Ethiopian Logo & Trademark Classification And Detection Using Convolutional Neural Network

By

Amante Tadesse Waldamariam

Adama Science and Technology University
School of Electrical Engineering and Computing
Department of Computer Science and Engineering

Master Thesis

A Thesis Submitted to
The School of Graduate Studies of Adama Science and Technology University in
Presented in Partial Fulfillment of the Requirement for the Degree of Master of Science in Computer Science and Engineering

Adama

June 2022

Ethiopian Logo & Trademark Classification And Detection Using Convolutional Neural Network

By

Amante Tadesse Waldamariam

Advisor

Biruk Yirga (Ph.D.)

A Thesis Submitted to

The Department of Computer Science and Engineering

School of Electrical Engineering and Computing

The School of Graduate Studies of Adama Science and Technology University in

Presented in Partial Fulfillment of the Requirement for the Degree of Master of Science in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

Adama

June 2022

Declaration

I declare that this thesis proposal entitled “Ethiopian Logo And Trademark Classification And Detection Using Convolutional Neural Network” is my own work and has not been submitted to any university for similar purpose. The references used in this thesis are duly recognized by proper citations.

Amante Tadesse

_____	_____	_____
Name of Student	Signature	Date

Acknowlegment

Foremost, I would like to thank the ultimate GOD for his almighty, and his mother saint merry. I am thankful Lord, for everything that you allow to cross my path. I am also thankful for decisions that you allowed me to make and the lessons that come from these decisions.

Then, I would like to express my profound gratitude to Dr. Biruk Yirga, whose expertise was invaluable in giving the right direction, useful comments, and engagement from the beginning till the end. He continuously encourages mean always has been ready and enthusiastic to assist in any way he could throughout the process.

Next, I would like to thanks my brother Lemi Tadesse for his motivates me on my thesis and he communicates me with his friend Getahun Tadesse.

Next, I would like to thanks Getahun Tadesse, for his helping me continuous and fruitful communication we had. He made me knowledgeable in the area of computer vision.

Finally , I would like to thank my beloved and missed wife for her hidden support in my educational effort in general. And also thanks my mother, brothers and all my classmates for their support me up to ending of my thesis. At all thank you so much!!!!

Recommendation of Advisors

I, the major advisor of this research proposal, hereby certify that I have closely advised the student while developing this proposal and read the draft thesis proposal entitled “Ethiopian Logo And Trademark Classification And Detection Using Convolutional Neural Network” prepared under my guidance by Amante Tadesse. Therefore, I recommend the submission of the thesis to the department for further review and evaluation.

Dr. Biruk Yirga

_____	_____	_____
Major Advisor	Signature	Date

_____	_____	_____
Co-Advisor	Signature	Date

Approval Page for Thesis

I hereby certify that the recommendation and suggestion given by the thesis review committee are appropriately incorporated into the final Thesis entitled "**Logo And Trademark Classification And Detection Using Convolutional Neural Network**" by **Amante Tadesse W/Mariam**.

Dr. Biruk Yirga

Major Advisor	Signature	Date
Co-advisor	Signature	Date

Approval of Board of Reviewers

We, the undersigned, members of the Board of Reviewers of the thesis open defense by **Amante Tadesse W/mariam** have read and evaluated the thesis entitled "**Logo And Trademark Classification And Detection Using Convolutional Neural Network**" and assessed the understanding of the candidate about the proposed research. This is, therefore, to certify that the thesis is accepted and we recommend the implementation of the thesis.

Chairperson	Signature	Date
Reviewer 1	Signature	Date
Reviewer 2	Signature	Date

.

Final approval and acceptance of the thesis proposal is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

Department Head	Signature	Date
School Dean	Signature	Date
Office of Post graduate studies, Dean	Signature	Date

