images are vital during the training and testing phases of the CNN models.

3.3 RESEARCH DESIGN

The research followed an experimental research design approach. This approach was chosen since the researcher intended to compare the performance between the chosen models. The experiment was conducted on 10 datasets each containing 100 images. The choice of the CNN meta-architecture was the independent variable while the model performance (mean average precision) was the dependent variable. Previous studies have implemented this experimental design to compare the performance of different algorithms. (Carranza-García et al., 2020) used experimental research design to compare the performance of four CNN algorithms; RetinaNet, FCOS, and YOLOv3, and Faster R-CNN). (Asad & Bais, 2020) also used experimental research design to compare the performance of deep learning meta-architectures like SegNet and UNET and encoder blocks like VGG16 and ResNet-50.

3.4 DATASETS

3.4.1 DATA COLLECTION

The images used to train the model were collected directly by the researcher. The researcher took the images using a Samsung A750 2018 mobile phone. The image resolution was (5664 x 3184). The researcher collected 1000 images of maize leaves affected by FAW infestation from five farms in Gichugu constituency, Kirinyaga county Kenya between the 1st and the 10th of October 2021. To ensure accuracy of the data collection process, the researcher sought the help of an entomologist from KARLO Embu branch in identifying maize leaves infested by FAW and distinguishing them from those attacked by other maize pests. The researcher then split the images into 10 datasets each containing 100 images using simple random sampling method. After that, the researcher performed data annotation on each image using the open-source software called LabelImg. The annotated regions fell under two categories: Early FAW Infestation and Late FAW Infestation and saved in the PASCAL VOC format. (Carreira et al., 2015)

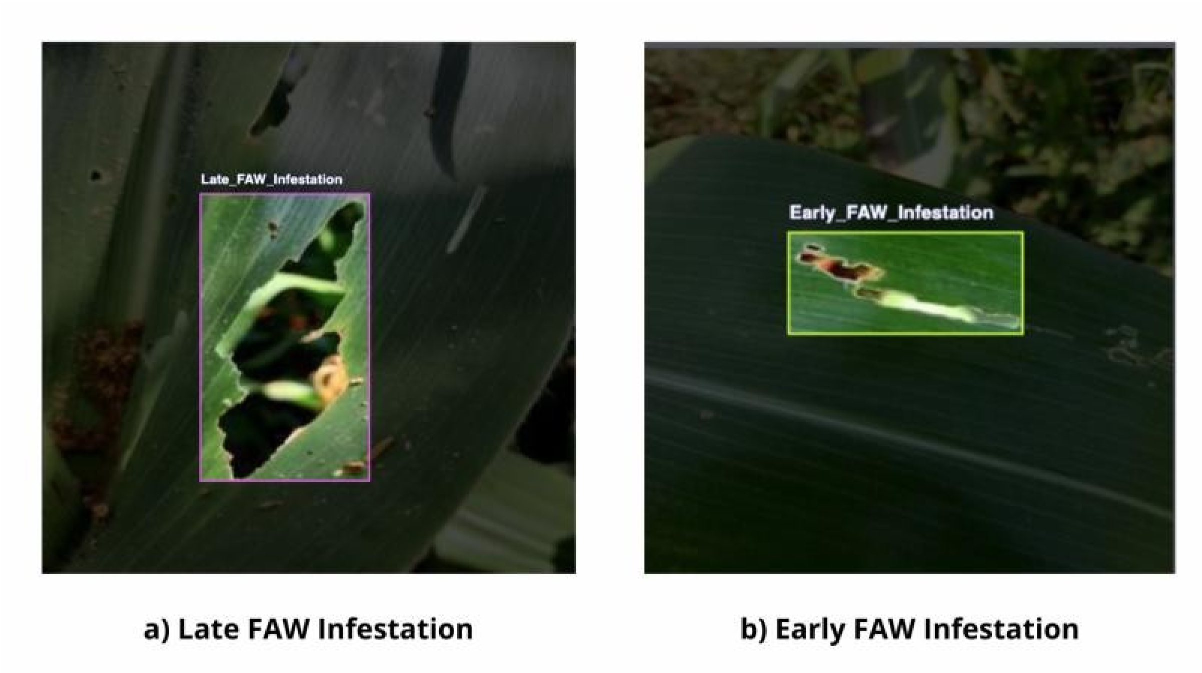


Figure 17: Late and Early FAW Infestation Annotated Images

3.4.2 DATA SELECTION AND DISTRIBUTION

After deriving 10 datasets from the total 1000 images and conducting data annotation, the images in each dataset were split into training, validation, and test sets. The train, validation and test splits are indicated in Table 1 below. This applied to each of the 10 datasets in this research

Table 1: Train, Validation, Test Split

	Training Set	**Validation Set**	**Test Set**	**Total**
Dataset	80	10	10	100
Split %	80%	10%	10%	

3.4 MODEL TRAINING AND TESTING

3.4.1 MODEL TRAINING

The researcher trained efficientdet lite04 and yolov4 tiny models for each dataset described above. Due to the computational requirements of training a CNN model, the researcher trained both models in Google Colab using the GPU hardware accelerator provided on the platform. To further reduce the training time, the researcher implemented deep transfer learning for both algorithms. The pretrained model weights initialized when training the models were used to train the COCO dataset. (Lin et al., 2014). The researcher followed the procedure proposed by (Jiang et al., 2020) to train the YOLOv4 tiny models. Efficentdet lite04 model was trained according to the procedure proposed by (M. Tan et al., 2020)

3.4.2 MODEL PERFORMANCE EVALUATION

The researcher used mean average precision to evaluate the performance of both CNN models. Mean Average Precision (mAP) is a popular metric used to measure the performance of object detection algorithms. To compute mean average precision, one must compute precision, recall and Intersection over Union (IoU). Object detection algorithms predict both the location of an object in an image and the class it belongs to. Intersection over Union computes the overlap between the predicted bounding boxes and the ground truth annotated bounding boxes. On the other hand, Precision computes correctly classified positive predictions as a ratio of all positive predictions while recall computes how the model correctly identifies true positives. Average Precision is computed to find the area under the precision-recall curve while mean Average Precision is the average of all computed Average Precision values. The formulas for calculating Intersection over Union, Precision and Recall, Average Precision and Mean Average Precision as shown below

