2021-07

Image - based poultry disease detection using deep convolutional neural network

Mbelwa, Hope

NM-AIST

https://doi.org/10.58694/20.500.12479/1344

IMAGE - BASED POULTRY DISEASE DETECTION USING DEEP CONVOLUTIONAL NEURAL NETWORK

Hope Emmanuel Mbelwa

A Dissertation Submitted in Partial Fulfilment of the Requirements for the Degree of Master's in Information and Communication Science and Engineering of the Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

July, 2021

ABSTRACT

The poultry sector in the country is highly affected by diseases including Coccidiosis, Salmonella and Newcastle that have a significant impact on production. Lack of reliable information and proper methods of farming has led to the spread of the diseases as the majority of farmers practice traditional farming, hence lack a systematic way to detect and diagnose the disorders. Poultry farmers rely on experts to diagnose and detect the diseases; access to the experts is also a challenge due to the limited number of extension officers. With the available tools from artificial intelligence and machine learning, there is a potential to semi-automate the diagnostics process for the most common diseases in chickens. This study proposes a solution for predicting diseases in chickens using faecal images and deep Convolutional Neural Networks (CNN). Additionally, the work leverages the use of pre-trained models and develop the solution for the same problem. Based on the comparison, it is indicated that the model developed from the XceptionNet deep learning framework outperforms other models for all the metrics used. The experimental results indicate the accuracy of transfer learning at 94% using pre-training over the other models from fully training on the same dataset. The results show that the pre-trained XceptionNet framework (94%) has the best overall performance and highest prediction accuracy, and can be suitable for chicken disease detection application. The findings show that the proposed model is ideal for poultry diseases detection method.

DECLARATION

I, Hope Emmanuel Mbelwa, do hereby declare to the Senate of Nelson Mandela African Institution of Science and Technology that this dissertation is my own original work and that it has neither been submitted nor being concurrently submitted for degree award in any other institution.

Hope Emmanuel Mbelwa		27/07/2021
Candidate Name	**Signature**	**Date**

The above declaration is confirmed by:

Dr. Dina Z. Machuve		27/07/2021
Supervisor Name	**Signature**	**Date**

Dr. Jimmy T. Mbelwa		27/07/2021
Supervisor Name	**Signature**	**Date**

COPYRIGHT

CERTIFICATION

The undersigned certify that they have read and hereby recommend for acceptance by The Nelson Mandela African Institution of Science and Technology, a dissertation entitled, *Image Based Poultry Diseases Detection using Deep Convolutional Neural Network* submitted in partial fulfilment of the requirements for award of the degree of "Master's in Information and Communication Science and Engineering" of the Nelson Mandela African Institution of Science and Technology.

Dr. Dina Z. Machuve		27/07/2021
Supervisor Name	**Signature**	**Date**

Dr. Jimmy T. Mbelwa		27/07/2021
Supervisor Name	**Signature**	**Date**

ACKNOWLEDGEMENTS

Without the encouragement of a large number of people who have supported me intellectually as well as emotionally, the work I report in this dissertation would never have been possible.

First and foremost, I thank God for his grace upon me with an opportunity to pursue studies at Nelson Mandela African Institution of Science and Technology (NM-AIST). I am very grateful to the African Development Bank (AfDB) project for granting me a scholarship to fund my studies.

It has been an honour to work under the supervision of Dr. Dina Machuve and Dr. Jimmy Mbelwa, their scientific and professional mentorship has helped me complete this work successfully. I appreciate all their efforts.

Over the two years of my study, I have also depended heavily on the NM-AIST community all lecturers and supporting staff, my colleagues, research group and friends. I thank you all for the helpful hand.

Lastly, on a more personal level, I would like to extend my sincere gratitude to my family for their forever support and love. On a special note, I thank my mom Leticia Mbelwa, uncle Gideon Mwesiga, aunt Eng. Immaculata Raphael, brother Romwald Byarugaba, sister Leticia Nshange and my best friend Rehema Mlangwa for believing in me and supporting my academic journey. To my sisters, brothers, aunts, uncles, cousins and friends your endless prayers and encouragement pushed me throughout this journey.

Thank you very much, and be blessed.

DEDICATION

To my loving mom Leticia Raphael Mbelwa.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS AND SYMBOLS

CNN	Convolutional Neural Network
DCNN	Deep Convolutional Neural Network
DL	Deep Learning
GDP	Gross Domestic Product
He Uniform	Variance Scaling Initializer
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
JPEG	Joint Photographic Group
ODK	Open Data Kit
PCR	Polymerase Chain Reaction
ReLU	Rectified Linear Activation function
RGB	Red Green Blue
SGD	Stochastic Gradient Descend
SVM	Support Vector Machine
TPU	Tensor Processing Unit
TL	Transfer Learning
VGG	Visual Geometry Group

CHAPTER ONE

INTRODUCTION

1.1 Background of the Problem

Poultry farming is a sub-sector of agriculture that involves the keeping of domesticated birds such as chicken, ducks, pigeons and turkeys for food and commercial purposes. Tanzania mainly depends on agriculture for its economic growth and the sector contributes up to 30% of the Gross Domestic Product (GDP). The poultry sector in Tanzania supports up to 37 million households, whereby farmers keep different birds, mainly chicken, with a population of 96% of livestock in the country (Food and Agriculture Organization [FAO], 2013).

Different birds, mostly chicken, are kept for food security (meat and eggs) and income (Mulisa *et al*., 2014). In Tanzania, poultry farming is practiced both traditionally and commercially, traditional poultry supplies 94% of the poultry meat and eggs (FAO, 2013). Most poultry farmers are located in rural areas (over 72%) with an average flock size per household of 20 animals (Ministry of Livestock and Fisheries, 2015), hence acting as a source of income and poverty reduction.

However, farmers face many challenges devastating diseases being the major one. Chicken are highly infected with coccidiosis, salmonella and Newcastle diseases, which cause a high mortality rate of the livestock. Some of the disorders, for instance, Salmonella is zoonotic, meaning it can even spread to human hence affecting the health of the community. In addition to diseases, farmers lack access to reliable sources of information on poultry due to a few numbers of extension officers, distant locations for consultations and lack of awareness on recommended animal husbandry practices (Lwoga *et al*., 2010). They instead rely on word of mouth from friends and their ways and tradition.

Farmers need solutions to the problems to increase productivity; the use of vaccines is the common countermeasure at hand though does not apply to all diseases. The recent use of deep learning in disease detection motivates the need to contribute to robust diagnostics (Albarqouni *et al*., 2016).

1.2 Statement of the Problem

The growth rate of the poultry population is low, with an average of 2.6% annual growth in Tanzania mainland (Ministry of Livestock and Fisheries, 2015). Poultry production is greatly affected by the Newcastle, Coccidiosis and Salmonella diseases (Quiroz-Castañeda *et al*., 2015; State, 2016). The diseases have led to severe losses and spending a large amount of money on treatment of the infected chicken, thus leading to low yield (Liakos *et al*., 2018). The diseases as mentioned earlier have a significant negative economic impact to poultry farmers resulting to high financial losses (Desin *et al*., 2013), failure to compete on the export market and spread of diseases to human. Existing methods for diagnosis of poultry diseases are laboratory procedures including the Polymerase Chain Reaction (PCR) procedure. It takes 3-7 days to get the results hence, time consuming and expensive (Abdisa *et al*., 2017). Treatment of the sicknesses after late identification results to the high mortality rate of the birds (Wong *et al*., 2017). Therefore, this study aims at developing a model for early detection of poultry diseases using deep learning for early detection of the diseases.

1.3 Rationale of the Study

The deployment of technology in the agriculture sector involving greenhouse farming and use of deep learning techniques in production has enabled an increase in production for the farmers. Poultry farmers also need advanced technology in their farming practices (Liakos *et al*. 2018). Improvement in the poultry keeping practices shall mitigate the effects caused by the diseases, as various studies indicate that the most efficient way to manage poultry diseases is via early detection and treatment (Yazdanbakhsh *et al*., 2017). The target group of this study is small to medium scale poultry farmers who use deep litter or semi-commercial production system (Wong *et al*., 2017). The farmers are challenged with inadequate biosecurity measures and limited access to poultry health services compared to the large-scale commercial poultry system (Hemalatha *et al*., 2014). Only a small proportion of the diseases that affect poultry can be controlled with vaccination and studies show that very few countries in Africa, 28% use models to solve different problems (Brooks-Pollock *et al*., 2015). Therefore, there is a need for automated tools with efficiency and effective methods for diagnostics of the diseases that will lead to better yield and increase in production.

1.4 Research Objectives

1.4.1 General Objective

To develop a model for early detection of poultry diseases using deep convolutional neural network for poultry farmers in Tanzania.

1.4.2 Specific Objectives

The following specific objectives guide the study:

(i) To review existing approaches for poultry diseases diagnostics and identify the requirements for the proposed model.

(ii) To develop a model based on deep convolutional neural network for early diagnostics of Newcastle, Coccidiosis and Salmonella diseases.

(iii) To evaluate the performance of the model and validation for poultry diseases diagnostics.

1.5 Research Questions

The objectives of this study are to address the following questions:

(i) What are the existing approaches for poultry diseases diagnostics?

(ii) How will the model for poultry diseases diagnostics be developed using deep convolution neural network?

(iii) What is the performance of the proposed model?

1.6 Significance of the Study

The study will be useful to farmers, as the outcome is an automated tool for early detection of diseases affecting chicken, contributing to robust diagnostic measures. This tool will enable farmers to overcome the loss incurred due to late diagnosis of the disorders affecting chicken.

In the scientific body of knowledge, the study has contributed in developing a model based on a deep learning approach that can be deployed in smartphones that will be easily used by the farmers, extension officers and other stakeholders. Also, the study has developed a dataset for diagnostics of poultry diseases that can be acquired on open access for further studies and research.

1.7 Delineation of the Study

Deep Convolutional Neural Networks (DCNN) application is a broad field. In this study, the objective is to classify Coccidiosis, Health and Salmonella infected chicken based on their dropping's images as a dataset. To develop a model that can accurately detect the diseases based on the images and as per physical knowledge, the authors claim this work to be unique since no work has been done to detect the diseases in chicken using faecal sample data.

Despite the better performance of the model, the study lacked enough data for computer vision tasks. It is observed that the objective of the study was to classify four classes; health, Coccidiosis, Salmonella and Newcastle, but it only worked on three classes due to the following reasons: (a) The nature of the data is faecal samples images from chicken, due to the nature of Newcastle disease; it is a very devastating and spreads fast with high mortality rate about 100%, (b) Obtaining enough Newcastle data was hindered by lack of digestive signs from the inoculation period leading to respiratory symptoms that led to the death of the chickens, (c) The few data of Newcastle disease were all obtained from natural infection but at rare cases, since it is difficult to identify the disease early.

CHAPTER TWO

LITERATURE REVIEW

2.1 Disease Overview

2.1.1 Coccidiosis

Coccidiosis is caused by parasites of the genus *Eimeria* that affects the intestinal tracts of poultry, whereby poultry is host to seven species of Eimeria. Coccidiosis is ranked as a leading cause of death in poultry with *Eimeria tenella* among the most pathogenic parasite (Lim *et al*., 2012). The typical diagnostic procedure involves counting the number of oocytes (expressed as oocytes per gram [opg]) in the feces and examining the intestinal tract to determine the lesion score (Grilli *et al*., 2018). Clinical signs include blood/ brown diarrhoea.

2.1.2 Newcastle

Newcastle disease is an acute viral infection in domestic poultry and other bird species. It is caused by avian paramyxovirus serotype 1 (APMV-1) viruses (State, 2016). Newcastle disease virus (NDV), APMV-1 is diagnosed by serology or virus isolation tests or real-time reverse- transcription Polymerase Chain Reaction (PCR) procedure. Clinical signs include Greenish watery diarrhoea.

2.1.3 Salmonella

Salmonella is bacterial pathogens of genus *Salmonella* that cause disease to poultry and humans. *Salmonella pullorum* (SP) and *Salmonella gallinarum* (SG) pathogens cause pullorum disease and fowl typhoid in poultry, respectively (Desin *et al*., 2013). *Salmonella enteritidis* (SE) and *Salmonella typhimurium* (ST) strains are associated with human infections transmitted through the food-chain of poultry and poultry products. The PCR procedure is used for detection and identification of the various Salmonella strains. Clinical signs include white diarrhoea.