Contents

Table of Contents

Lists of tables

Lists of figures

Lists of abbreviations

DL Deep Learning

FC Fully Connected

GPU Graphical Processing Unit

ML Machine Learning

SVM Support Vector Machine

YOLO You Only Look Once

ANN Artificial neural network

Adam Adaptive learning rate optimization algorithm

CPU Central processing unit

Conv Convolution

CL Convolution layer

ConvNets Convolutional neural networks

FN False negatives

FCL Fully connected layer

GHz Giga hertz GB Gigabytes

GPU Graphical processing unit

PL Pooling layer

FP False positives

PPV Positive predictive value

CNN P Convolutional Neural Network Precision

Abstract

In this paper we propose a method for logo Classification and detection using deep learning. Our recognition pipeline is composed of a logo region proposal followed by a Convolutional Neural Network (CNN) specifically trained for logo classification, even if they are not precisely localized. Experiments are carried out on the Fifteen logo and trademark, and we evaluate the effect on classification performance of synthetic versus real data augmentation, and image pre-processing. Moreover, we systematically investigate the benefits of different training choices such as class-balancing, sample-weighting and explicit modeling the background class (i.e. no-logo regions). Experimental results confirm the feasibility of the proposed method, that outperforms the methods in the state of the art. The features extracted are fit into the neural network with 100 epochs, 80/20 splitting ratio, and 0.0001 learning rate. EthioLogoNet model achieved Training accuracy 98.14%, and validation accuracy 98.27. The overall accuracy obtained 98.1% can be detect the logo and trademark detect and classify the types of logo and trademark with a high rate of accuracy.

Keywords: **logo Classification, deep learning, Convolutional Neural Networks (CNNs), artificial intelligenc**

Chapter One Introduction

1.1.Background of the study

Logos are symbols that are generally used by firms to identify themselves and their products. They normally contain colors, shapes, textures, and/or text [1]. Logo recognition is a key problem for many applications such as copyright infringement detection, online brand management, vehicle recognition, contextual advertisement placement, etc. [2, 3]. Although companies do not change their logos often, and it is only the context in which the logo appears that changes for each product of the same company, logo recognition is still a challenging task. Some of the challenges for accurate logo recognition are perspective deformations, varying background, occlusions, warping, varying size, varying colors, etc. [1]. Moreover, the growing number of brands having personalized logos makes the logo recognition task even more challenging. The logo recognition systems require high computational powers to support multi-class classification efficiently.

Pattern Recognition is a branch of digital image processing and artificial intelligence. Researchers and technicians of this science aim to develop techniques that identify specific patterns or structures in the digital images. In daily life, there are many applications and technologies related to this science such as recognizing faces in the control systems, logos and geometric shapes. The general idea of pattern recognition is all the processes that make the computers understand the patterns in the same way that is understood by the human and even more efficiently in some cases. As mentioned earlier about the wide usage and multiple applications of pattern recognition, in this thesis, we will focus on logo recognition.

Nowadays logos are important patterns because there is a significant dependency on logos and the existence of the logo obviates the name a lot of times. Also, the logos are used widely in various domains. For example, shops, productivity companies, humanitarian organizations and universities use the logos in advertising, documentation and transportation systems. Logos can be written by characters, expressive drawn or both. Each logo has specific features including

the color, the shape, and symbol. So most companies and organizations are seeking for a unique logo for them.

Generally, logo recognition is a part of pattern recognition which has an essential role in image processing. Many types of research and studies have worked on logo recognition and still take more attention in this time. Also, logo recognition is considered a wide field of research and innovation. At this time, the technological revolution and the dependence on devices and artificial intelligence techniques, require more studies and an expansion at this science. Logo recognition science requires precision, concentration, and development depending on the areas evolution where it is used.

Logo Classification and detection is a process of understanding and defining the logos automatically by the automated systems. Logo classification and detection works on avoiding the human mistakes in works which require great patience and accuracy.

One of the domains that use the logo and trademark recognition in University, Bank government and non-Government system and trader area. Logo and trademark recognition in different system can use many modern technologies for monitoring and controlling of mistakes and similarities of logo and trademark.

1.2.Statement of the Problem

Logo and trademark classification and detection with high accuracy and high processing speed has great importance for the community. Today, how to extract high-quality monetary features from logo images is a key problem area that needs solutions in logo and trademark classification and detection. In this regard, in a different part of the world, different researchers have been doing research and suggest a software and hardware solution in recognizing logo. To mention some of the studies which contributed to this concern.