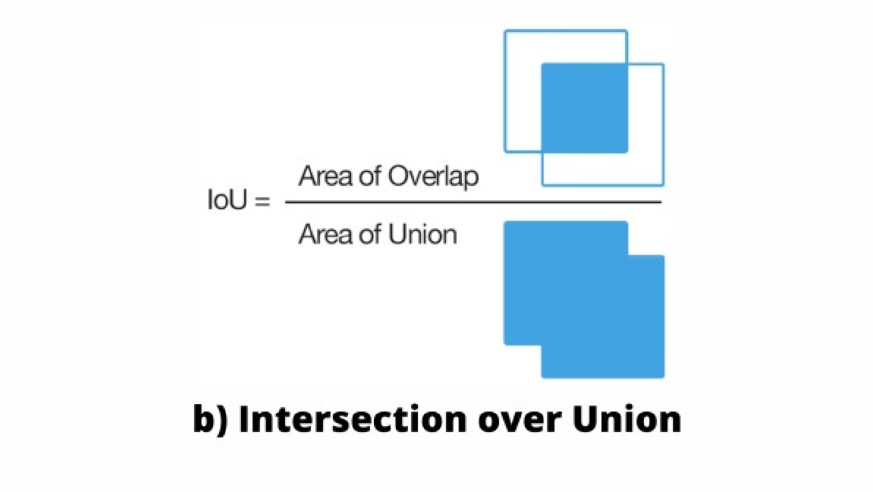
Equation 1: Precision and Recall

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

a) Precision and Recall

Equation 2: Intersection Over Union



Equation 3: Average Precision

$$\mathbf{AP} = \sum_{k=0}^{k=n-1} [\text{Recalls}(k) - \text{Recalls}(k+1)] * \text{Precision}(k)$$

Recalls(n) = 0, Precisions(n) = 1
n = number of thresholds

c) Average Precision

Equation 4: Mean Average Precision

$$mAP = \frac{1}{n}\sum_{k=1}^{k=n} AP_k$$

AP_k = the AP of class k

n = number of classes

d) Mean Average Precision

3.5 SOFTWARE DEVELOPMENT METHODOLOGY

Due to the limited time required to develop the project, the researcher used the rapid application development SDLC methodology. RAD model is an incremental model used when short development cycles are required. The development cycle has five phases namely, analysis and quick design, prototype cycles (Develop, Demonstrate, Refine), testing and finally deployment. (Geambaşu et al., 2011). During the analysis and quick design phase, the researcher conducted unstructured interviews with five farmers on the approaches they used to scout for FAW infestation in their farms. The researcher also conducted unstructured interview with an entomologist on FAW attack patterns within Kirinyaga county. The information gathered from both farmers and the entomologist guided the researcher in coming up with user requirements and other specifications relevant to the project. System modeling (Business Modeling, Data Modeling, Process Modeling) was also performed during the analysis and design phase. (Chrismanto et al., 2019). During the prototype cycles, the developer rapidly develops prototypes which were continuously refined based on feedback from the client / user. After coming up with a satisfactory product, the researcher deployed the model on the mobile application. The suitability of this methodology to this project was validated by three reasons. This project needed to be developed in extremely short time frame i.e. less than 3 months which fit well with the provisions of the RAD model. Secondly, user requirements were identified in the beginning of the project. This meant that an incremental model would be effective. Finally, it allows for component-based construction where the developer focuses on one component at a time then integrates the system. This allowed the researcher to develop the CNN model and the mobile application concurrently and test different modules according to their functionality.

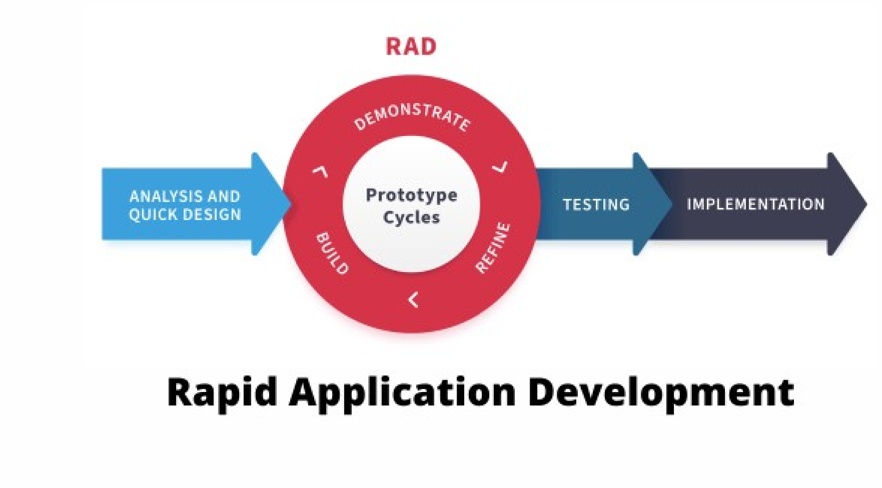


Figure 18: Rapid Application Development

3.6 ETHICAL CONSIDERATIONS.

When conducting the research, the researcher observed various ethical considerations. Before commencing any data collection, the farmers and entomologists were informed on the objectives of the study and the data that would be collected. This was a way of seeking consent before commencing the data collection process. The images needed in training the model were collected from local farms where no endangered species were present therefore no special permits were required to accomplish that. Confidentiality was observed throughout the research process.

CHAPTER FOUR: SYSTEMS DESIGN AND ARCHITECTURE

This section explores the process followed to design and develop the FAW damage detection mobile application. It highlights the interconnection between various components and interactions using UML diagrams.

4.1 REQUIREMENTS ANALYSIS

This subsection will provide the user requirements for the proposed fall armyworm damage detection mobile application. Both functional and non-functional requirements will be outlined. A combination of the research objectives covered in the first chapter and the user requirements will highlight different requirements that will be addressed in this research.

4.1.1 FUNCTIONAL REQUIREMENTS

Functional requirements specify what the system is required to do regarding inputs, outputs, and system behavior. The proposed system which will be implemented on a mobile application has a user interface and API module, libraries and services. The functional requirements identified in this research include:

a) The system should allow a user to take an image within the application
b) The system should be able to accurately perform FAW damage object detection on preset images.
c) The system should be able to accurately perform FAW damage object detection on images taken by the mobile camera
d) The system should be able to display bounding box, class, and confidence score on the user interface.

4.1.2 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are the properties the system should have. The most common non-functional requirements include architecture standards, coding standards, response time, processing time, query & reporting time among others. The non-functional requirements need to be addressed to provide excellent user experience / boost user satisfaction. The following non-functional requirements were identified for the FAW damage detection application:

a) Usability – the system interface should be easy to learn and perform the required functions

b) Reliability – the system should be able to perform its required functions in a reliable and consistently manner without failure.
c) Maintainability – the system should be easy to manage, find and fix bugs to match user requirements.
d) Response time – the system should display object detection results within 10 seconds after submitting the image.

4.2 SYSTEM ARCHITECTURE

The purpose of the system architecture is to provide an overview of how system components are interconnected and the communication going on between them. The goal being the achievement of the intended system functionalities. Figure 19 below shows the system architecture proposed for this research. It shows the flow of data after taking a maize leaf image using the device camera to the display of object detection results on the application's user interface.