2.2 Deep Learning

Deep Learning (DL) is a class of machine learning algorithms inspired by the structure of the human brain. The DL algorithms use complex, multi-layered neural networks, where the level of abstraction increases gradually by non-linear transformations of input data. It has been applied in various fields, to mention a few as follows:

2.2.1 Image Recognition

Images are artefacts that depict visual perception. They have been used for diagnosis and detection in various fields including medical, agricultural and other fields. There are different existing image datasets that are on the cloud accessed when training different models. These datasets include Fashion-MNIST, CIFAR-10, Caltech-101 (Iorga & Neagoe, 2019; Wang *et al*., 2020) to mention a few.

Various computer vision studies have used image datasets in either classification, detection, recognition and segmentation of different research problems. Recently, the character portrayal is commonly used in disease diagnostics using images. Different levels of features are captured and analysed based on various aspects, including colour. Images are pre-processed (labelled and tuned) in order to maintain the realization of the former facts. A study by Albarqouni *et al*. (2016) used breast cancer histology images for detection of breast cancer disease; another research by Zhang and Han (2020) applied ultrasound images in the detection of ovarian tumours.

Zhang *et al.* (2020) worked to improve disease diagnostics using different body parts image dataset. Similarly, In the agricultural field, researchers (Ferentinos, 2018; Owomugisha *et al*., 2014; Ramcharan *et al*., 2017) have created and used leaf image datasets in the diagnosis of diseases in different plants like tomato, cassava, bananas and wheat.

2.2.2 Natural Language Processing

This deals with the interaction of human language and the computer. Deep Learning has enabled recommender systems that assist in different areas like health field, banking systems and many others. It helps in polysemous resolution, linguistic inference, word portrayal, sentimental implication, knowledge uncovering and text classification (Cambria & White, 2014).

2.2.3 Speech and Audio Recognition

The aim of speech recognition is to recognize the words spoken by absolutely everyone effectively on a computer or machine. There is no need to pay attention to more personal information, such as an accent. With this technology, the primary objective is to achieve optimum precision and speed with speech recognition, far above even the highest skill of humans. Speech recognition, on the other hand, is about being able to recognize and comprehend one particular voice example Siri on Apple and Alexa on Amazon (Rani *et al*. 2017) .

2.2.4 Autonomous Driving and Robots

This involves self-driving cars sensing the environment and moving safely with no human input. Deep Learning has enabled functionalities in the programmed vehicles and reduce the need for person interference to operate the vehicles (Bojarski *et al*., 2016). Beyond DL this study introduces the Convolutional Neural networks concept that will be applied.

2.3 Convolutional Neural Networks

Convolutional Neural Networks (CNN's) is possibly the most common architecture for deep learning due to its effective performance. They enable different activities like anomaly detection, natural language processing, web search and advertising. It is also a significant step up where it is getting good at better health care, reading x-rays images, drug discovery, precision agriculture and self-driving cars. The CNN's mimic how the brain works, enabling them to interpret images similar or more than humans can. They comprise of several hidden layers that help to analyse visual imagery. They consist of an input layer, several hidden layers and an output layer. When counting the number of layers, the input layer is ignored, consider the hidden layers, and output layer. Figure 1 illustrates how CNN processes an image. It consists of different layers input, convolutional, pooling, fully connected, and softmax (output) layers.

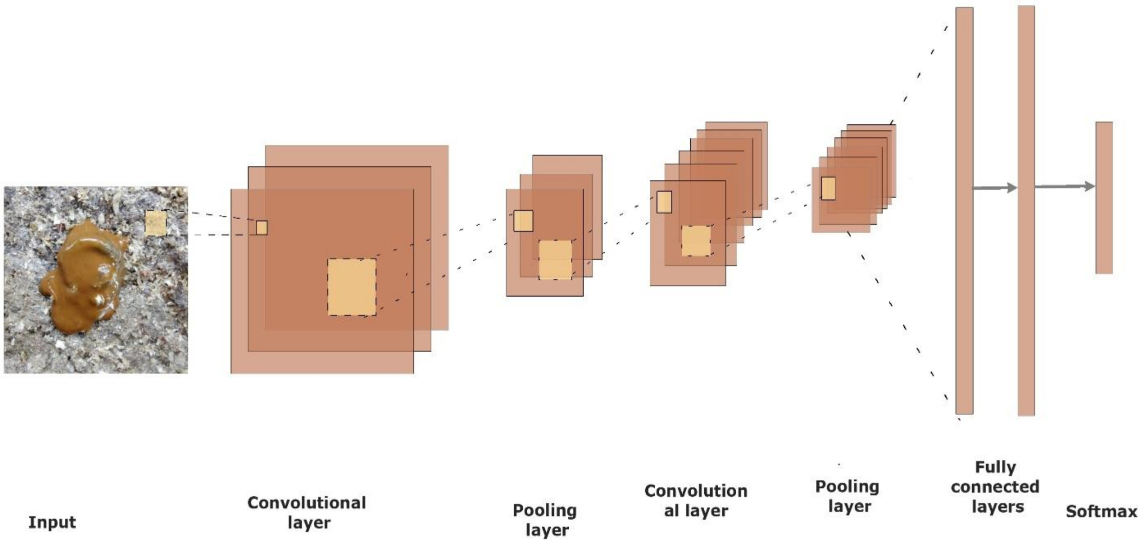


Figure 1: Layers used to build Convolutional Neural Networks

2.4 Convolutional Layer

A convolutional layer is the primary building block of a neural network; mathematical operation convolution is applied to the input data using a convolution filter/kernel to produce a feature map as illustrated in Fig. 2 and Fig. 3. The convolution are named based on the shape of the filter either 3*3 or 5*5, most commonly used is the 3*3 convolution. The filter is slid over the input at each location and performs element-wise matrix multiplication and sum the result and store it in the future map. The convolutions are performed in 3D and images are presented as a 3-dimensional matrix with dimensions of height, width and depth, whereby depth matches to colour channels (RGB). In this layer different operations take place including Non-linearity, Initialization of weights and Strides and padding.

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

1	0	1
0	1	0
1	0	1

Figure 2: Input to the convolution layer and convolution filter

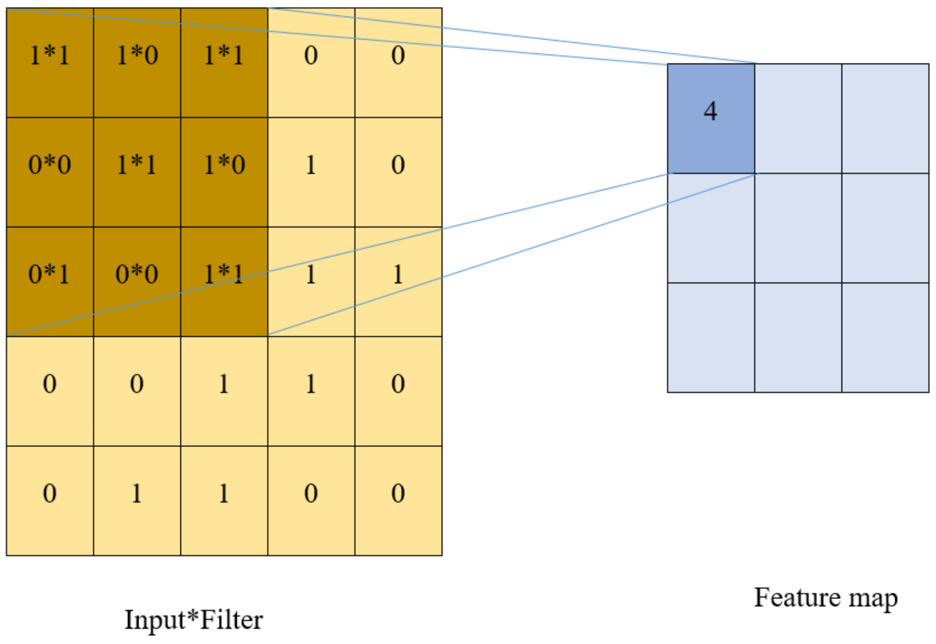
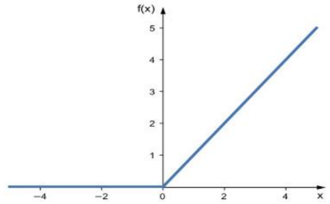
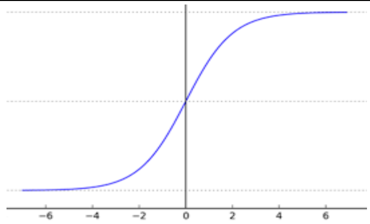
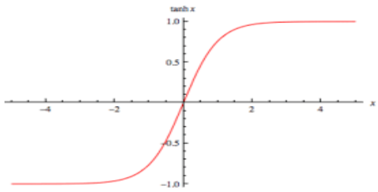
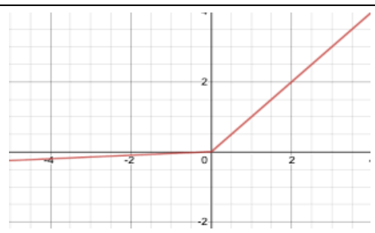
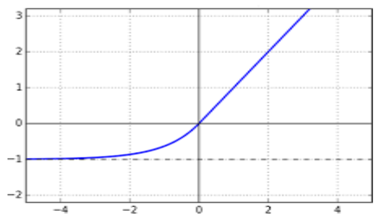
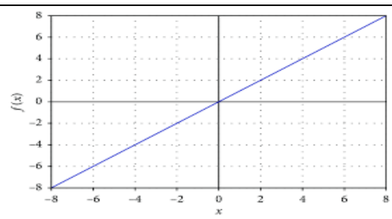


Figure 3: Convolution operations and output

2.4.1 Non-linearity

Non-linearity is a technique of ensuring that the output at any point in a neural network is not reproduced from a linear function of the input but a non-linear function. To achieve this, the activation functions are used whereby the results from the convolution operation pass through rectified linear activation function (ReLU). There are different activation functions including sigmoid, tanh, linear, ReLU, LeakyReLU and ELU. The sigmoid function is between 0 and 1 and used for models that predict the probability, tanh function is used for regression problems and, can also be applied to classification between two classes. In this study ReLU is used because it has a non-negative activation which leads to mean activation being more significant than zero (equal to 1). Also, it is easy to evaluate as compared to others. In the last layer of the output, softmax activation is applied since the output is a multi-class classification. Table 1 summarizes the activation functions with their mathematical equations and plots.

Table 1: Activation functions

Name	Equation	Plot
ReLU	$f(x) = \max(0, x)$	
Sigmoid	$f(x) = \dfrac{1}{1 + e^{-x}}$	
tanh	$f(x) = \tanh(x) = \dfrac{2}{1 + e^{-2x}} - 1$	
LeakyReLU	$f(x) = 1(x < 0)(ax) + 1(x >= 0)(x)$	
ELU	$f(x) = \{xx > 0a.(ex - 1)x <= 0\}$	
Linear	$f(x) = x$	

2.4.2 Initialization

Initialization of weights is a process whereby the weight directions for the neurons are established. Without this process, the training process cannot proceed; there are different types of weight initialization methods, including zero initialization and random initialization. Initializing weights starting with zero may lead to no progress of the neural network. On the other hand, randomly initializing of the weights can also lead to a neural network's gradient demolishing and loosing off, since the value of the vectors assigned maybe 0 or any other large amounts. Due to the short backs, this led to the introduction of initialization functions Xavier and He to optimize model performance.

2.4.3 Strides and Padding

Strides specify the number of steps a convolution filter is moved in the input. In most cases, 1 is used, but also more significant strides can be used though they result in smaller feature maps. Padding can be applied to the input to maintain the same dimensionality with the feature map, as shown in Fig. 4 and Fig. 5. When padding zeros are added at the edges surrounding the input hence preserving the size and avoiding shrinking of layers. The dimensions of the width and height are maintained; only the depth increases.