Different researcher can do logo classification and detection in different method. Sahel et al(2021)[17], proposed deep learning pre trained model. In this paper, RCNN, FRCNN and retinanet for logo recognition. Faster R-CNN model is composed of RPN(Region Proposal Network) and a firm R-CNN with communal convolutional feature layer. The used flickr logos-32 common public dataset. Steven C-h. hoi et al(2015)[48] proposed logonet for large scale logo detection and brand recognition. Logonet has two dataset logo18 and logo160. Logo18 has 18 logo classes,10 brands and 16043 logo objects and logo160 has 160 logo classes, 100 brands and

130,608 logo objects. For logo 18 50% for training, 20% for validation and 30% for testing and logo 160 20% for training, 20% for validation and 60% for testing they used. Karim et al(2019) [49] propose logo recognition using deep CNN. Pre-trained are employed for feature extraction and classification using SVM classifiers. Pre-trained are modified for logo recognition. They used voting algorithm for training and testing of fine tuned deep models. Alsheikhy et al(2020)[50], proposed authomatic logo recognition using CNN. In this paper, DCNN ensure logo recognition without using huge computation resources. Densnet model was trained and tested on Flicker logo 32 dataset. Ping shi et al(2016)[51], proposed region based CNN for TV logo classification. In this paper, maximally stable extermal Region(MSER) used as method of generating candidate boxes per image. CNN used for TV logo classifying. Yuanyuan li(2017)[52], proposed graphic logo detection using CNN. They use Flicker logo 32 dataset and flicker logo 32 has 32 classes and each classes 70 images, 8240 logo images and 6000 non logos images.

From study wise, there are a lot of studies worked on different kinds of logo detections and recognition. But, with all their performances, at the same time, they do have their own drawbacks as no research has a perfect ending. This study tried to figure out some of the drawbacks that occurred in previous researches. Here are some basic problems tried to be figured out in this study. And there is no research done in case of Ethiopian logo and trademark recognition.

This study addresses these problems by building a new dataset of utilizing machine learning and deep learning methods this study addressed the problem by answering the following questions

1.3. Research Questions

The questions that this research is going to answer are as follows:-

- How to build deep learning model to detect fake letters by classifying logo?
- How to build deep learning model to detect fake beverage, food and other trademark?
- How to implement deep learning models to accurately logo and trademark recognition?
- Which deep learning approach is better for fake logo and trademark detection in Ethiopia?

1.4.Objective of the Study

General Objectives

The general objective of this study is to design and develop an efficient and more accurate Logo and trademark Classification and detection model using the Convolutional Neural Network for logo & trademark

Specific Objective

The following are the specific objectives of the research that help to achieve the general objective of the study.

- ✓ To review various literatures in order to have an understanding on concepts, principles, Terminologies, Technologies of studying logo and trademark recognition frameworks and forward research ideas.
- ✓ Data Source Identification.
- ✓ Evaluate the performance of the model and the prototype.
- ✓ Interpret the results and draw conclusions
- ✓ To identify and recommend future research directions for further investigation on the benefits of Logo and trademark classification and detection

1.5 Significance of the study

For many stakeholders and other researchers, the study has distinct benefits. It can be used in a variety of fields following the outcomes evaluations

- The study benefits the government and private sectors such as Drug and Food Control authority
- This work could be beneficiary to Federal and regional Police institutions
- Government and nongovernment Universities and colleges are beneficiary from this work

1.6.Scope and Limitations of the Study

1.6.1.Scope of the Study

The study focuses on designing and implementing a model that can automatically detect if a logo and trademark fake. We have fifteen types logo and trademark for detection and classification of the fake or genuine.

1.6.2.Limitation of the study

The limitation of this proposed research focuses on fifteen(15) types of Logo and Trademark. Logo and Trademark are AASTU, ASTU, Arsi University, Arki Water, Coca Cola, Mirinda, Sprite, Bahar University, Gondar University, Fanta, One Water ,Pepsi, Tena Oil, Top Water, Yes Water. The proposed system not including TV, Clothes and Shoes logo.

1.7. Organization of the Study

The rest of this thesis is organized as follows. Chapter one is an introductory chapter that focuses on the background of the study, the motivation behind the study, statement of problems with research questions to be raised and addressed, the scope and limitation of the study, objectives to be reached and application of the study. Chapter two presents the review of literature on the domain of pattern recognition, image processing and classification, machine learning, deep learning, and studies that are conducted in those domains. At the end of chapter two reviews of studies that directly related to this thesis are discussed and summarized. In chapter three the methodologies that are used to conduct this thesis are discussed briefly including data collection methods, materials, and tools which have been used, selection of appropriate model, the architectural design of the proposed model, and evaluation techniques are discussed. Chapter four deals with the proposed model and experimental parameters on Logo and Trademark classification and detection fake logo and trademark images. In addition to these different experimental results with detailed discussions are presented. Finally, the conclusion and recommendations are presented in chapters five and six.