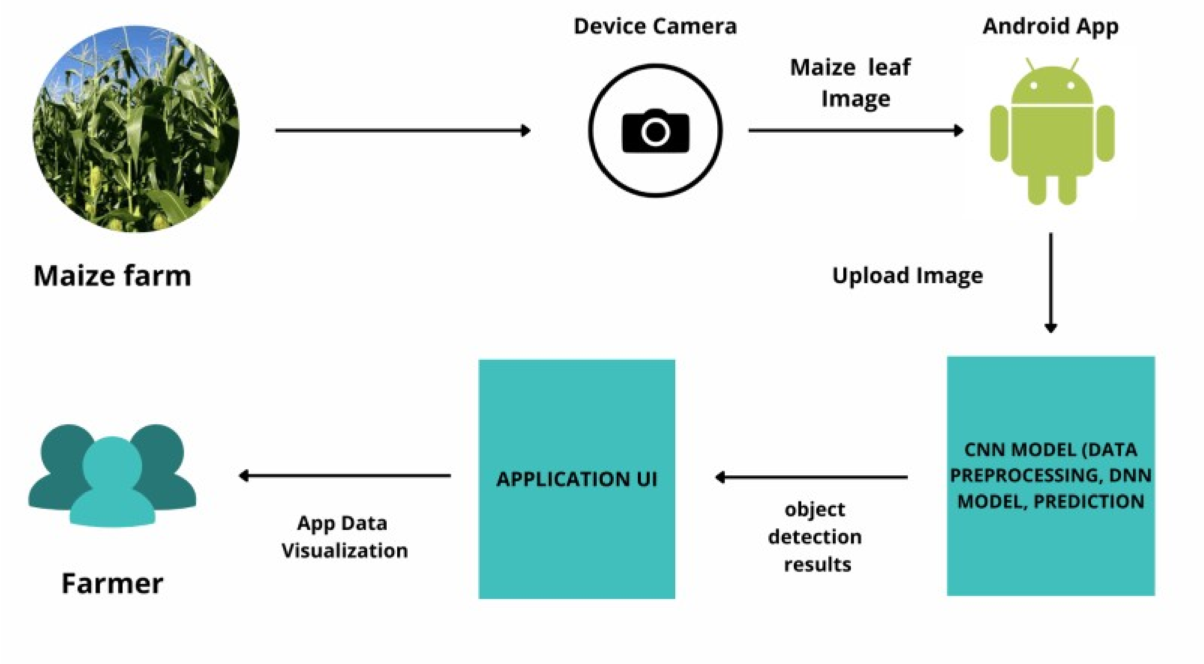


Figure 19: System Architecture

First, the system user will take a maize leaf image within the mobile application using a camera enabled device and submit it for further processing. On the mobile application, the image is uploaded and undergoes preprocessing after which it is object detection is performed. If the model's confidence level of detected objects are above a preset threshold, the application API displays the results on the application user interface otherwise they are discarded.

4.3 SYSTEM BEHAVIOR MODELING

In this subsection we will discuss how system behavior modeling was used to model user requirements above. The main actor in the system is the farmer who will use the application in the farm to access fall armyworm damage.

4.3.1 USE CASE DIAGRAMS

Use case diagrams provide a graphical representation of possible ways the user will interact with the system by showing the actors, use cases and how they relate. Use cases are represented using unified modeling language. The primary actors in the proposed system were the farmer referred to as the user and the system.

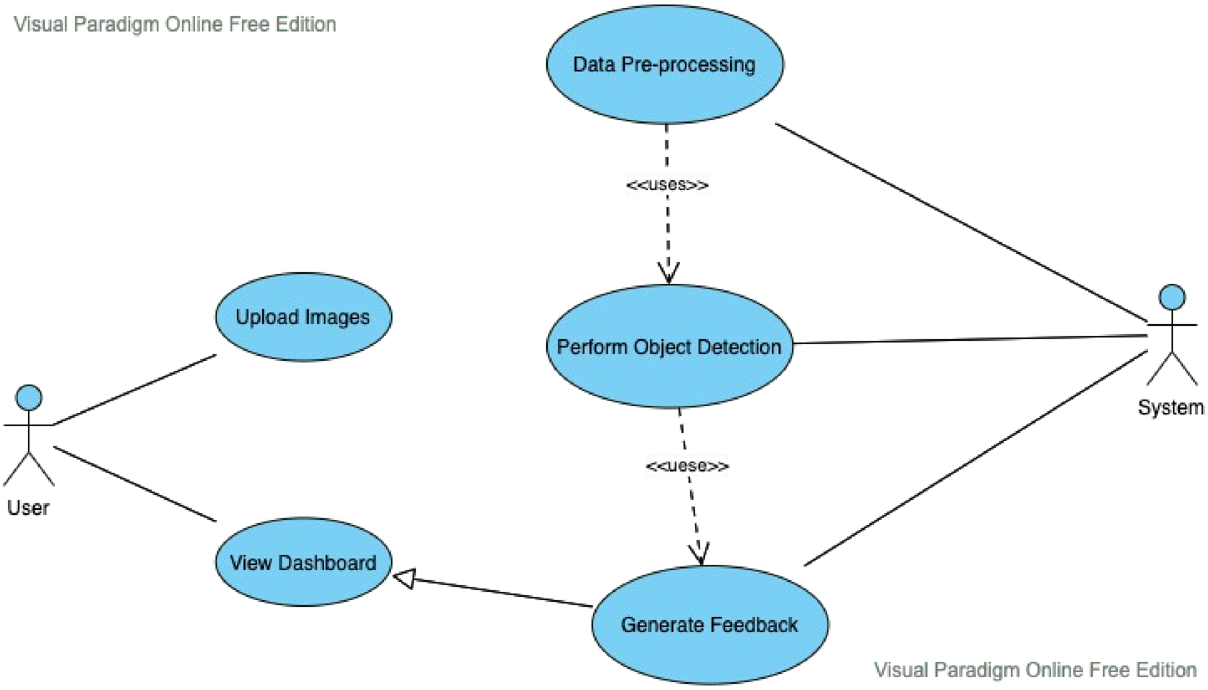


Figure 20: Use Case Diagram

4.3.2 USE CASE NARRATION

This is the textual narration of the events that occur when a user interacts with the proposed system.

Table 2: Upload Image use case narration

#	Use Case	Primary Actor	Pre-Condition	Post Condition
1	Upload Image	Farmer	User must have a smartphone with a camera and the FAW detection mobile application	Image successfully uploaded to the system
	Main Success Scenarios			
	Course of Events		**System Actions**	
	a) User navigated to the mobile app home page b) User clicks on the take image button c) User submits the image by clicking on the OK button		System uploads the image to the model for inference.	
	Alternative Course of Events			
	The user does not take an image			

Table 3: View Dashboard use case narration

#	Use Case	Primary Actor	Pre-Condition	Post Condition
2	View Dashboard.	Farmer	User must be on the mobile application user interface	User can view the dashboard

	Main Success Scenarios	
	Course of Events	**System Actions**
	User navigated to dashboard page	System takes user to dashboard page
	Alternative Course of Events	

Table 4:Perform Object Detection use case narration

#	Use Case	Primary Actor	Pre-Condition	Post Condition
3	CNN Model Inference.	System	Client device has a camera	Object Detection successfully performed on the image
	Main Success Scenarios			
	Course of Events		**System Actions**	
	a) System receives image as input parameter b) Model conducts preprocessing, performs object classification and localization c) System outputs the results to the user		a) The system performs object detection	
	Alternative Course of Events			
	b) System receives image of healthy maize leaves and does not perform any object detection task. c) System receives image containing maize leaves infested by other pests and does not			

	perform any object detection task.