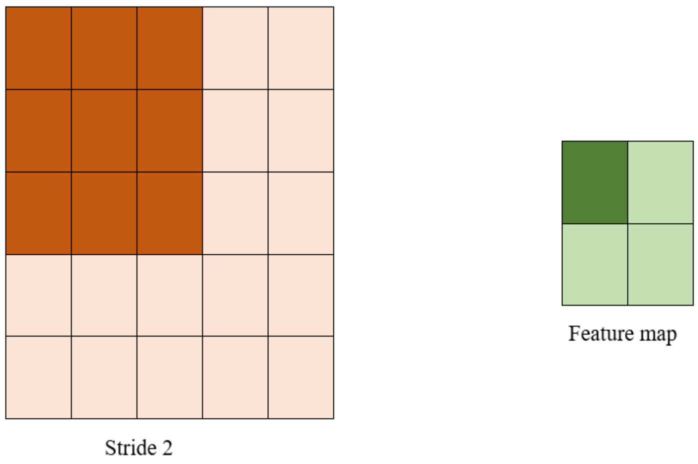


Figure 4: Convolution filter with no padding

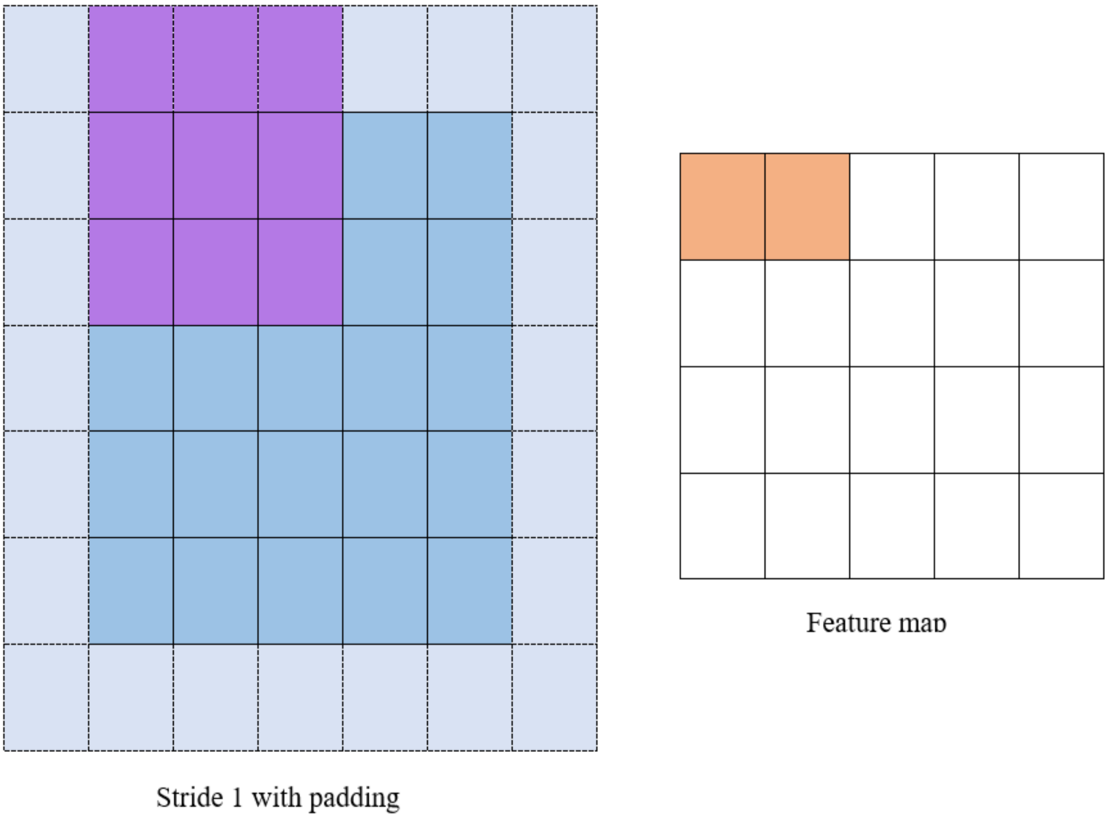


Figure 5: Convolution filter with padding

2.5 Pooling Layer

In this layer, pooling operation takes place whereby the number of parameters is reduced to shorten training time and as well as reduce overfitting. There are different types of pooling, max pooling, average pooling and global pooling. Max pooling selects the highest value in the input set while average pooling selects the average value; hence both have no specific parameters. The pooling operation also used strides sliding over the window in the input set; as a result, window and stride configuration is equal to half the magnitude of the feature map. For instance, before pooling the size of a feature map is 64*64*20, this will be 32*32*20 after pooling. The depth dimension doesn't change. It is normally operated with 2*2 windows, stride two and no padding. Figure 6 illustrates the pooling operation.

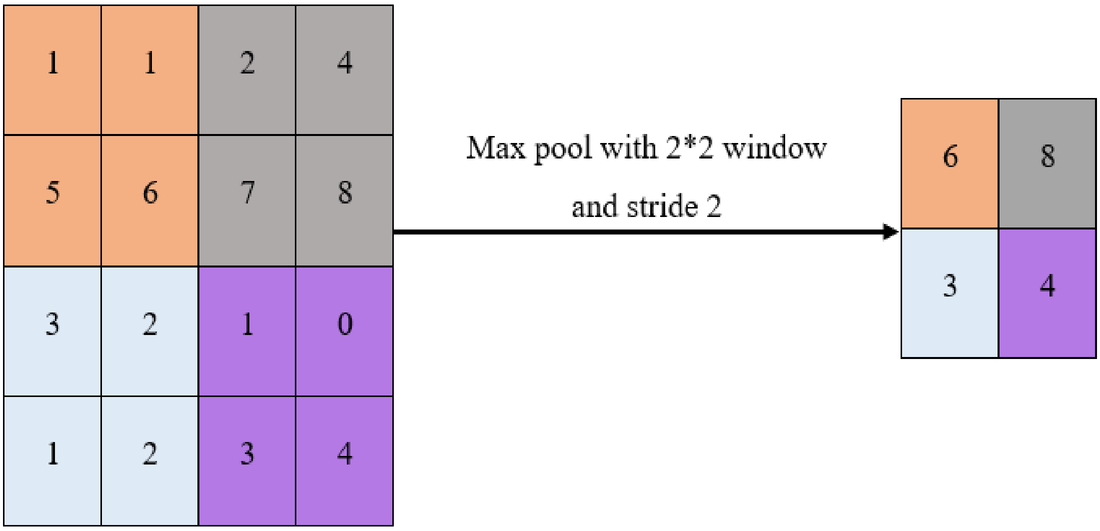


Figure 6: Max pooling operation

2.6 Fully Connected Layer

The fully connected layer is added at the end of the architecture, its input is the output of both convolution and pooling layers which is in 3 dimensions. In contrary, the fully connected layer needs input in 1 dimension. Hence, the output from the pooling layer is flattened. The mechanism behind is simple; the elements in 3 dimensions are arranged in 1 dimension.

2.6.1 Dropout

For deep neural networks, dropout is the utmost regularization technique applied to hidden layer nodes of the neural network. Also, state-of-the-art models with 96 % accuracy get a 2% boost by just incorporating dropout, which at that stage is a reasonably significant benefit. Dropout is used to avoid overfitting; a neuron is deactivated at each iteration during the training period. Inputs to this neuron and outputs from this neuron are frozen as illustrated in Fig. 7. A probability value of 0.5, which is a dropout rate hyperparameter (*p*) is considered at each step of training the network. Dropout makes a neuron to be active in one step and can be inactive at the next step since the released nodes variate in every step when training the network. The advantage is to enable the network not to depend entirely on a few neurons instead each neuron to function self-reliantly.

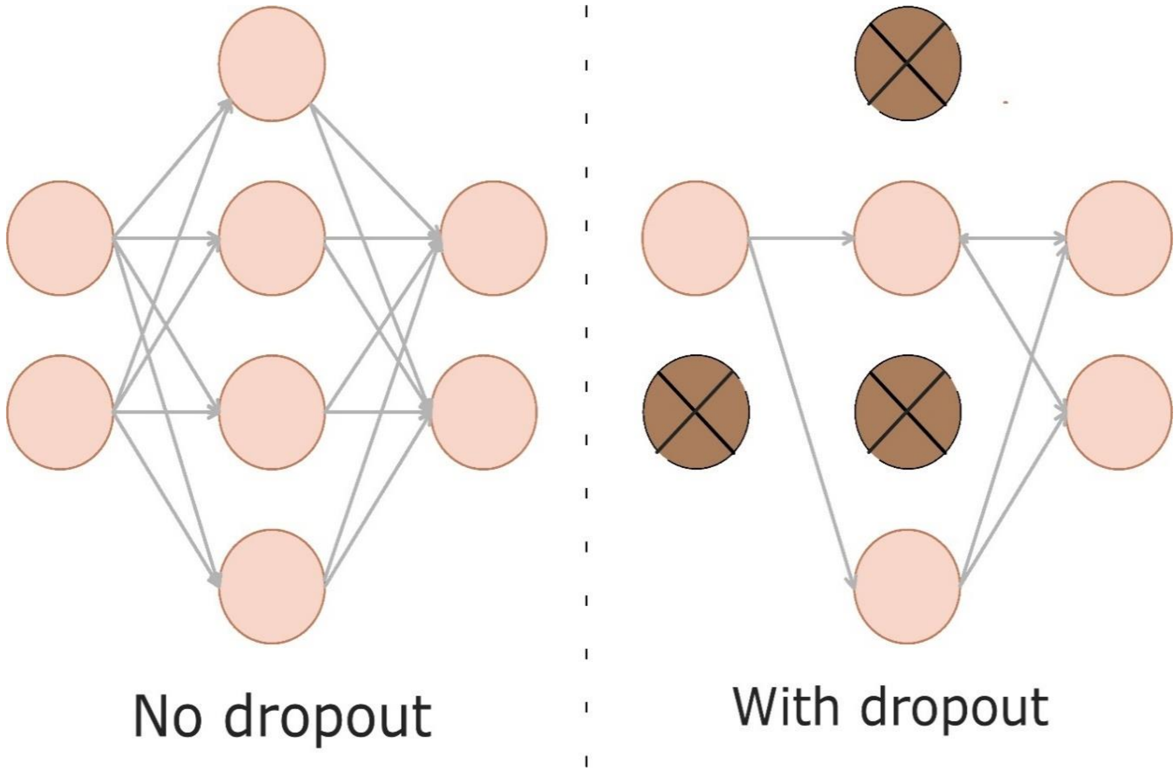


Figure 7: Dropout visualization

2.7 Transfer Learning

Transfer Learning (TL) refers to the use of a known model previously trained in a known data set to another application in the same Machine Learning domain. For computer vision, transfer learning is widely used in different applications. In computer vision, the most common pre-trained models include VGG 16, Resnet, Inception Net, MobileNet, XceptionNet and many others. The idea of using pre-trained models makes a considerable revolution in the field of Machine Learning and Artificial Intelligence. There are two main advantages when using TL; first, it performs well on both large and smaller datasets. Secondly, it is easy to reduce overfitting of the model with larger dataset when it is applied to the pre-trained model (Lee *et al*., 2019).

2.8 Fine-tuning the Pre-trained Networks

Since the study uses a pre-trained network architecture, a technique whereby the model weights are either fine-tuned in all the layers, some of the layers or some higher layers in the network is applied. In this case, fine-tuning is applied in all layers. This allows more features to be learned considering the size of the dataset.

2.9 Convolutional Neural Networks Architectures

2.9.1 Visual Geometry Group Architecture

The Visual Geometry Group developed the VGG16 and 19 architectures. The numbers 16 and 19 refer to the number of weighted layers in each network. It consisting of blocks with incremental convolution layers with 3*3 size filters, pooling layer, fully connected layers as indicated in Fig. 8 with 4096 channels on each and an output layer with 1000 channels. It also has a high number of parameters of about 144 million weights. The VGG architecture is trained on a large dataset of different images taken from multiple sources, then labelled in 1000 classes. During training, the architecture is fed with an input size of images in 224*224 pixels size followed by augmentation of the training set and random initialization was applied. The parameters used are batch size, momentum and regularization technique used is a dropout. During testing the architecture was also compared to other models and proved to outperform (Simonyan & Zisserman, 2014). The VGG architecture has been used in previous studies and portrayed high accuracy in classification problems, research by Iorga and Neagoe (2019) used transfer learning with pre-trained VGG architecture and attained an accuracy of 87%. Researchers Wulandari *et al*. (2019) also used VGG 16 and 19 models and gained an accuracy of 90% and 92% respectively.