Chapter Two Literature Review

2.1.Pattern Recognition

Pattern recognition is the science for observing the environment, learning to recognize patterns of interest from their background and making right decisions about the patterns or pattern classes. [4].

Pattern recognition definition has been introduced by [5] according to earlier studies. pattern recognition as a field which is concerned with machine recognition of meaning regularities in complex and noisy environment [6]. As well as defined the pattern recognition in his book as the term pattern is derived from the same root as the term patron and in its original use; it means something that is set up as an optimal example to be imitated pattern recognition as classifications of input data by extracting the features which have importance from a lot of noisy data [7]. pattern recognition can be imaginable as the classifications problem, structure analysis, inductive process, discrimination approach and so on [8]. pattern recognition is A problem of assessing density functions in a high- dimensional space and separating the space to regions of classifications of classes [9]. pattern recognition as The science that deals with the classification or description of the measurements [10].

pattern recognition as the discipline that learn some theories and approaches in order to design machines which have the ability to identify the patterns in noisy data and complex environment [11].pattern recognition in his book as Given some examples of complex signals and the correct decisions for them, make decisions automatically for a stream of future examples [12].pattern recognition is an engineering approach and its pattern recognition final goal is designing machine that can solve the existed gap between the application and the theory [13].pattern recognition as a scientific discipline and its aim is to classify the object into many categories of classes [14].

The design of a pattern recognition system essentially involves the following phases, one is data building; the other two are pattern analysis and pattern classification. Data building transforms original information into vector which can be processed by computer. Pattern analysis task is to process the data (vector), such as feature selection, feature extraction, data dimension compress and so on. The goal of pattern classification is to use the information obtained from pattern analysis to discipline the computer in order to accomplish the classification [5].

Pattern recognition can be used in any area which uses the observations structures. Currently, pattern recognition is commonly used in numerous applications related to military, healthcare and manufacturing industry, such as face recognition, character recognition, computer vision, and speech recognition, recognizing fingerprints, recognition of automobile type personal identification systems, fault diagnosis for vehicle system, and enhance the automobile safety performance [5].

2.1.1. Image Processing

Image Processing is a method to transform an image into digital form and implement some operations on it, in order to get an enhanced image or to extract some useful features from it. It is a type of signal release in which input is image, like video frame or photograph and output may be image or characteristics associated with that image. Usually Image Processing system includes handling images as two dimensional signals while applying already set signal processing methods to them [15], [16].

The purpose of image processing is separated into five groups

- ✓ Visualization – To observe the invisible objects.
- ✓ . Image sharpening and restoration - To generate a better image.
- ✓ Image retrieval – To seek for the image of interest.
- ✓ . Pattern measurement– To measure several objects in the image.
- ✓ Image Recognition – To differentiate the objects in the image.[16]

Medical Image Retrieval image processing has been extensively used in medical research and has enabled more efficient and accurate treatment plans. For example, it can be used for the early detection of breast cancer using a sophisticated nodule detection algorithm in breast scans. Since medical usage calls for highly trained image processors, these applications require significant implementation and evaluation before they can be accepted for use.

Traffic Sensing Technologies in the case of traffic sensors, we use a video image processing system or VIPS. This consists of an image capturing system, a telecommunication system and an image processing system. When capturing video, a VIPS has several detection zones which output an "on" signal whenever a vehicle enters the zone, and then output an "off" signal whenever the vehicle exits the detection zone. These detection zones can be set up for multiple lanes and can be used to sense the traffic in a particular station. Besides this, it can auto record the license plate of the vehicle, distinguish the type of vehicle, monitor the speed of the driver on the highway and lots more.[55]

Image Reconstruction Image processing can be used to recover and fill in the missing or corrupt parts of an image. This involves using image processing systems that have been trained extensively with existing photo datasets to create newer versions of old and damaged photos.

Face DetectionOne of the most common applications of image processing that we use today is face detection. It follows deep learning algorithms where the machine is first trained with the specific features of human faces, such as the shape of the face, the distance between the eyes, etc. After teaching the machine these human face features, it will start to accept all objects in an image that resemble a human face. Face detection is a vital tool used in security, biometrics and even filters available on most social media apps these days.[55]

The implementation of image processing techniques has had a massive impact on many tech organizations. Here are some of the most useful benefits of image processing, regardless of the field of operation

The digital image can be made available in any desired format (improved image, X-Ray, photo negative, etc).It helps to improve images for human interpretation. Information can be processed and extracted from images for machine interpretation.The pixels in the image can be manipulated

to any desired density and contrast. Images can be stored and retrieved easily. It allows for easy electronic transmission of images to third-party providers [55].