4.3.3 SEQUENCE DIAGRAMS

Sequence diagrams will illustrate the sequence of messages between the objects. After the CNN model is trained and integrated on the mobile application, the application user (farmer) will take an image of a maize leaf in the farm and upload it. The image is the input parameter for the model. It undergoes data preprocessing before model inference occurs. The model inference results; that is class and bounding box are added on the image then displayed on the user interface as illustrated in the figure below

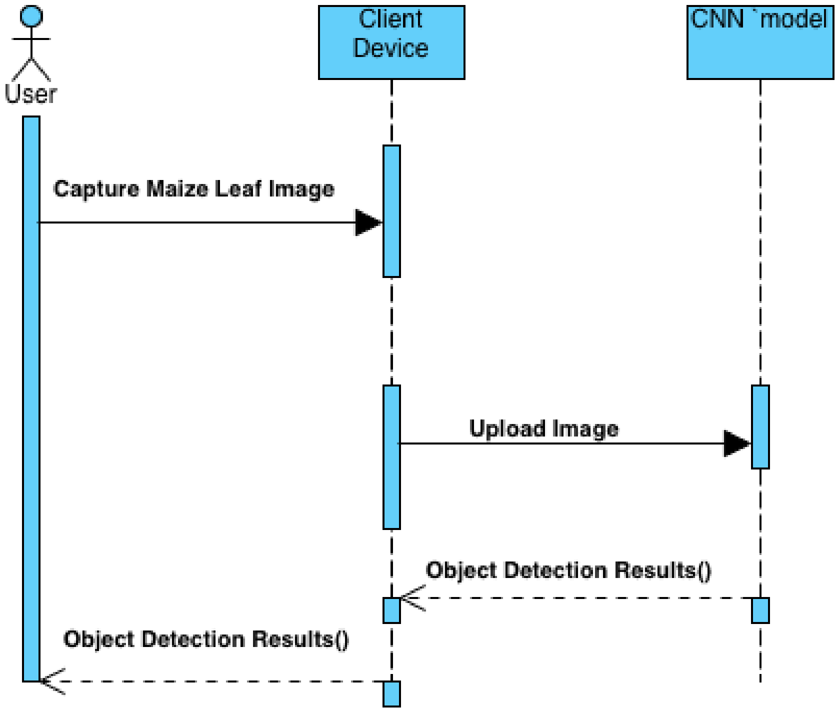


Figure 21: Sequence Diagram

4.4 PROCESS MODELING

4.4.1 CONTENT DIAGRAM

This is a model describing the interaction between the fall armyworm detection system and the immediate external entities (farmer). Figure 22 below shows the message exchanged between the two entities. The farmer takes maize leaf image and submits it and views the detection results from the FAW damage detection system.

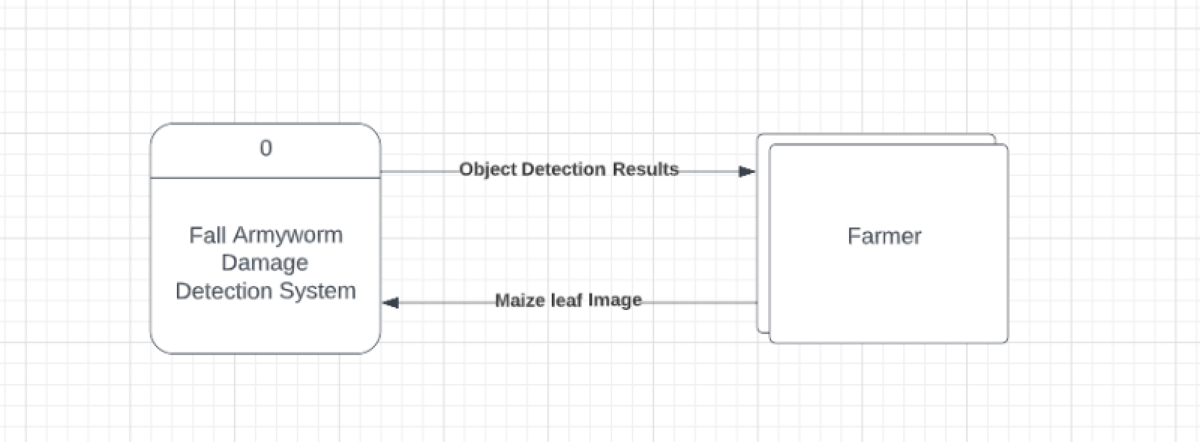


Figure 22: Context Diagram

4.4.2 LEVEL 1 DATA FLOW DIAGRAM

The level 1 data flow diagram provides a more detailed depiction of the content diagram. It highlights the constituent parts of the FAW damage detection system. After the farmer takes the image, it undergoes preprocessing before object detection is performed. The CNN model then analyzes the image, identifying regions of interest and assigning bounding box and class predictions to those that were above the preset threshold. The detection results are then fed into the feedback generator that includes the detection results and the original image. The output image is then viewed by the farmer.