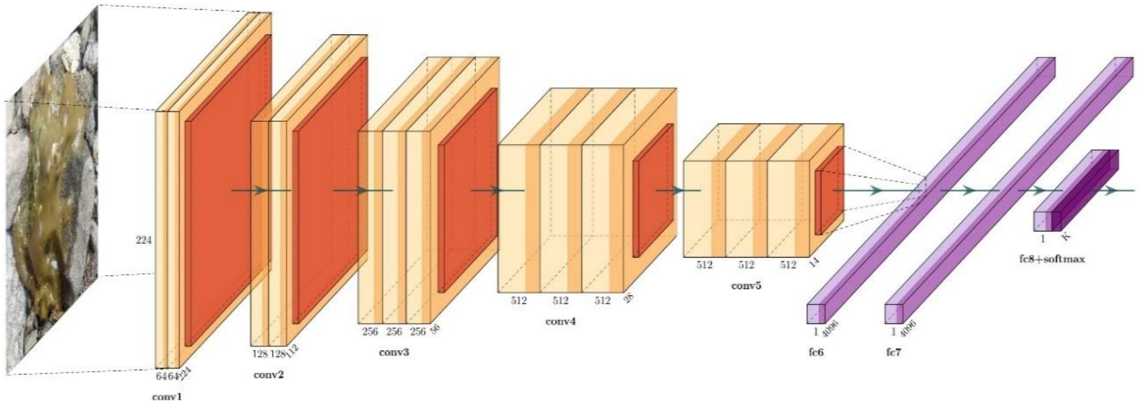


Figure 8: Visual Geometry Group (VGG) 16 Architecture

2.9.2 Resnet Architecture

Resnet architecture won first place in the classification competition in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2015. It is built up with residual connections that are networks within networks, as shown in Fig. 9. Different Resnets are available varying in a number of layers, 18-layer, 34-layer, 50-layer, 101-layer, 152-layer commonly used is the Resnet 50 that consists of 50 layers. Similar to other CNN architectures, it is comprised of the convolution, pooling and fully connected layers stacked in each residual block. During training, Xavier initializer was used with parameters batch size, SGD optimizer, weight decay and learning rate. Even though it has more layers than the VGG, it consumes less memory when training (He & Sun, 2015). Residual networks have been applied in some studies and have confidentially gained satisfactory accuracy (Yue *et al.*, 2020) compared VGG models and the Resnet, the Resnet accuracy was 87%.

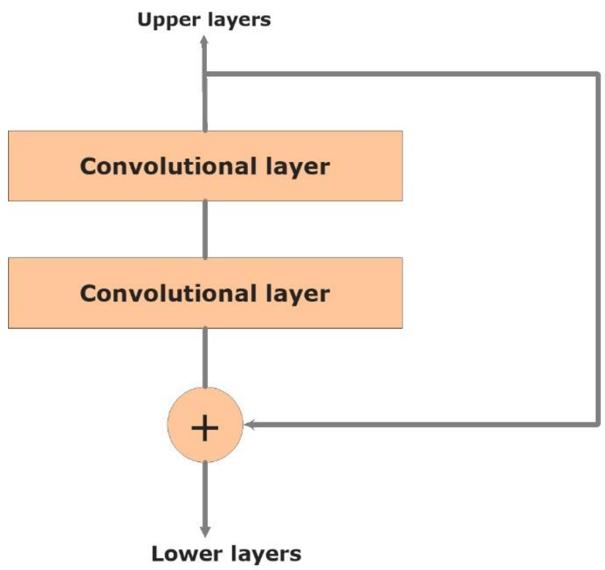


Figure 9: Resnet basic building block

2.9.3 MobileNet Architecture

MobileNet is an efficient architecture for models deployed in mobile devices. It consists of trivial separable layers of neural networks created from depthwise distinguishable convolution filters. The mechanism behind each input network is from one convolution filter that is 1*1 convolutions (Wang *et al*., 2020). The regular operation of convolution has the effect of filtering features that are focused on convolutionary kernels and which combine characteristics to create new representations. It is possible to divide the filtering and combination phases into two steps to significantly minimize computational convolutions by the use of factorized convolutions to obtain less computational rate (Howard *et al*., 2017).

2.9.4 XceptionNet Architecture

Xception architecture was adapted from inception, it optimizes the convolutions in inception so that they consume less memory during training. The mechanism is that each channel has one kernel to perform convolution, 1*1 convolutions capturer cross-channel connections, as illustrated in Fig. 10. Correspondingly, spatial relations are captured trough the standard 3*3 or 5*5 convolutions. From the architecture, XceptionNet as a lightweight network has a lesser number of parameters and a more substantial classification impact (Chollet, 2017).

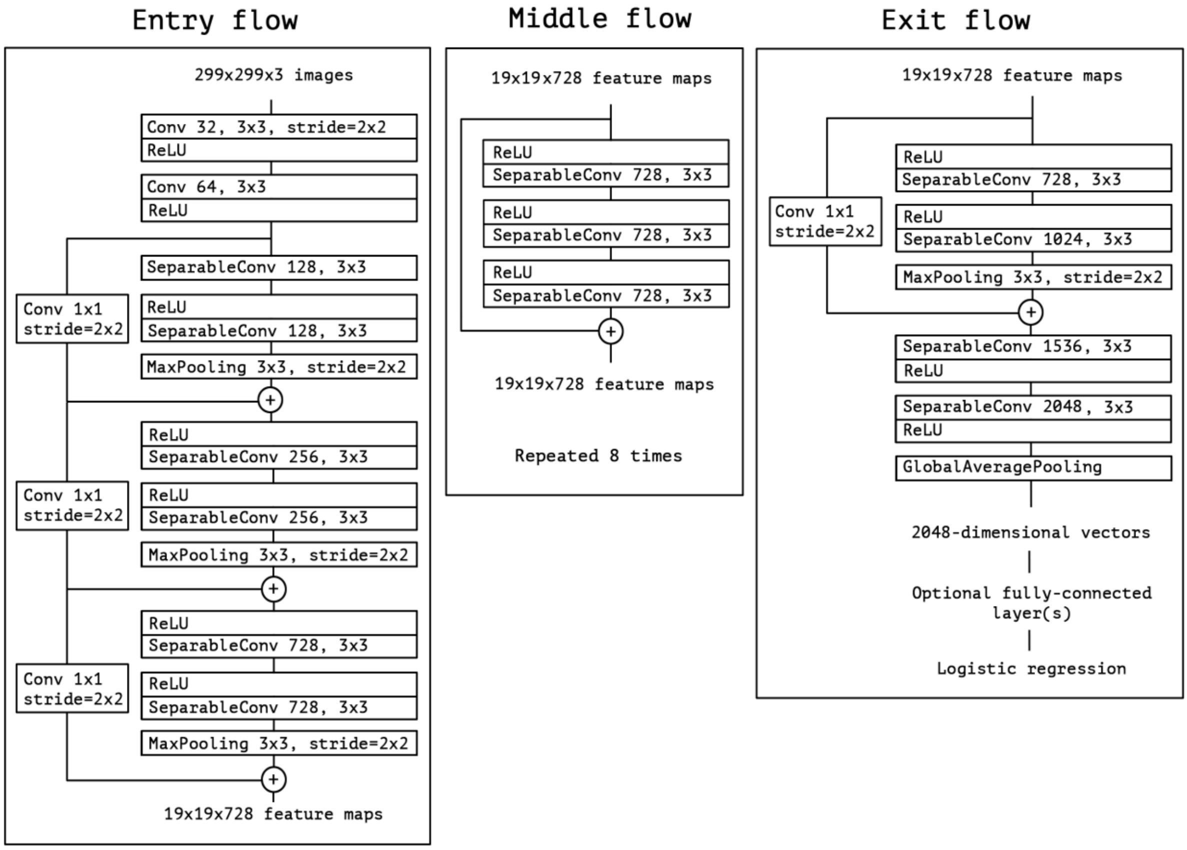


Figure 10: XceptionNet Architecture (Chollet, 2017)

2.9.5 The Convolutional Neural Network Architecture

The proposed Convolutional Neural Network (CNN) architecture involves stacking of multi convolution layer, and the whole architecture is given in the Fig. 11. In the first layer, images with the size 224*224 or 512*512 are fed to the stack of convolution layer as the input. The convolutional layers have filters with the small receptive field of 3*3 and are followed by max-pooling layer, which operates over a 2*2 pixel window. These layers form a single block, and repeatedly apply the block by increasing the depth of filters in the network in the following order 32, 64, 64, 128, 128, 256, 256, 512 for the full convolution blocks. In each block, the same padding is applied to ensure that the height and the width of the output features match the input features. Rectified Learning Unit (ReLU) activation is used for all layers; meanwhile, He uniform is considered for weight initialization for all blocks.

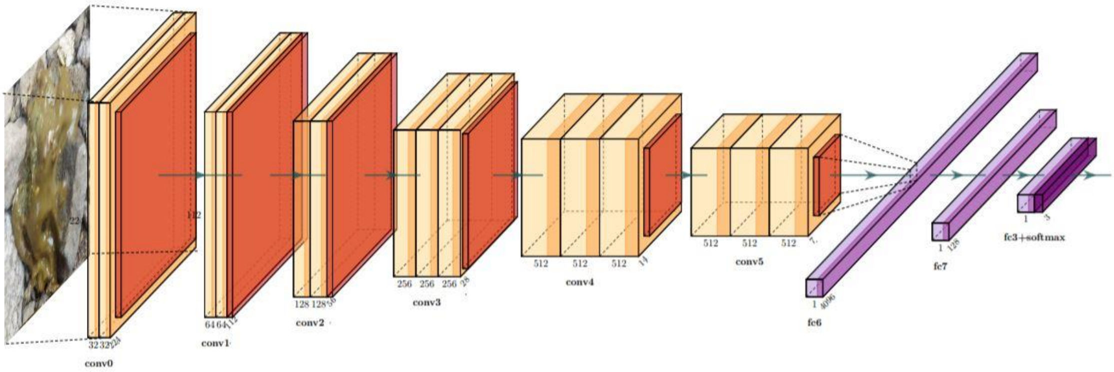


Figure 11: Developed Convolutional Neural Network Architecture

2.10 Convolutional Neural Networks in Agriculture

2.11 Related Works

Deep Convolutional Neural Networks (DCNN) enable the computer to interpret captured data objects (feature extraction and representation) for classification, localization, and recognition to be automatically learned (Kumar *et al*., 2018). Data object recognition involves predicting the region where the objects are present and classifying all items in the image, object localization indicates the area of the image where the main object is current (Ning *et al*., 2017). In contrast, image classification classifies the main object in the image (Wang *et al*., 2020). It includes various stages of representation so that each step can be progressively transferred to learn complex features (Makantasis *et al*., 2015).

Deep Convolutional Neural Networks have been used for early disease detection in both plants (crops) and animals; the approach has worked well in contrast to traditional machine learning approaches that are constrained in processing (Ferentinos, 2018). A study by Kumar *et al*. (2018) to identify cattle based on the images of the muzzle points used a Deep Learning (DL) framework and proved to be useful in recognition. In the study, a dataset of 5000 images was collected and use to develop a deep learning model with an accuracy of 95%.

In plants, a study by Owomugisha *et al*. (2014) focused on early detection of banana bacterial wilt and banana black sigatoka diseases caused by bacteria. The research involved computer vision techniques mainly image classification with several classifiers including Support Vector Machine (SVM), Naïve Bayes, Random Trees and others. A dataset of 630 images consisting of both healthy and infected banana leaves was collected. The data was then augmented and used to train the classifiers that obtained a high accuracy of 96% and 91% for the two diseases respectively. Similarly, Ferentinos (2018) used leaf images to train a model for disease detection in different crops, the study had a dataset of 87 848 images from healthy and diseased plant leaves. The developed CNN model had a 99.53% accuracy. Ramcharan *et al*. (2017) used DCNN to identify five cassava diseases, a dataset comprising of 11 670 images was created and transfer learning was applied to train the model with a maximum accuracy of 93%.

In poultry, Machine Learning (ML) is deployed in various studies; one study aimed to identify problems in egg production in using an SVM approach. Morales, Cebrián, Fernandez-Blanco and Sierra (2016) recorded egg production data in a period of 12 months; relevant features were used to train the classifier and attained a 96% accuracy. The authors conclusively suggested that the approach was suitable for detection compared to classification. Similarly, Zhuang *et al*. (2018) used SVM to develop an algorithm for identification of broilers with bird flu. The study based on several features targeting the postures of the chicken both healthy and sick. Data of chicken images was collected from real time environment whereby they were isolated and inoculated. The research analysed 200 images for training and 500 images for testing the model and the overall accuracy was 99%.