2.1.2.Image Classification

Image Pre-processing: The aim of this process is to improve the image data (features) by suppressing unwanted distortions and enhancement of some important image features so that the computer vision models can benefit from this improved data to work on. Steps for image pre-processing includes Reading image, Resizing image, and Data Augmentation (Gray scaling of image, Reflection, Gaussian Blurring, Histogram, Equalization, Rotation, and Translation).[18]

Detection of an object: Detection refers to the localization of an object which means the segmentation of the image and identifying the position of the object of interest.

Feature extraction and training: This is a crucial step wherein statistical or deep learning methods are used to identify the most interesting patterns of the image, features that might be unique to a particular class and that will, later on, help the model to differentiate between different classes. This process where the model learns the features from the dataset is called model training.[18]

Classification of the object: This step categorizes detected objects into predefined classes by using a suitable classification technique that compares the image patterns with the target patterns.

Supervised classification is based on the idea that a user can select sample pixels in an image that are representative of specific classes and then direct the image processing software to use these training sites as references for the classification of all other pixels in the image. Training sites (also known as testing sets or input classes) are selected based on the knowledge of the user. The user also sets the bounds for how similar other pixels must be to group them together. These bounds are often set based on the spectral characteristics of the training area. The user also designates the number of classes that the image is classified into. Once a statistical characterization has been achieved for each information class, the image is then classified by examining the reflectance for each pixel and making a decision about which of the signatures it resembles most. Supervised classification uses classification algorithms and regression techniques to develop predictive models. The algorithms include linear regression, logistic regression, neural networks, decision tree, support vector machine, random forest, naive Bayes, and k-nearest neighbor.[18]

Unsupervised classification is where the outcomes (groupings of pixels with common characteristics) are based on the software analysis of an image without the user providing sample classes. The computer uses techniques to determine which pixels are related and groups them into classes. The user can specify which algorithm the software will use and the desired number of output classes but otherwise does not aid in the classification process. However, the user must have knowledge of the area being classified when the groupings of pixels with common characteristics produced by the computer have to be related to actual features on the ground. Some of the most common algorithms used in unsupervised learning include cluster analysis, anomaly detection, neural networks, and approaches for learning latent variable models.[18].

2.2Why Deep Learning over Machine Learning?

2.2.1 Machine Learning

Machine learning is a branch of Artificial intelligence which is a way of making machines intelligent to learn from a bunch of learning algorithms and apply what they have learned to make brilliant decisions like human being, but as long as they are still machine and in some situations they need human intervention, because they work only for what they are designed for no more no less. That‘s why we need deep learning, which gives us a bit better performance.

2.2.2 Deep Learning

Practically, it is[26][27] a subset of machine learning with more power and flexibility than traditional machine learning approaches. The biggest advantage of **Deep Learning** algorithms is that they try to learn high-level features from data in an incremental manner. This prevents the need for manual feature extractions which is done by human intervention in traditional.

machine learning. Deep learning emulates the human brain and it works with unstructured data, especially for applications like computer vision(image processing) and speech recognition. [28]

2.2.3 When to Use Deep Learning?

Deep learning performs more and more when we have a large data size and good computational performances. It outperforms other approaches when it comes to really complex problems like image classifications, natural language processing, and speech recognition. Basically, we need to have two crucial things in Deep Learning. The first one is large data size to train our model and the second one is a good computational device because deep learning learns up to millions of parameters in a model and our device should be capable of overcoming those bunches of parameters easily.

2.2.4. Convolutional Neural Networks

CNN is the class of deep neural networks (deep learning), most commonly used for image processing tasks. It can obtain effective representations of the original image, which makes it possible to recognize visual patterns directly from raw pixels with little-to-none preprocessing. ConvNets are constructed with processing layers namely, Convolutional, pooling and fully connected or Dense. In CNN, each layer takes information from the previous layer[29] and processes it to the succeeding layer, especially CNN works best for image data[30], basically large image data set. The Design and layout of CNN the layers are decided by the model designer. The design depends on the type and complexity of the problem to be solved as well as the expected output of the network. The output of a ConvNets can be either a posterior probability of a sample belonging to various classes, or discriminant features that can be used with any other classifier.