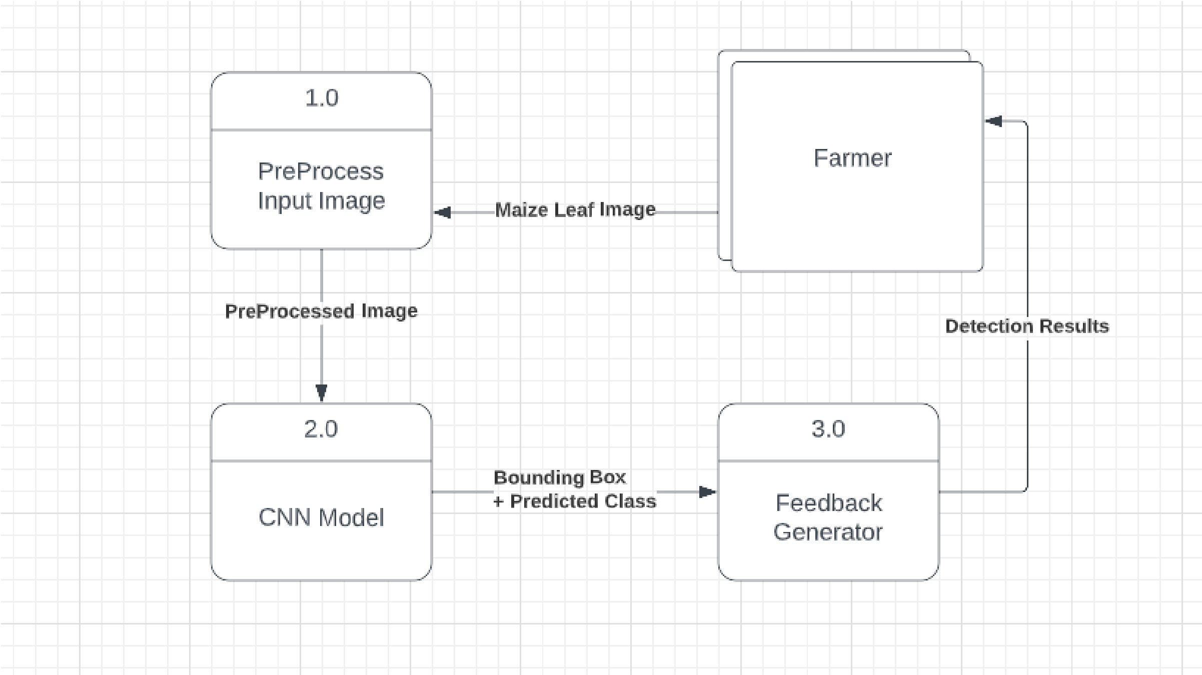


Figure 23: Level 1 Data Flow Diagram

4.5 SYSTEM IMPLEMENTATION

4.5.1 MOBILE APPLICATION

The mobile application used in this research was adopted from google object detection code lab. To meet the user requirements identified, the user interface was modified by adjusting the application dashboard name. However, the object detection capability of the application was added during this research. The mobile application was developed using Java and Kotlin. The researcher used android studio arctic fox 2020.3.1 integrated development environment. The object detection capability was added on the main activity file. The module Gradle script was also updated by adding the TensorFlow lite library on the dependencies section.

4.5.2 CNN MODEL TRAINING AND TESTING

The models were trained using python programming language on google colab. This was to overcome the computational limitations of the researcher’s computer. Google colab offers GPU hardware accelerators which speeds up the training process. In developing the CNN model, transfer learning was applied for both algorithms. By downloading pretrained weights used in the COCO dataset competition, the training time was greatly reduced. After training, the models were tested using the test images and the performance of the two models across

10 datasets the results were recorded, and the better performing algorithm was converted to a TensorFlow lite model.

4.5.3 SYSTEM TESTING

The researcher used three test cases to access the functionality of the mobile application. These tests were:

a) Does the application allow a farmer to take a picture?
b) Does the application allow a farmer to upload the picture captured in the first test case?
c) Does the application return object detection results to the farmer?

4.5.4 CNN MODEL DEPLOYMENT ON MOBILE APPLICATION

After identifying the better performing algorithm, the tflite model was incorporated on the mobile application. The model itself was added to the assets folder while the code was added to the main activity kit file. TensorFlow object detection API was also included in the Gradle script as one of the dependencies.

CHAPTER FIVE: RESULTS, EVALUATION AND DISCUSSION

5.1 INTRODUCTION

This chapter discusses the results and main findings observed during the research for the development of a fall armyworm damage detection system using CNN. This is in line with the research objectives and methodology described in prior chapters. The main aim of the study was the develop a fall armyworm damage detection system using convolutional neural networks, compare the performance of two one stage CNN meta-architectures and develop a mobile application that can be used in the field.

5.2 CONVOLUTIONAL NEURAL NETWORK TRAINING

5.2.1 YOLOv4 TINY PERFORMANCE

Following the model training method proposed in chapter three, the YOLOv4 TINY model was trained on all 10 datasets and the performance evaluated. The model was trained using deep transfer learning over 6000 training iterations. The mean average precision for each dataset is shown below. The best performing model had an accuracy of 82.5% and the worst performing one had an accuracy of 24.1%.

Table 5: YOLOv4 TINY mAP

# DATASET	1	2	3	4	5	6	7	8	9	10
mAP %	60.0	63.6	58.7	50.2	24.1	32.6	49.5	82.5	43.0	69.6

5.2.2 EFFICIENTDET LITE PERFORMANCE

The efficientdet lite model was trained over 100 iterations. The best performing model had an accuracy of 85.85% and the worst performing one had an accuracy of 29.3%.

Table 6: EfficientDet lite04 mAP

# DATASET	1	2	3	4	5	6	7	8	9	10
mAP %	54.04	67.89	55.61	45.85	85.85	42.11	29.30	57.10	55.09	56.53

5.3 QUANTITATIVE RESULTS

The two algorithms chosen in this study are optimized for mobile devices / edge devices. (Jiang et al., 2020; Nguyen et al., 2020). To compare the performance between the algorithm the researcher used paired samples t-test. This approach has been implemented in other studies to compare the performance of two algorithms trained on the same dataset. (Patel & Chatterjee, 2016). The experiments null hypothesis was that there is no statistical difference between the mean mAP of the models while the alternative hypothesis is that there is statistical difference between the mean mAP of the models

Null Hypothesis: There is no statistical difference between the performance of efficientdet lite model and yolov4 tiny model

Alternate Hypothesis: There is statistical difference between the performance of efficientdet lite model and yolov4 tiny model

The researcher used SPSS software to compute the paired t-test. The alpha value was 0.05 and the degrees of freedom were 9. The Null Hypothesis was that there is no statistical difference between the performance of the models while the alternative hypothesis is that there is statistical difference between the performance of the models. The figures below show the results from the statistical tests. The two dependent variables in this study were the yolov4 tiny CNN algorithm and the efficientdet lite CNN algorithm. Both algorithms were trained and tested on each of the 10 datasets used in this research.

Paired Samples Statistics

		Mean	N	Std. Deviation	Std. Error Mean
Pair 1	efficientdet_lite	54.9370	10	15.04162	4.75658
	yolov4_tiny	53.3800	10	17.35152	5.48703

Figure 24: Paired Sample Statistics

The mean mAP for the efficientdet lite4 algorithm was 53.38 (N=10) while the mean for the yolov4 tiny algorithm was 54.937 (N=10). The means of the efficientdet lite model was higher than that of the yolov4 tiny model.

Paired Samples Correlations

		N	Correlation	Sig.
Pair 1	efficientdet_lite & yolov4_tiny	10	-.125	.731

Figure 25: Paired Samples Correlations

Paired Samples Test

		Paired Differences					t	df	Sig. (2-tailed)
					95% Confidence Interval of the Difference				
		Mean	Std. Deviation	Std. Error Mean	Lower	Upper			
Pair 1	efficientdet_lite - yolov4_tiny	1.55700	24.33984	7.69693	-15.85467	18.96867	.202	9	.844

Figure 26: Paired Sample Differences

Figure 26 shows the experimental results of the paired samples t-test. The mean difference between the two algorithms is 1.557. The researcher observed a wide standard deviation of 24.34 showing a wide variability from the mean value. The t statistic from the experimental results was .0202 and the significance value of 0.844. The p value is greater than 0.05. This means that we failed to reject the null hypothesis. Therefore, we can conclude based on the results that there is no statistical difference between the mAP values of efficientdet lite model and yolov4 tiny model.