Felip *et al*. (2014) also used neural networks comparing them with other models for prediction of egg production in poultry. In the study, quails were isolated and monitored for 35 days with different features like weight and number of eggs produced considered. The study concluded a Bayesian network as the best for prediction. Another study by Hepworth *et al*. (2012) used Support Vector Machine (SVM) approach to detect broilers that where healthy or infected based on the levels of hock burn from data gathered in a three year period. Variables of hock burn levels either high or low where recorded then processed to train a SVM classifier and had a 78% accuracy, the model to have worked well with the detection. The CNN was used to identify and recognize *Clostridium perfringens* disease in birds based on the sound they make (Sadeghi *et al*., 2015), 30 birds were monitored for 30 days and different features were recorded. Five features were selected to train the model, had an overall accuracy of 95% proving to work efficiently, and can be used for early detection of the disease.

Likewise, Zhang *et al*. (2020) trained an improved Resnet model to detect sick chicken using chicken images and the model reached an accuracy of 93.7% on test set prediction. A study by

Pu *et al*. (2018) developed a system based on CNN for identification of chicken behaviour. The model was trained on a dataset of both healthy and sick broilers and later compared to other models where it outperformed them with a 99% accuracy. Okinda *et al*. (2019) developed a model to distinguish healthy and unhealthy chicken monitored in a controlled environment considering features like mobility and shape. The model had an accuracy of 97% outperforming other models tested on the test dataset. In the same way, Zhuang and Zhang (2019) proposed a method to determine healthy status of broilers using image processing based on deep learning. The model was trained on a dataset of both healthy and sick broilers and later compared to other models where it outperformed them with a 99.7% accuracy.

From the literature above few studies have been conducted on the detection and classification of diseases using classical machine learning approaches like SVM, Decision Tree and Random Forest. In particular, different studies have focused on poultry diseases identification using DL. These studies showed promising results, but most of them detected a single disease or performed their methods via laboratory experiments or by used features of images found in online repositories that do not reveal the natural features in the setting. Therefore, this study proposes a model for early detection of three poultry diseases Coccidiosis, Newcastle and Salmonella using faecal images data from both natural infections and inoculation for poultry farmers in Tanzania.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Data Collection

Data was collected for a period of five (5) months, from February 2020 to July 2020. Poultry farms were identified with the help of the veterinary officers in the regions. Different smallholder farmers and inoculation sites were visited in Arusha, Kilimanjaro and Manyara regions. In each farm, researchers had a short discussion with the farmer concerning the poultry farm information to understand better the management practices in particular poultry diseases. The information included the age of the chickens, common conditions that frequently affect the chicken and how they decide on the medication to offer the sick chicken. Figure 13 shows the researcher collecting data (image and sample) in the field.

Figure 12: Data collection in the poultry farm

Open Data Kit (ODK) tool was used to collect the poultry farm information. Open Data Kit is open-source software for organizing and managing data that is collected from the field. The tool was designed with different interfaces as illustrated in Fig. 14. The ODK was used because it can operate in resource-constrained environments, i.e. data can be collected offline. The information and data were both stored in Google Drive.

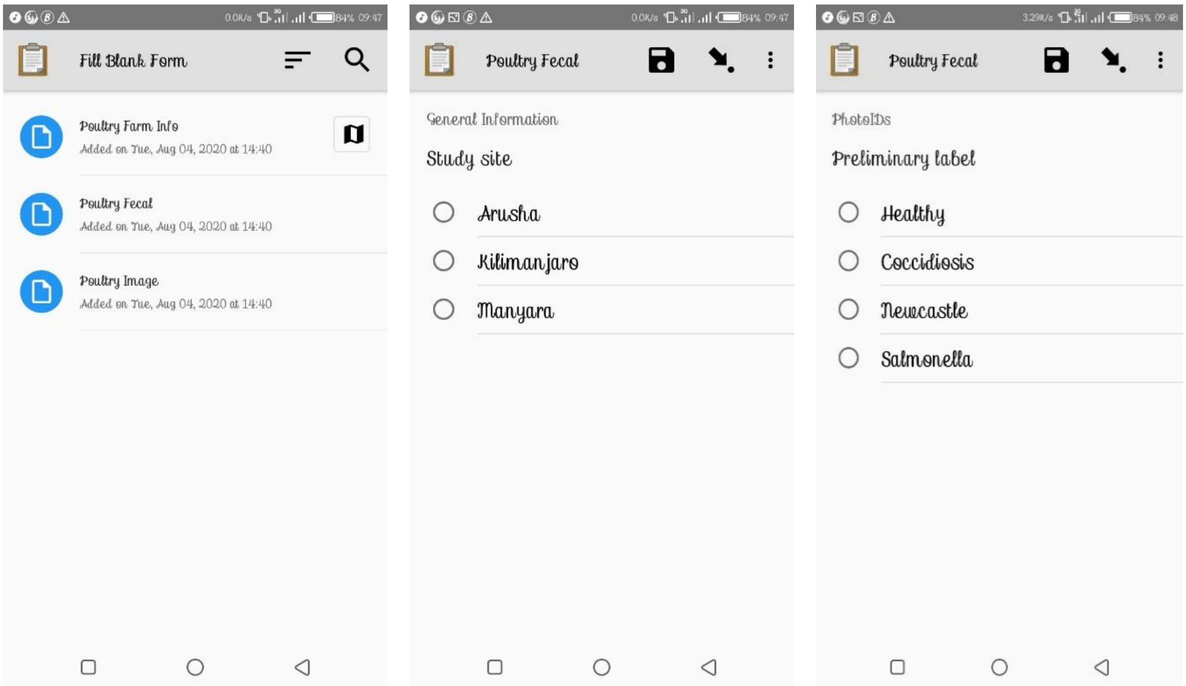


Figure 13: Open Data Kit data collection interface

This study collected data in the form of images, of faecal samples from chicken. Devices used in the activity were smartphones of different types with the following specifications:

(i) Huawei GT3

(ii) Tecno CX-Air-resolution (320*240)

(iii) Samsung Galaxy A1

(iv) Infinix

Different mobile devices were used, resulting to images of different resolutions. Images of both healthy and infected chicken faecal samples were taken; the infected droppings samples were from both natural infections and inoculated infections as shown in Fig. 15.

The target data for the study was faecal images to be collected from poultry farms and inoculation sites. A total of 508 images of healthy samples, 516 of Coccidiosis, 40 of Newcastle and 566 of Salmonella a dataset of 1630 images were collected for the development of the model. The dataset was considered adequate since we were using the transfer learning approach when developing the model. Transfer learning allows training models using very little data. In addition, different augmentation techniques were applied in order to increase the size of the dataset (Zhang & Chen, 2020). The images were labelled to distinguish the four classes and were used for training the models applying transfer learning (Ramcharan *et al*., 2017).

(a) (b) (c) (d)

Figure 14: Sample images from the faecal image dataset (a) Coccidiosis, (b) Health, (c) Newcastle, (d) Salmonella

3.2 Research Pipeline

This is the framework that clearly illustrates the research plan. Figure 16 illustrates the steps and activities to follow when developing the model. First, collected data (faecal images) and attained the dataset. The dataset is then pre-processed as it is either labelled, resized or augmented. The pre-processed data is stored in the cloud. In this work, data was stored in Google Drive. The dataset is split into training and testing set before training it on the models. In this study, data is trained on pre-trained Deep Neural Networks.

After training the proposed models on the data and achieving an optimized model, it is then visualized in a web browser. Dash and flask frameworks are used to create the dashboard that the model will be demonstrated. Dash is a Python framework used to deploy a model and create a test application on the web. It runs the application locally as can be customized to process the raw data, run the predict method from the developed model then finally returns the accuracy.

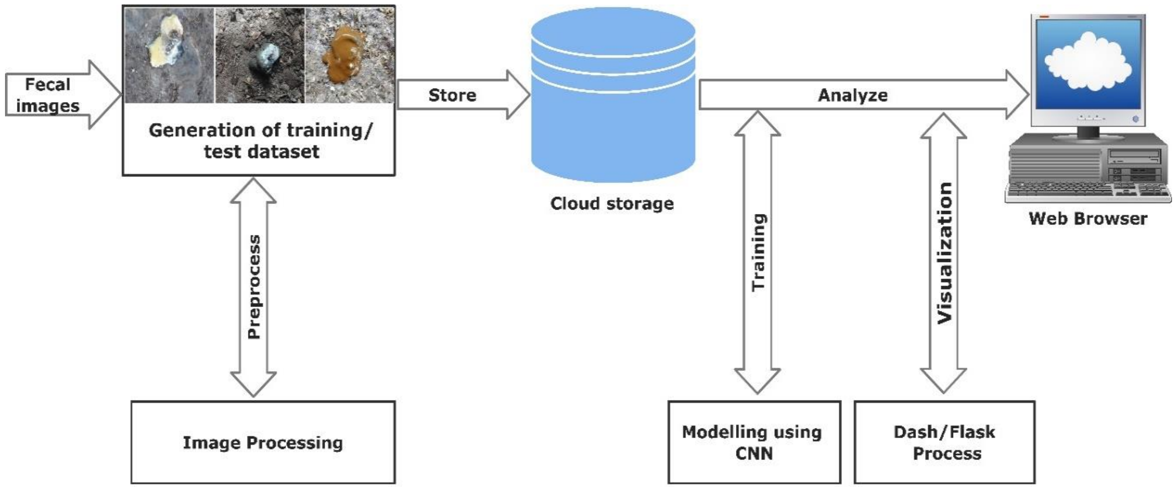


Figure 15: Research framework

3.3 Data Pre-processing

This procedure involves transforming raw data into an understandable format. Data used in this study were faecal images collected in the poultry farms. During pre-processing activity the following tasks were performed: Labelling and resizing.

3.3.1 Labelling

The images were labelled corresponding to the four classes health, Coccidiosis, Newcastle and Salmonella. The images were labelled because the objective is to identify and classify the diseases early; this falls under a supervised machine learning category. In supervised learning, the inputs and outputs are known. In this work, the inputs are correctly labelled images of the faecal samples. The labelling task was done manually by the help of the veterinary officer.

3.3.2 Resizing

The dataset comprised of images in a Joint Photographic Group (JPEG) format, with different sizes. The images were resized to a uniform pixel 224*224 and 512*512 based on the requirements of the proposed method.

3.4 Tensor Records Generation

Tensor records format is used for storing a sequence of binary records. The binary file contains sequences of byte strings. In this case, data needs to be serialized whereby it is encoded as a byte string before being written into a tensor flow record. This study uses an image dataset therefore, converting the dataset into tensor records includes serializing images (writing tensor records), this involves different encoders and should correspond to the decoder that will be used during deserialization (reading tensor records). Since a Tensor Processing Unit (TPU) environment is used to train the models, jpeg encoding is applied. It compresses the data and improves data transfer time. Then, the images were resized into 224*224 and 512*512 resolutions as shown in Fig. 17. This is done to achieve a reasonable resolution to train the models and get the best performing model since the model will be deployed on the mobile phone.

Resolution	Coccidiosis	Health	Salmonella
224*224			
512*512			

Figure 16: The sizes of images in the dataset changed to 224*224 and 512*512

3.5 Augmentation

Augmentation of data is a way of generating further training data from a present set. By generating new examples by a random transformation of existing ones, it supplements the training data. This way, it increases the size of the training set artificially, decreasing chances of overfitting. Overfitting occurs when there are too few instances to train, resulting in a model with low efficiency of generalization. Augmentation is performed dynamically when training the model producing accurate images referring to the faecal image dataset. Augmentation is not performed on the validation and test sets.

As humans, it can be deduced, so the model should be able to learn that as well. The primary benefit of augmentation is increasing the size of the training data up to 50 times more. It's a very powerful approach which is used in any single deep learning model based on image data (Dawud *et al*., 2019). It involves different transformations like flipping, rotation, greyscale, saturate, centre crop, padding and many others. In this work, the transformation chosen was suitable for the kind of data involved (images). Therefore, image flipping, image cropping, padding and saturation were used as follows:

(i) Image flipping: It is a technique to turn over the image either vertically or horizontally.

(ii) Image cropping and padding: It involves resizing an image to a target width and height by either centrally cropping the image or padding it evenly with zeros.