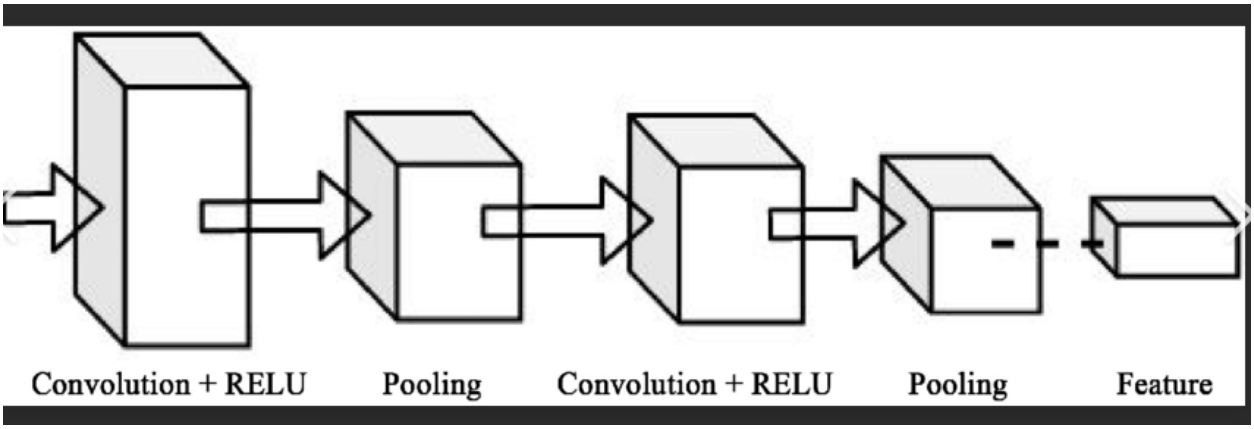


Figure 1 CNN Architecture[53]

2.2.5.CNN Layers

2.2.5.1.Convolution Layer (CL)

In computer vision, images are read as pixels and are represented as a matrix of (NxNx3) (height by width by depth respectively). Colored Images make use of three channels (RGB), so that is why we have a depth of 3 [31][32]**.** Convolution filter in basic CNNs is a Generalized Linear Model (GLM) for the underlying local image patch. It works well for abstraction when instances of latent concepts are linearly separable. Here we introduce some works which aim to enhance its representation ability. The main processing component of this layer is a filter or mask which is a matrix of weights. This mask is applied to a specified region of the feature map. The Convolutional Layer makes use of a set of learnable filters. A filter is used to detect the presence of specific features or patterns present in the original image(input). It is usually expressed as a matrix (MxMx3), with a smaller dimension but the same depth as the input file.

The convolution operation is mathematically defined a

$$S[i, j] = \Sigma.m\Sigma[m, n].K[u-m, j-n]$$

where I is the input image with dimensions m and n, K is the kernel, and S is the resulting output at location i and j. Applying the convolution operation provides several benefits that are based on three important ideas sparse interactions (or sparse weights), parameter sharing, and equivalent representations. The term sparse interactions means that fewer connections between input and output values exist. Therefore, fewer parameters are needed which reduces the memory requirements and improves the statistical efficiency. This is achieved by applying smaller kernel sizes than the input image because fewer operations are required to produce the output. In contrast, traditional neural network layers, which do not employ convolutions, use matrix multiplication to describe the relationships between each input value and each output value. This means that all outputs are connected to all inputs.

Convolutions are very efficient operations to describe transformations of small receptive fields across input images. Parameter sharing refers to using the same values for more than one function in a network. In traditional neural networks, each element of the weight matrix is used exactly once for computing the layer output. Each value of the kernel is used at every position of the input. The parameter sharing used by the convolution operation means that rather than learning a separate set of parameters for every location, only one set is learned. This does not affect the runtime but further reduces the storage requirements of the model to k parameters. When convolution is applied to images, convolutions create 2D feature maps that show where certain features appear in the input. Parameter sharing causes the layer to be equivariant to translation, i.e. if the input changes, the output changes in the same way. If the object in the input is moved, its representation will move the same amount in the output.