5.4 SYSTEM PROTOTYPE TESTING

This subsection discusses the test cases used to access the functionality of the mobile application. It includes the test cases, their level of importance and the test results

Table 7: Test Case Results

TEST CASE	IMPORTANCE	TEST RESULTS
Does the application allow a farmer to take a picture?	HIGH	The farmer successfully took a picture of a maize leaf.
Does the application allow a farmer to upload the picture captured in the first test case	HIGH	The farmer successfully uploaded the picture on the application
Does the application return object detection results to the farmer?	MEDIUM	The farmer observed object detection results if there was FAW damage on the maize

		leaves.

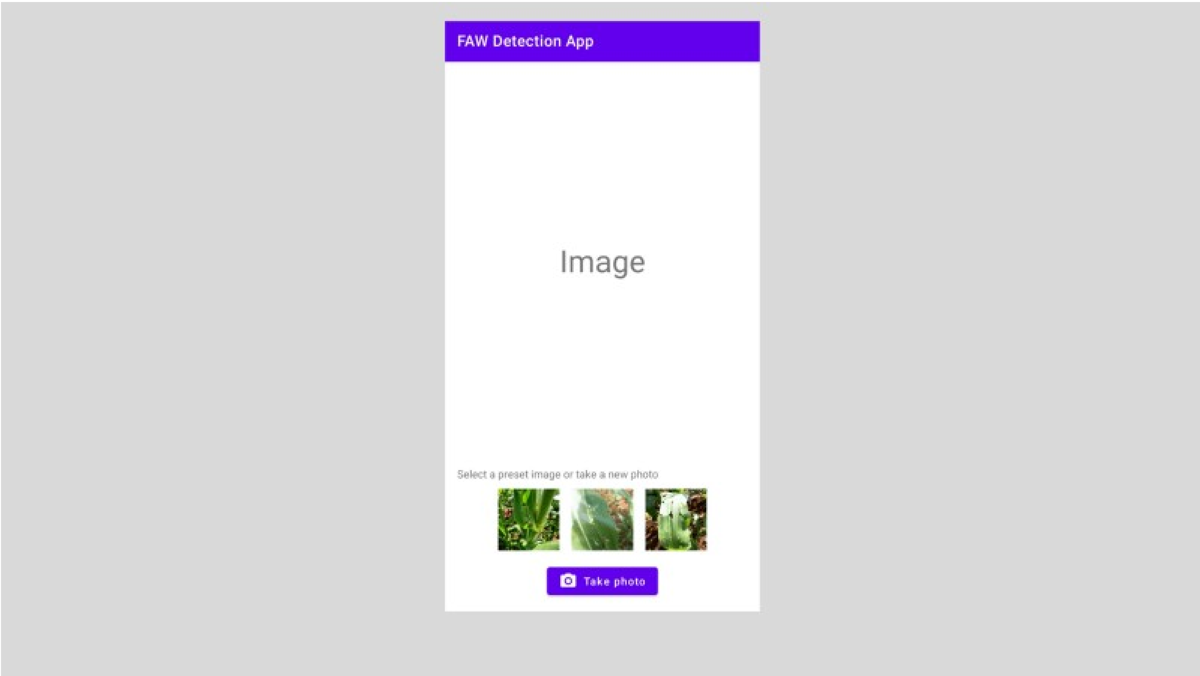


Figure 27: Sample Mobile Application Screenshot

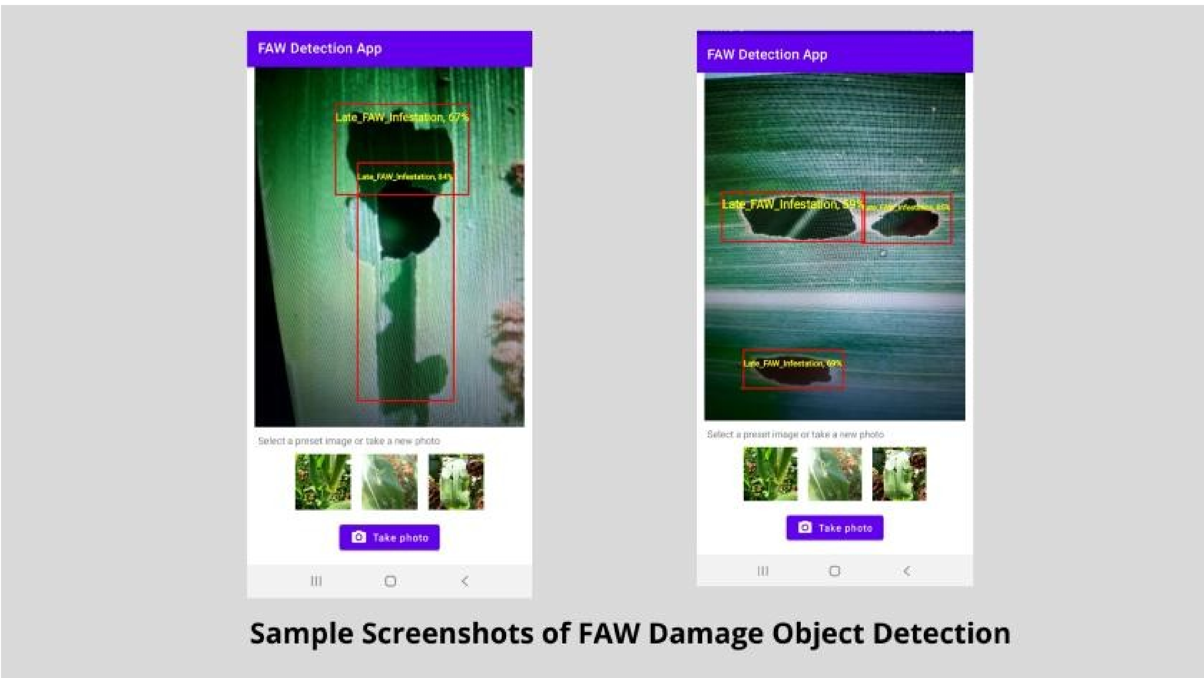


Figure 28: Sample Screenshots of FAW Damage Object Detection

5.5 PERFORMANCE COMPARISION WITH PREVIOUS STUDIES.

Other CNN based systems have been proposed in pest detection. To the best of our knowledge the proposed system is the first that focuses on FAW damage on plant leaves it invades. The proposed system utilizes one stage CNN model on mobile device to access the damage on maize leaves. The researcher also looked at other studies conducted on this field and summarized how the proposed system performed against others. The proposed system performs comparatively well considering no data augmentation was performed and the dataset used to train and test the model was small compared to the other two studies.