(iii) Image saturation: Is a method that converts coloured images to float representation, converts them to hue, saturation and value scale adds an offset to the saturation channel converts back to coloured then back to the original data type.

3.6 Environment Setup

3.6.1 Baseline Models

The baseline model VGG 16 for Keras was trained in a Desktop computer pre-installed with Ubuntu 18.04 and Anaconda set up with a TensorFlow backend. The Resnet 50 in Google Collaboratory environment with a Fastai Library.

3.6.2 Training the Baseline Models

Two baseline models, VGG 16 for Keras and Resnet 50 were trained on two classes Health and Coccidiosis. The 508 images for health and 516 images for Coccidiosis. In training the models, the dataset was split 80-20, 80% of dataset for training and 20% for validation respectively. From a dataset of 1024 images, 818 images for training and 206 images for validation were obtained for the two classes as shown in Table 1.

Table 2: Dataset splitting for training baseline models

Class	Image in each class	Training	Validation
Health	508	406	101
Coccidiosis	516	412	103

Training the baseline models included different hyperparameters learning rate, epochs, batch size and optimizer including:

(i) Learning Rate: This parameter controls how much to change the model in response to the estimated error each time the model weights are updated.

(ii) Epochs: The number of rounds a model takes to train a batch of data.

(iii) Batch size: The number of samples that are propagated through the network in one round.

(iv) Optimizer: Is the extended class, which includes added information to train a specific model. Some examples of commonly used optimizers are SGD, Adam, RMSprop.

Table 2 and Table 3 summarize the hyperparameters used when training the baseline models VGG 16 for Keras and in Resnet 50.

Table 3: Hyperparameters used when training Visual Geometry Group (VGG) 16 baseline model

Parameter	Value
Epochs	50
Batch size	16
Optimizer	rmsprop

Table 4: Hyperparameters used when training Resnet 50 baseline model

Parameter	Value
Epochs	50
Batch size	16
Optimizer	SGD
Learning rate	1e-3
Weight decay	1e-9

3.6.3 Evaluation of the baseline models

The baseline models were evaluated based on their performances. The accuracy metric was used when evaluating the models. The training accuracy, training loss, validation accuracy and validation loss were observed during training, from the 1st epoch to the 50th epoch. The metrics include:

(i) Training accuracy: Is the accuracy achieved when the model is applied to the training data.

(ii) Training Loss: This is the error on the training set of the data.

(iii) Validation Accuracy: This is the accuracy used to evaluate the model's performance.

(iv) Validation Loss: This is the error after running the validation set of data through the trained network.

3.7 Proposed Model

3.7.1 Developed Convolutional Neural Network Model

In this work, a fully developed from scratch CNN model is proposed and trained on the dataset. Initially the models are trained on three classes Health, Coccidiosis and Salmonella. A total dataset of 1590 images, 508 health, 516 Coccidiosis and 566 Salmonella. The dataset was split 60-20-20, 60% of dataset for training, 20% for validation and 20% for testing respectively. The data was split in three sets because during training phase, the model is validated based on the training set and may lead to overfitting of the model in the validation set. The test set (unseen data) makes sure that the model generalizes the problem well. Table 5 summarizes how the dataset was split.

Table 5: Dataset splitting training, validation and test sets

Class	Image in each class	Training	Validation	Test
Health	508	305	102	101
Coccidiosis	516	310	104	103
Salmonella	566	340	114	113

The training was performed in Kaggle Environment, accelerator used was TPU-v3.8 using Python language v3.7, and Tensor Flow was the backend. TensorFlow framework was used because it supports models to be deployed and interpreted in other devices. Table 5 below summarizes the hyperparameters used when training the CNN model. During training, normal stochastic gradient descent is used to minimize the error, and in evaluation, log loss and accuracy as metrics were levaraged. The findings show that some of the images from the dataset may contain more than one disease; hence categorical cross-entropy loss function seems to fit the problem with the log loss as an evaluation metric.

An output layer with three nodes and softmax activation is used. Since the problem is multi-class classification, softmax is the right choice because the output from the node is the likelihood for the output to be either of the three classes. The schedule learning rate is used in the experiment with some call back features. Call backs are used in order to store the best performing model ensuring that the test set accuracy is equal or greater than the validation set accuracy. When training the CNN model, a number of epochs is set to 500 which is a large number of iterations because the developed dumb converging neural network with no weights is dumb. The He uniform initializer is then used to random initialize the weights during training. After training the CNN model, transfer learning was applied and train the dataset on pre-trained models.

Table 6: Hyperparameters for training the Convolutional Neural Network model

Parameter	Value
Learning rate maximum	0.000012
Learning rate minimum	0.00001
Learning rate attained at an epoch	200
Learning rate exponential decay rate after 10 epochs	0.6

3.7.2 Pre-trained Models

The study proposed different pre-trained models suitable for the classification task based on the architectures reviewed in Chapter 2. The study trained the VGG- 16, MobileNet, Resnet 50 and XceptionNet models on the collected dataset as split in Table 4. The models were also trained in the same environment as the CNN model. Table 6 summarizes the hyperparameters used when training the models.

Table 7: Hyperparameters for training Pre-trained models

Parameter	Value
Learning rate maximum	0.000012
Learning rate minimum	0.00001
Learning rate attained at an epoch	10
Learning rate exponential decay rate after 10 epochs	0.8

3.7.3 Evaluation of the Pre-trained Models

The pre-trained models were evaluated based on different metrics. Evaluation metrics used were accuracy, precision, recall, F1 score and log loss as explained below:

(i) Accuracy: It calculates how often predictions are equals to labels.

(ii) Precision: Number of correctly identified results divided by the number of all positive results.

(iii) Recall: Number of correctly identified positive results divided by the number of all samples that should have been identified as positive.

(iv) F1 score: It is a measure of test accuracy.

(v) Log loss: This is a classification function often used as an evaluation metric often used in multiclass classification. It quantifies the accuracy of a classifier by penalizing false classifications. The log loss was minimized by assigning a probability to each class in this problem resulting to maximizing the accuracy of the model.

The models were set to train the dataset on 50 epochs with 8 steps per epoch, resulting to 400 iterations to complete the training process. The models were trained observing the accuracy and loss trend in both training and validation sets. The logarithmic loss was used to evaluate the models' performance on the test set whether the prediction was correct. Then plotted the training and validation accuracy and loss against the number of epochs. The study presents and discusses the results in the next chapter.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Data Collected

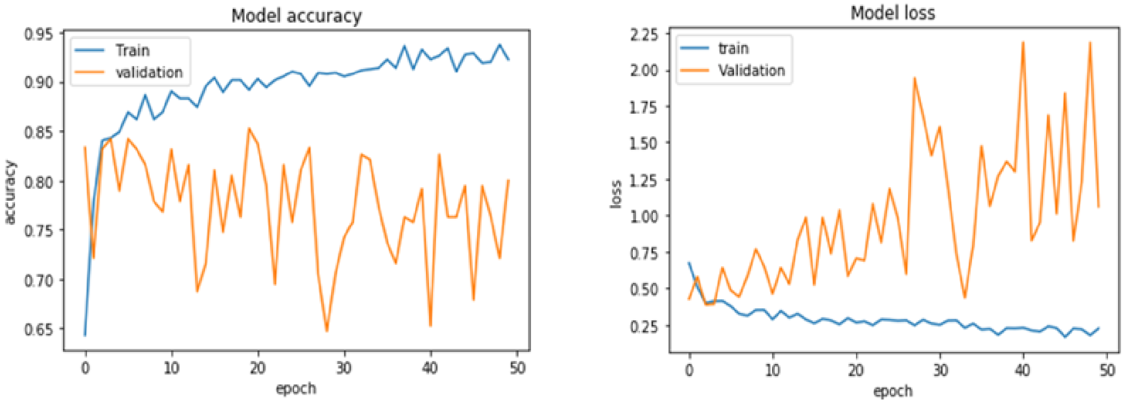
In the review, activity round table discussions were held with the veterinary officers and identified the target clinical signs of the three diseases focused in this study. The study identified the digestive clinical signs of the diseases mainly the occurrence of the chicken droppings. The work also learned how the chicken droppings can be used for early identification of the diseases thus guidance to the data collection procedure. The target was to have a dataset of 4000 images, 1000 images for each class. A dataset of 3448 labelled images suitable for classification tasks was collected. Table 8 illustrates the dataset distribution.

Table 8: Dataset

Class	Images collected
Healthy	1384
Coccidiosis	891
Newcastle	40
Salmonella	1133

4.2 Results for the Baseline Models check

The performance of the two baseline models was evaluated based on training of the dataset for binary classification. The VGG 16 for Keras and Resnet 50 models were trained on the data shown in Table 2. The metrics used were training and validation accuracy, as shown in the plots below.



(a) (b)

Figure 17: Training and validation plots for baseline models

Figure 18 (a) and (b) illustrate the trend in the training and validation accuracy, and training and validation loss while training VGG for Keras model for 50 epochs. Table 9 shows the accuracies and losses for the models. Visual Geometry Group 16 model attained an accuracy of 93.7% and Resnet 50 an accuracy of 90.7%.

Table 9: Training and validation results for baseline models

Model	Training accuracy	Training loss	Validation accuracy	Validation loss
VGG 16	0.9193	0.2346	0.9372	1.1840
Resnet 50	0.8527	0.0723	0.9077	0.3718

4.3 Results for the Proposed Model

4.3.1 Developed Convolutional Neural Network

In this study, a fully connected CNN model was trained on the dataset illustrated in Table 5. The performance was evaluated on the training and validation accuracies observing the validation accuracy and loss trend from the 1st epoch to the 500th epoch with the logarithmic loss. The validation accuracy kept increasing from the 100th epoch as the validation loss decreased. Figure 19 (a) and (b) shows the plots of the accuracies and losses against the number of epochs. Convolutional Neural Network model gained an accuracy of about 92%.

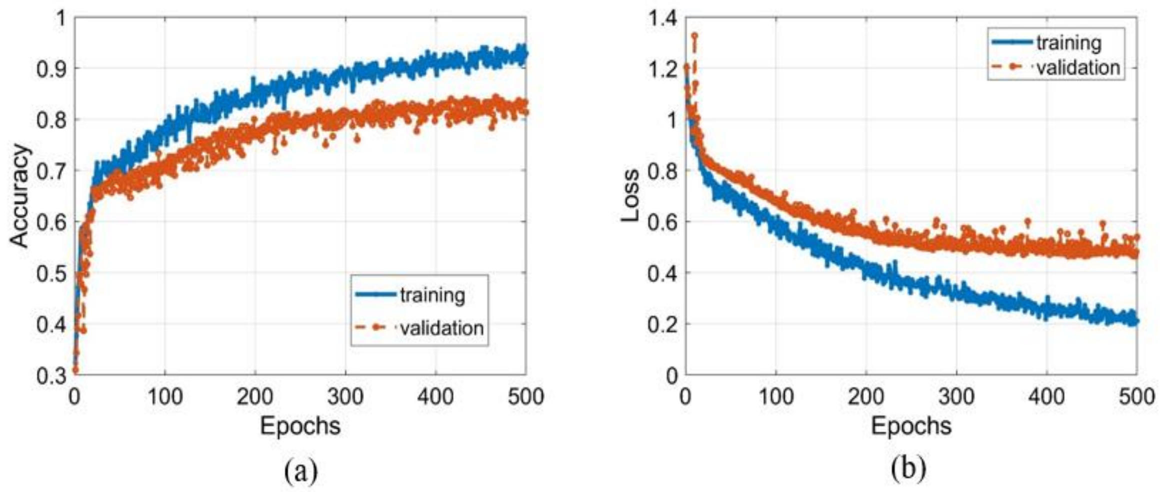


Figure 18: Training and validation plots for the Convolutional Neural Network model

4.3.2 Pre-trained Models

The pre-trained models used include VGG 16, Resnet 50, MobileNet and Xception net. These are trained on the dataset presented in Table 5. Evaluation of these models was also based on the validation accuracy and loss trends observed from the 1st to the 50th epochs with the logarithmic loss value. Table 10 summarizes the results for all the proposed models.

Table 10: Training and validation results for the proposed models

Model Name	Training loss	Training accuracy	Validation loss	Validation accuracy	Log loss
The CNN	0.5400	0.8133	0.2282	0.9267	0.20
XceptionNet	0.0494	0.9855	0.161	0.94	0.15
Resnet 50	0.013	0.9989	1.1131	0.3133	4.8
VGG 16	0.1337	0.9487	0.3522	0.8933	0.35
MobileNet	0.0971	0.9665	0.9005	0.6833	0.89

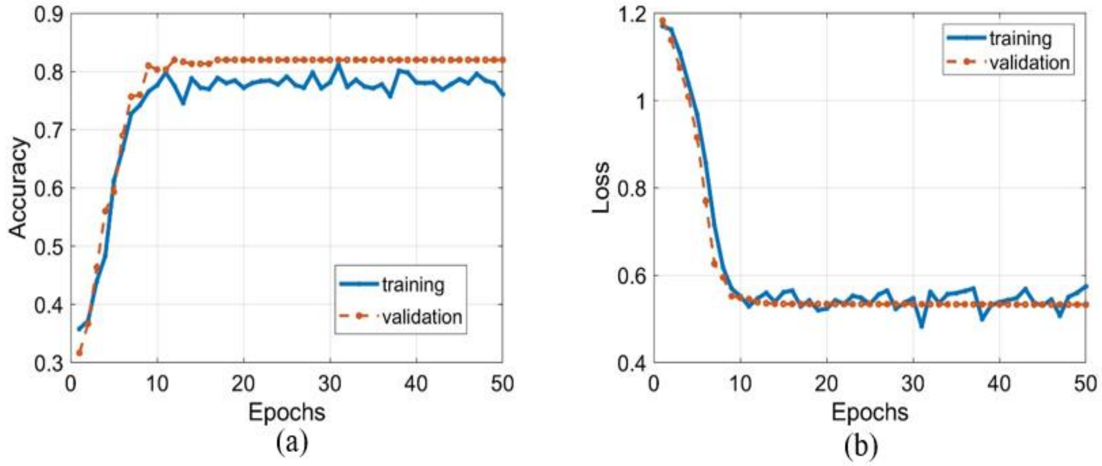


Figure 19: Training and validation plots for Visual Geometry Group (VGG) 16 model

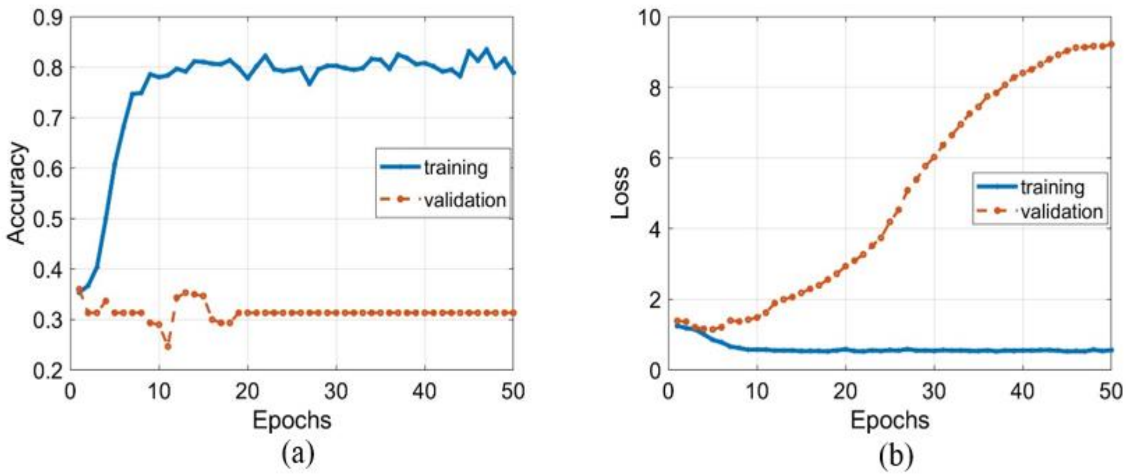


Figure 20: Training and validation plots for Resnet 50 model

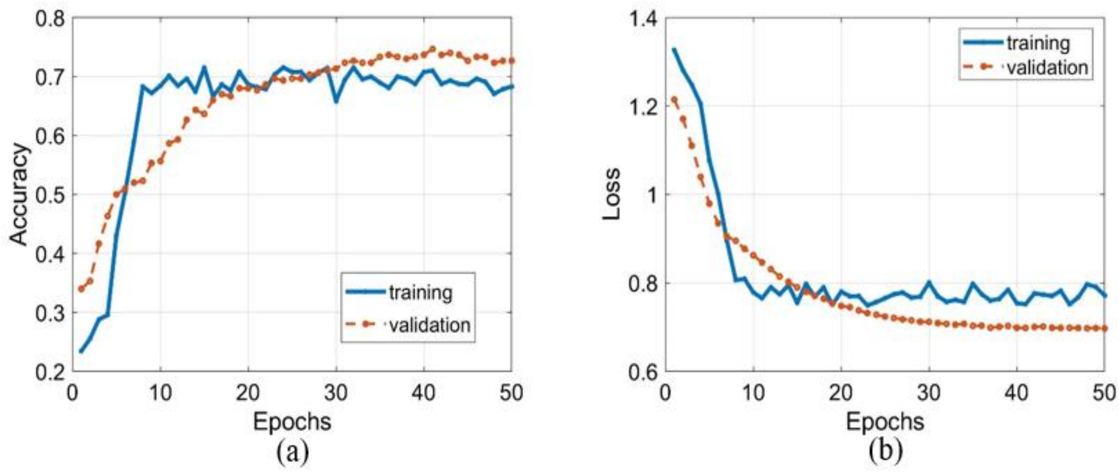


Figure 21: Training and validation plots for MobileNet model

Figures 20-22 (a) and (b) illustrate the accuracy and loss plots against the number of epochs during training for all the pre-trained models used on the dataset. The graphs illustrate the trend between the training and validation steps from the first to the last epoch. When training the VGG 16, Resnet 50 and Mobile net models there was a huge variation between the training accuracies and validation accuracies. The models appear to generalize well the training data and fail to learn the features on the test data thus the models were overfit.

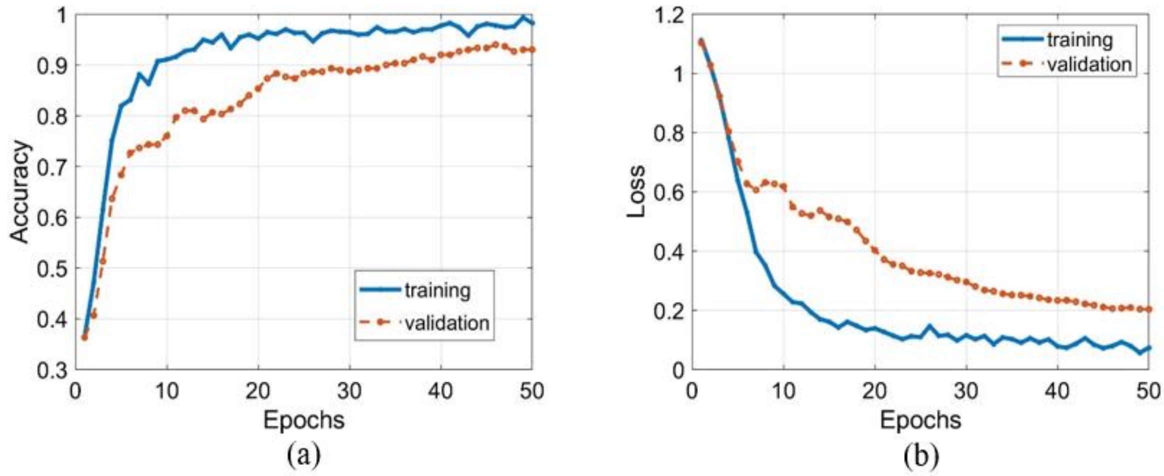


Figure 22: Training and validation plots for XceptionNet model

Figure 23 (a) portrays the accuracy trend of the training and validation of the best performing model in this work, XceptionNet model. When training the XceptionNet model, the accuracy increased from the 10^{th} epoch this shows that the model had started to learn well the features in the dataset. From the 20th epoch the accuracy trend was stable and increased gradually up to 94%, this illustrates that the model generalizes well and has fit the data thus, suitable for prediction. Figure 23 (b) shows the validation loss trend, the loss started decreasing from the 10th epoch reaching the 20th epoch, loss decreased up to the 50th epoch. The study concluded that the model was best fit.

4.4 Discussion

The study starts with training a portion of the dataset (two classes coccidiosis and healthy) binary classification on the baseline models to observe the behaviour of the data and validation of the requirements. Then proposes a fully CNN model and trains it on three classes for multi-class classification, and it achieves an accuracy of 84%. As stated earlier, the work leverages use of transfer learning and train pre-trained models on the dataset, and XceptionNet model outperforms other models with an accuracy of 94% and a log loss of about 0.15. The study also uses the logarithmic loss as a metric since the problem is a multi-class classification. In addition, it is observed that some images in the dataset may contain more than one disease hence categorical cross-entropy loss function fits this case. Comparing the time taken for single prediction, in the normal laboratory procedure it takes 3-7 days but the XceptionNet model takes about 3 seconds. Hence, conclusively the model is efficient and can be used for robust diagnosis of the diseases. Based on the metrics used and accuracies obtained by all the models used in this study, the best model is the XceptionNet with an accuracy of 94%.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The study presents the novel chicken disease detection model using Transfer Learning approach on a pre-trained CNN. The key elements that can improve the performance of the extension officers and poultry farmers in the early detection of chicken diseases are the use of computer-aided instruments and accurate data. The creation of such image processing techniques that can assist farmers is a necessity of the present era. These methods are valuable to reduce the losses incurred and increase productivity, and it is clear that the diseases can be detected at an early stage before they lead to deaths of the chicken. Computer vision work has been trying to reduce the gap for the past few decades by designing automated systems that can process images for decision making using computers. Specifically, generation of the CNN model, which learns the hidden pattern among the different faecal images in the dataset. The supervised learning model predicts the three categories, which are named as Coccidiosis, Healthy and Salmonella. The results obtained show that the XceptionNet model proposed model achieved the best accuracy. The experimental findings indicate that the model works well on early chicken diseases detection.

5.2 Recommendations

From the findings in this work, recommendations are stakeholders' inclusion that is farmers and extension officers to collect more data and expand the dataset repository to give room for future studies. Furthermore, the model can be deployed in the mobile device for easy use by the end-users and the addition of more features like recommender systems on poultry keeping. Future research can be done using the dataset since it is on open access.

REFERENCES

Abdisa, T., & Tagesu, T. (2017). Review on Newcastle Disease of Poultry and its Public Health Importance. *Journal of Veterinary Science and Technology*, *08*(03), 1-7. https://doi.org/ 10.4172/2157- 7579 .1000441

Albarqouni, S., Baur, C., Achilles, F., Belagiannis, V., Demirci, S., & Navab, N. (2016). AggNet: Deep Learning from Crowds for Mitosis Detection in Breast Cancer Histology Images. *Transactions on Medical Imaging*, *35*(5), 1313–1321. https://doi.org/10.1109/ TMI.2016.2528120

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., Zhang, X., Zhao, J., & Zieba, K. (2016). *End to End Learning for Self-Driving Cars.* http://arxiv.org/abs/1604.07316

Brooks-Pollock, E., de Jong, M. C. M., Keeling, M. J., Klinkenberg, D., & Wood, J. L. N. (2015). Eight challenges in modelling infectious livestock diseases. *Epidemics*, *10*, 1–5. https://doi.org/10.1016/j.epidem.2014.08.005

Cambria, E., & White, B. (2014). Jumping NLP curves: A review of natural language processing research. *Computational Intelligence Magazine*, *9*(2), 48–57. https:// doi.org/10.1109/ MCI.2014. 2307227

Chollet, F. (2017). *Xception: Deep learning with depthwise separable convolutions. Proceedings - 30th Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *January*, 1800–1807. https://doi.org/10.1109/CVPR.2017.195

Dawud, A. M., Yurtkan, K., & Oztoprak, H. (2019). Application of deep learning in neuroradiology: Brain haemorrhage classification using transfer learning. *Computational Intelligence and Neuroscience*, *2019*, 1-12. https://doi.org/10.1155/2019/4629859

Desin, T. S., Köster, W., & Potter, A. A. (2013). Salmonella vaccines in poultry: Past, present and future. *Expert Review of Vaccines*, *12*(1), 87–96. https://doi.org/10.1586/erv.12.138