Table 8: Performance Comparison with other Studies

Proposed By	**Architecture**	**Pest Classification / Pest Detection**	**Best Performing mAP**
This research	Efficientdet lite	Pest Detection	85.85%
This research	Yolov4 - tiny	Pest Detection	82.5%
Fuentes (2017)	SSD	Pest Detection	85.10%
Lippi (2021)	YOLO	Pest Detection	94.5%

CHAPTER SIX: CONCLUSION AND RECOMMENDATION

6.1 CONCLUSION

This research study aimed at developing a mobile based fall armyworm damage detection system using one stage convolutional neural networks. From the reviewed literature, there is little research on CNN based fall armyworm damage detection system specifically tailed to mobile or edge devices. Upon completion of this project, all the objectives set were met. The researcher was able to collect maize leaf images in 5 farms within Kirinyaga county. The images were used to train and test the performance of the yolov4 tiny and efficientdet lite CNN models. Experimental results showed that efficientdet lite performed better than the yolov4 tiny model. To test the statistical significance between the mean performance of the two algorithms, the researcher used two tailed paired t-test. Hypothesis testing results showed that there is no statistical significance between the two algorithms that belong to the same class of one stage CNN meta-architectures. However, the researcher observed that the mean performance of the efficientdet lite models was higher than that of the yolov4 tiny models making it a better choice for the intended task in this research. The best performing efficientdet model had a mAP of 85.85% which is impressive for object detection task. This model was integrated on a mobile application for use in the farm. The successful training and deployment of the models led to achievement of the objectives set out in chapter one. It is hoped that the artifact developed in this study will be a vital integrated pest management tool in the fight against fall armyworm.

6.2 RECOMMENDATIONS

In view of this study the researcher has the following recommendations

1. The developed models and tools can be extended in detecting FAW infestation on other plants or an entirely different pest.
2. The mobile application can incorporate Spatial data to facilitate easy monitoring of locations experiencing rampant FAW infestation and take the necessary measures to contain the infestation.
3. Develop an iOS equivalent of the android application that can be used for integrated pest management.

References.

Arakeri, M. P. (2016). Computer vision based fruit grading system for quality evaluation of tomato in agriculture industry. *Procedia Computer Science*, *79*, 426–433.

Asad, M. H., & Bais, A. (2020). Weed detection in canola fields using maximum likelihood classification and deep convolutional neural network. *Information Processing in Agriculture*, *7*(4), 535–545.

Ateya, S. M. (2018). *Fall army-worm prediction model on the maize crop in Kenya: an internet of things based approach*. Strathmore University.

Baudron, F., Zaman-Allah, M. A., Chaipa, I., Chari, N., & Chinwada, P. (2019). Understanding the factors influencing fall armyworm (Spodoptera frugiperda JE Smith) damage in African smallholder maize fields and quantifying its impact on yield. A case study in Eastern Zimbabwe. *Crop Protection*, *120*, 141–150.

Carranza-García, M., Torres-Mateo, J., Lara-Benítez, P., & García-Gutiérrez, J. (2020). On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data. *Remote Sensing*, *13*(1), 89.

Carreira, J., Vicente, S., Agapito, L., & Batista, J. (2015). Lifting object detection datasets into 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(7), 1342–1355.

Cerda, R., Avelino, J., Gary, C., Tixier, P., Lechevallier, E., & Allinne, C. (2017). Primary and secondary yield losses caused by pests and diseases: Assessment and modeling in coffee. *PloS One*, *12*(1), e0169133.

Chen, J.-W., Lin, W.-J., Cheng, H.-J., Hung, C.-L., Lin, C.-Y., & Chen, S.-P. (2021). A smartphone-based application for scale pest detection using multiple-object detection methods. *Electronics*, *10*(4), 372.

Chen, X., & Gupta, A. (2017). An implementation of faster rcnn with study for region sampling. *ArXiv Preprint ArXiv:1702.02138*.

Chiu, M. T., Xu, X., Wei, Y., Huang, Z., Schwing, A. G., Brunner, R., Khachatrian, H., Karapetyan, H., Dozier, I., & Rose, G. (2020). Agriculture-vision: A large aerial image database for agricultural pattern analysis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2828–2838.

Chiwamba, S. H., Phiri, J., Nkunika, P. O. Y., Nyirenda, M., & Kabemba, M. M. (2018). An Application of Machine Learning Algorithms in Automated Identification and Capturing of Fall Armyworm (FAW) Moths in the Field. *ICICT2018, Lusaka, Zambai*.

Chrismanto, A. R., Santoso, H. B., Wibowo, A., Delima, R., & Kristiawan, R. A. (2019). Developing agriculture land mapping using Rapid Application Development (RAD): A case study from Indonesia. *International Journal of Advanced Computer Science and Applications*, *10*(10).

Dai, J., Li, Y., He, K., & Sun, J. (2016). R-fcn: Object detection via region-based fully convolutional networks. *Advances in Neural Information Processing Systems*, *29*.

Day, R., Abrahams, P., Bateman, M., Beale, T., Clottey, V., Cock, M., Colmenarez, Y., Corniani, N., Early, R., & Godwin, J. (2017). Fall armyworm: impacts and implications for Africa. *Outlooks on Pest Management*, *28*(5), 196–201.

De Groote, H., Kimenju, S. C., Munyua, B., Palmas, S., Kassie, M., & Bruce, A. (2020). Spread and impact of fall armyworm (Spodoptera frugiperda JE Smith) in maize production areas of Kenya. *Agriculture, Ecosystems & Environment*, *292*, 106804.

Early, R., González-Moreno, P., Murphy, S. T., & Day, R. (2018). Forecasting the global extent of invasion of the cereal pest Spodoptera frugiperda, the fall armyworm. *BioRxiv*, 391847.

Ebrahimi, M. A., Khoshtaghaza, M. H., Minaei, S., & Jamshidi, B. (2017). Vision-based pest detection based on SVM classification method. *Computers and Electronics in Agriculture*, *137*, 52–58.

Edoh Ognakossan, K., Affognon, H. D., Mutungi, C. M., Sila, D. N., Midingoyi, S.-K. G., & Owino, W. O. (2016). On-farm maize storage systems and rodent postharvest losses in six maize growing agro-ecological zones of Kenya. *Food Security*, *8*(6), 1169–1189.

Ensari, T., Armah, D. C., Balsever, A. E., & Dağtekin, M. (2020). Convolutional Neural Networks for Image-Based Digital Plant Phenotyping. *Avrupa Bilim ve Teknoloji Dergisi*, 338–342.

Fuentes, A., Yoon, S., Kim, S. C., & Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, *17*(9), 2022.

Geambaşu, C. V., Jianu, I., Jianu, I., & Gavrilă, A. (2011). Influence factors for the choice of a software development methodology. *Accounting and Management Information Systems*, *10*(4), 479–494.

Ghadge, P., Sand, I. P., & Jadhav, I. P. (n.d.). *APPLICATIONS OF COMPUTER VISION IN AGRICULTURE*.