FAO. (2013). *Poultry Development Report*. http://www.fao.org/3/i3531e/i3531e.pdf

Felip, V. P. S., Silv, M. A., Valent, B. D., & Ros, G. J. M. (2014). Using Multiple Regression, Bayesian Networks and Artificial Neural Networks for Prediction of Total Egg Production in European Quails. *Proceedings of 10th World Congress of Genetics Applied to Livestock Production*, 772–781. https://doi.org/10.3382/ps/pev031

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, *145*(2018), 311–318. https://doi.org/10.1016/j.compag.2018.01.009

Grilli, G., Borgonovo, F., Tullo, E., Fontana, I., Guarino, M., & Ferrante, V. (2018). A pilot study to detect coccidiosis in poultry farms at early stage from air analysis. *Biosystems Engineering, 2018*, 1–7. https://doi.org/10.1016/j.biosystemseng.2018.02.004

He, K., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition (CVPR)*, 770-778. https://openaccess.thecvf.com/contentcvpr 2015/html/He Deep Residual Learning

Hemalatha, Muruganand, S., & Maheswaran, R. (2014). Recognition of Poultry Disease in Real Time. *Proccedings of the International Conference on Inter-Disciplinary Research in Engineering and Technology 2014*. https://edlib.net/2014/icidret/icidret2014008.pdf

Hepworth, P. J., Nefedov, A. V., Muchnik, I. B., & Morgan, K. L. (2012). Broiler chickens can benefit from machine learning: Support vector machine analysis of observational epidemiological data. *Journal of the Royal Society Interface*, *9*(73), 1934–1942. https://doi.org/10.1098/rsif.2011.0852

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. http://arxiv.org/abs/1704.04861

Iorga, C., & Neagoe, V. E. (2019). A Deep CNN Approach with Transfer Learning for Image Recognition. *Proceedings of the 11th International Conference on Electronics, Computers and Artificial Intelligence, ECAI 2019*. https://doi.org/10.1109/ECAI 46879.2019.9042173

Kumar, S., Pandey, A., Sai, S, K., Kumar, S., Singh, S. K., Singh, A. K., & Mohan, A. (2018). Deep learning framework for recognition of cattle using muzzle point image pattern. *Measurement: Journal of the International Measurement Confederation*, 116, 1–17. https://doi.org/10.1016/j.measurement.2017.10.064

Lee, H., Eum, S., & Kwon, H. (2019). Is Pretraining Necessary for hyperspectral image classification? *International Geoscience and Remote Sensing Symposium (IGARSS)*, 3321–3324. https://doi.org/10.1109/IGARSS.2019.8898734

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine learning in agriculture: A review. *Sensors (Switzerland)*, *18*(8), 1–29. https://doi.org/10.3390/s 18082674

Lim, L., Tay, Y., Alias, H., Wan, K., & Dear, P. H. (2012). Insights into the genome structure and copy-number variation of Eimeria tenella. *Springer*, *13*(389), 1-10.

Lwoga, E., Ngulube, P., & Stilwell, P. (2010). Information needs and information seeking behaviour of small-scale farmers in Tanzania. *Journal of appropriate librarianship and information work in Southern Africa*, *2010*(40). https://doi.org/10.4314/innovation.v 40i1. 60088

Makantasis, K., Karantzalos, K., Doulamis, A., & Doulamis, N. (2015). Deep supervised learning for hyperspectral data classification through convolutional neural networks. *International Geoscience and Remote Sensing Symposium, 2015-(11)*, 4959–4962. https://doi.org/10.1109/IGARSS.2015.7326945

Ministry of Livestock and Fisheries. (2015). *Tanzania Livestock Modernization Initiative. Ministry of Livestock and Fisheries Development: United Republic of Tanzania*. https://livestocklivelihoodsandhealth.org/wpcontent/uploads/2015/07/Tanzania_Livestock_Modernization_Initiative_July_2015.pdf

Morales, I. R., Cebrián, D. R., Fernandez-Blanco, E., & Sierra, A. P. (2016). Early warning in egg production curves from commercial hens: A SVM approach. *Computers and Electronics in Agriculture*, 121, 169–179. https://doi.org/10.1016/j.compag.2015.12.009

Mulisa, D. D., W/Kiros, M. K., Alemu, R. B., Keno, M. S., Furaso, A., Heidari, A., Chibsa, T. R., & Chunde, H. C. (2014). Characterization of Newcastle Disease Virus and poultry-handling practices in live poultry markets, Ethiopia. *SpringerPlus*, *3*(1), 1–6. https://doi.org/10.1186/2193-1801-3-459

Ning, G., Zhang, Z., Huang, C., Ren, X., Wang, H., Cai, C., & He, Z. (2017). Spatially supervised recurrent convolutional neural networks for visual object tracking. *Proceedings - IEEE International Symposium on Circuits and Systems*, *1,* 1–4. https://doi.org/10.1109/ISCAS.2017.8050867

Okinda, C., Lu, M., Liu, L., Nyalala, I., Muneri, C., Wang, J., Zhang, H., & Shen, M. (2019). A machine vision system for early detection and prediction of sick birds: A broiler chicken model. *Biosystems Engineering*, *188,* 229–242. https://doi.org/10.1016/j. biosystemseng. 2019.09.015

Owomugisha, G., Quinn, J. A., Mwebaze, E., & Lwasa, J. (2014). Automated Vision-Based Diagnosis of Banana Bacterial Wilt Disease and Black Sigatoka Disease. *International Conference on the Use of Mobile ICT in Africa, 2014,* 1-5

Pu, H., Lian, J., & Fan, M. (2018). Automatic Recognition of Flock Behavior of Chickens with Convolutional Neural Network and Kinect Sensor. *International Journal of Pattern Recognition and Artificial Intelligence*, *32*(7), 1–15. https://doi.org/10.1142/S021800 1418500234

Quiroz-Castañeda, R. E., & Dantán-González, E. (2015). Control of avian coccidiosis: Future and present natural alternatives. *BioMed Research International*, *2015,* 1-10. https://doi.org /10.1155/2015/430610

Ramcharan, A., Baranowski, K., McCloskey, P., Ahmed, B., Legg, J., & Hughes, D. P. (2017). Deep learning for image-based cassava disease detection. *Frontiers in Plant Science*, *10,* 1–7. https://doi.org/10.3389/fpls.2017.01852

Rani, P. J., Bakthakumar, J., Kumaar, B. P., Kumaar, U. P., & Kumar, S. (2017). Voice controlled home automation system using natural language processing (NLP) and internet of things (IoT). *ICONSTEM 2017 - Proceedings: 3rd IEEE International Conference on Science Technology, Engineering and Management, 2018-January*, 368–373. https://doi.org/10.1109/ICONSTEM.2017.8261311

Sadeghi, M., Banakar, A., Khazaee, M., & Soleimani, M. (2015). Classification of Chickens Infected by Clostridium Perfringens Based on their Vocalization. *Brazilian Journal of Poultry Science*, *17*(4), 537-5444

Simonyan, K., & Zisserman, A. (2015). *Very deep convolutional neural networks for large-image scale recognition*. 1–14. https://arxiv.org/abs/1409.1556

State, O., Owade, A.A., Sonibare., A.O (2016). Persistence of Newcastle disease virus in poultry flocks : diagnostic challenges. *Journal of veterinary Science*, 11, 25–29

Wang, W., Hu, Y., Zou, T., Liu, H., Wang, J., & Wang, X. (2020). A New Image Classification Approach via Improved MobileNet Models with Local Receptive Field Expansion in Shallow Layers. *Computational Intelligence and Neuroscience*, *2020*, 1-10. https:// doi.org/10.1155 /2020/ 8817849

Wang, W., Li, Y., Zou, T., Wang, X., You, J., & Luo, Y. (2020). A novel image classification approach via dense-mobilenet models. *Mobile Information Systems*, 2020, 1-8. https://doi.org/10.1155/2020/7602384

Wong, J. T., de Bruyn, J., Bagnol, B., Grieve, H., Li, M., Pym, R., & Alders, R. G. (2017). Small-scale poultry and food security in resource-poor settings: A review. *Global Food Security*, *15*, 43–52. https://doi.org/10.1016/j.gfs.2017.04.003

Wulandari, M., Basari, & Gunawan, D. (2019). Evaluation of wavelet transform preprocessing with deep learning aimed at palm vein recognition application. *AIP Conference Proceedings, 2193*(12), 1-9. https://doi.org/10.1063/1.5139378

Yazdanbakhsh, O., Zhou, Y., & Dick, S. (2016). An intelligent system for livestock disease surveillance. *Information Sciences, 2016*, 1-54. Https://doi.org/10.1016/j.ins.2016.10.02

Yue, Z., Ma, L., & Zhang, R. (2020). Comprison and Validation of Deep Learning Models for the Diagnosis of Pneumonia. *Computational Intelligence and Neuroscience*, *2020*, 1–8. https://doi.org/10.1155/2020/8876798

Zhang, H., & Chen, C. (2020). Design of Sick Chicken Automatic Detection System Based on Improved Residual Network. *Proceedings of 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference*, *ITNEC 2020*, *Itnec*, 2480–2485. https://doi.org/10.1109/ITNEC48623.2020.9084666

Zhang, Z., & Han, Y. (2020). Detection of Ovarian Tumors in Obstetric Ultrasound Imaging Using Logistic Regression Classifier with an Advanced Machine Learning Approach. *Access*, *8*, 44999–45008. https://doi.org/10.1109/ACCESS.2020.2977962

Zhuang, X., Bi, M., Guo, J., Wu, S., & Zhang, T. (2018). Development of an early warning algorithm to detect sick broilers. *Computers and Electronics in Agriculture*, *144*(11), 102–113. https://doi.org/10.1016/j.compag.2017.11.032

Zhuang, X., & Zhang, T. (2019). Detection of sick broilers by digital image processing and deep learning. *Biosystems Engineering*, *179*, 106–116 https://doi.org/10.1016/j. biosystemseng.2019. 01 .003

APPENDICES

Appendix 1: CNN Model source codes

```
import numpy as np
import pandas as pd
import tensorflow as tf
from matplotlib import pyplot as plt

BATCH_SIZE = 16 * strategy.num_replicas_in_sync
LR_START = 0.00000001
LR_MAX = 0.000001 * strategy.num_replicas_in_sync
LR_MIN = 0.00000001
LR_RAMPUP_EPOCHS = 6
LR_SUSTAIN_EPOCHS = 0
LR_EXP_DECAY = .5

def lrfn(epoch):
    if epoch < LR_RAMPUP_EPOCHS:
        lr = (LR_MAX - LR_START) / LR_RAMPUP_EPOCHS * epoch + LR_START
    elif epoch < LR_RAMPUP_EPOCHS + LR_SUSTAIN_EPOCHS:
        lr = LR_MAX
    else:
        lr = (LR_MAX - LR_MIN) * LR_EXP_DECAY**(epoch - LR_RAMPUP_EPOCHS - 
LR_SUSTAIN_EPOCHS) + LR_MIN
    return lr

lr_callback = tf.keras.callbacks.LearningRateScheduler(lrfn, verbose=True)

rng = [i for i in range(EPOCHS)]
y = [lrfn(x) for x in rng]
plt.plot(rng, y)
print("Learning rate schedule: {:.3g} to {:.3g} to {:.3g}".format(y[0], max(y), 
y[-1]))

from keras import metrics
def create_model(num):

    pretrained_model = tf.keras.applications.Xception(input_shape=[*IMAGE_SIZE, 
3], include_top=False)
    pretrained_model.trainable = True

    model = tf.keras.Sequential([
        pretrained_model,
        tf.keras.layers.GlobalAveragePooling2D(),
        tf.keras.layers.Dense(3, activation='softmax')
       ])
    model.compile(
        optimizer='adam',
        loss = 'categorical_crossentropy',
        metrics = ['accuracy']
    )

    return model
    history = model.fit(training_dataset, validation_data=validation_dataset,
                    steps_per_epoch=TRAIN_STEPS, epochs=EPOCHS, 
callbacks=[lr_callback])
```

RESEARCH OUTPUTS

Research Output 1: Publication

Mbelwa, H., Mbelwa, J., & Machuve, D.(2021). Deep convolutional Neural Network for Chicken Diseases Detection. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 12(2), 759-765. https://dx.doi.org/10.14569/ IJACSA. 2021. 0120295

Research Output 2: Poster Presentations