Groote, H. De, Overholt, W. A., Ouma, J. O., & Wanyama, J. (2011). Assessing the potential economic impact of Bacillus thuringiensis (Bt) maize in Kenya. *African Journal of Biotechnology*, *10*(23), 4741–4751.

He, Y., Zeng, H., Fan, Y., Ji, S., & Wu, J. (2019). Application of deep learning in integrated pest management: A real-time system for detection and diagnosis of oilseed rape pests. *Mobile Information Systems*, *2019*.

Howard, A., Zhmoginov, A., Chen, L.-C., Sandler, M., & Zhu, M. (2018). *Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation*.

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., & Guadarrama, S. (2017). Speed/accuracy trade-offs for modern convolutional object detectors. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7310–7311.

Jayswal, H. S., & Chaudhari, J. P. (2020). *Plant Leaf Disease Detection and Classification using Conventional Machine Learning and Deep Learning*.

Jiang, Z., Zhao, L., Li, S., & Jia, Y. (2020). Real-time object detection method based on improved YOLOv4-tiny. *ArXiv Preprint ArXiv:2011.04244*.

Kassie, M., Wossen, T., De Groote, H., Tefera, T., Sevgan, S., & Balew, S. (2020). Economic impacts of fall armyworm and its management strategies: evidence from southern Ethiopia. *European Review of Agricultural Economics*, *47*(4), 1473–1501.

Kemajou, V. N., Bao, A., & Germain, O. (2019). Wellbore schematics to structured data using artificial intelligence tools. *Offshore Technology Conference*.

Khatri, S., Pakuwal, P., & Khanal, S. (2020). Archives of Agriculture and Environmental Science. *Archives of Agriculture and Environmental Science*, *5*(4), 583–591.

Kumela, T., Simiyu, J., Sisay, B., Likhayo, P., Mendesil, E., Gohole, L., & Tefera, T. (2019). Farmers' knowledge, perceptions, and management practices of the new invasive pest, fall armyworm (Spodoptera frugiperda) in Ethiopia and Kenya. *International Journal of Pest Management*, *65*(1), 1–9.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. *European Conference on Computer Vision*, 740–755.

Liu, J., & Wang, X. (2021). Plant diseases and pests detection based on deep learning: a review. *Plant Methods*, *17*(1), 1–18.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. *European Conference on Computer Vision*, 21–37.

Meissle, M., Mouron, P., Musa, T., Bigler, F., Pons, X., Vasileiadis, V. P., Otto, S., Antichi, D., Kiss, J., & Pálinkás, Z. (2010). Pests, pesticide use and alternative options in European maize production: current status and future prospects. *Journal of Applied Entomology*, *134*(5), 357–375.

Mohajan, H. (2014). *Food and nutrition scenario of Kenya*.

Nguyen, H.-H., Tran, D. N.-N., & Jeon, J. W. (2020). Towards real-time vehicle detection on edge devices with nvidia jetson tx2. *2020 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, 1–4.

Oerke, E.-C. (2006). Crop losses to pests. *The Journal of Agricultural Science*, *144*(1), 31–43.

Padhee, A. K., & Prasanna, B. M. (2019). The emerging threat of Fall Armyworm in India. *Indian Farming*, *69*(1), 51–54.

Patel, A. K., & Chatterjee, S. (2016). Computer vision-based limestone rock-type classification using probabilistic neural network. *Geoscience Frontiers*, *7*(1), 53–60.

Patil, A. R. B., Sharma, L., Aochar, N., Gaidhane, R., Sawarkar, V., Fulzele, P., & Mishra, G. (2020). A literature review on detection of plant diseases. *European Journal of Molecular & Clinical Medicine*, *7*(07), 2020.

Patrício, D. I., & Rieder, R. (2018). Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and Electronics in Agriculture*, *153*, 69–81.

Paul, A., Ghosh, S., Das, A. K., Goswami, S., Das Choudhury, S., & Sen, S. (2020). A review on agricultural advancement based on computer vision and machine learning. *Emerging Technology in Modelling and Graphics*, 567–581.

Qin, Z., Li, Z., Zhang, Z., Bao, Y., Yu, G., Peng, Y., & Sun, J. (2019). ThunderNet: Towards real-time generic object detection on mobile devices. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6718–6727.

Rustia, D. J. A., Chao, J., Chiu, L., Wu, Y., Chung, J., Hsu, J., & Lin, T. (2021). Automatic greenhouse insect pest detection and recognition based on a cascaded deep learning classification method. *Journal of Applied Entomology*, *145*(3), 206–222.

Selvaraj, M. G., Vergara, A., Ruiz, H., Safari, N., Elayabalan, S., Ocimati, W., & Blomme, G. (2019). AI-powered banana diseases and pest detection. *Plant Methods*, *15*(1), 1–11.

Sonka, M., Hlavac, V., & Boyle, R. (2014). *Image processing, analysis, and machine vision*. Cengage Learning.

Soviany, P., & Ionescu, R. T. (2018). Optimizing the trade-off between single-stage and two-stage deep object detectors using image difficulty prediction. *2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, 209–214.

Sun, Y., Liu, X., Yuan, M., Ren, L., Wang, J., & Chen, Z. (2018). Automatic in-trap pest detection using deep learning for pheromone-based Dendroctonus valens monitoring. *Biosystems Engineering*, *176*, 140–150.

Syarief, M., & Setiawan, W. (2020). Convolutional neural network for maize leaf disease image classification. *Telkomnika (Telecommunication Computing Electronics and Control)*, *18*(3), 1376–1381.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A survey on deep transfer learning. *International Conference on Artificial Neural Networks*, 270–279.

Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10781–10790.

Thenmozhi, K., & Reddy, U. S. (2019). Crop pest classification based on deep convolutional neural network and transfer learning. *Computers and Electronics in Agriculture*, *164*, 104906.

Tian, H., Wang, T., Liu, Y., Qiao, X., & Li, Y. (2020). Computer vision technology in agricultural automation—A review. *Information Processing in Agriculture, 7*(1), 1–19.

Tuggener, L., Elezi, I., Schmidhuber, J., Pelillo, M., & Stadelmann, T. (2018). Deepscores-a dataset for segmentation, detection and classification of tiny objects. *2018 24th International Conference on Pattern Recognition (ICPR)*, 3704–3709.

Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, *2018*.

Wang, R., Liu, L., Xie, C., Yang, P., Li, R., & Zhou, M. (2015). Title. Sen-sors 2021, 1, 0. *Neural Information Processing Systems*, 91–99.

Yu, D., Wang, H., Chen, P., & Wei, Z. (2014). Mixed pooling for convolutional neural networks. *International Conference on Rough Sets and Knowledge Technology*, 364–375.

Zhao, W., Wang, X., Qi, B., & Runge, T. (2020). Ground-level mapping and navigating for agriculture based on IoT and computer vision. *IEEE Access*, *8*, 221975–221985.