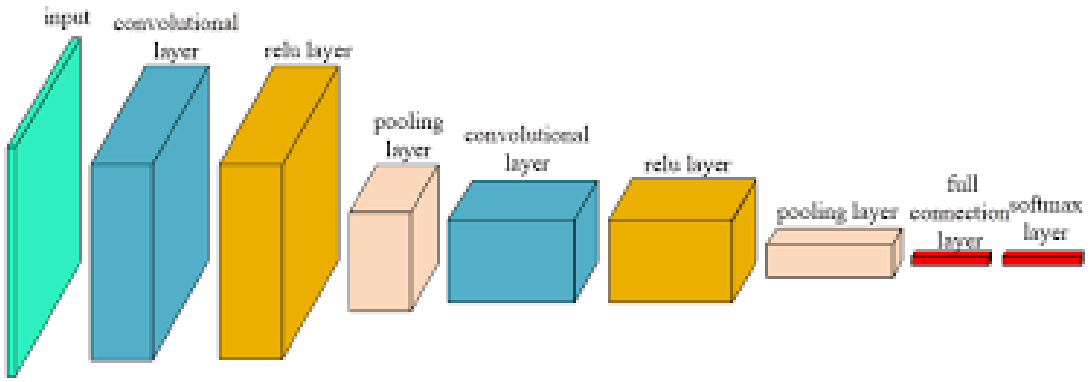


Figure 2:- convultion layer[54]

2.2.5.2.Pooling Layer (PL)

The pooling layer is an important component of CNN which lowers the computational burden, by reducing the number of connections between convolutional layers, which is basically also used controlling over fitting by progressively reducing the spatial size of the network**.** Unlike the convolution layer, the PL does not alter the depth of the network, the 17 depth dimension remains constant. Max pooling is the most commonly used pooling layer in the Convolutional neural network. [31][33]

2.2.5.3 Max-Pooling

As the name states it takes out only the maximum from a pool. This is actually done with the use of filters sliding through the input and at every stride, the maximum parameter is taken out and the rest is dropped.

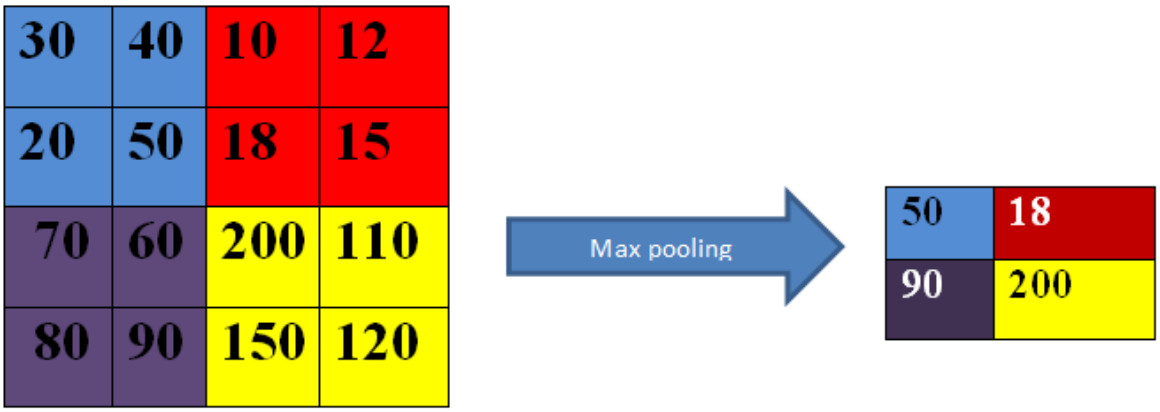


Figure 3:- Max Pooling Layer

2.2.5.4.Fully Connected Layer (FCL)

The last layers in the network are fully connected, meaning that neurons of preceding layers are connected to every neuron in subsequent layers. This mimics high-level reasoning where all possible pathways from the input to output are considered. Linear or Fully Connected Mathematically, we can think of a linear layer as a function that applies a linear transformation on a vectored input of dimension I and output a vector of dimension *O*. Usually the layer has a bias parameter. In this layer, the neurons have a complete connection to all the activations 18

from the previous layers. Their activations can hence be computed with a matrix multiplication followed by a bias offset. This is the last phase for a CNN network.

The Convolutional Neural Network is actually made up of hidden layers and fully-connected layer(s)**.** [31][34]

2.2.5.5. Activation

The activation layer controls how the signal flows from one layer to the next, emulating how neurons are fired in our brain. Output signals which are strongly associated with past references would activate more neurons, enabling signals to be propagated more efficiently for identification. ConvNet is compatible with a wide variety of complex activation functions to model signal propagation, the most common function is the Rectified Linear Unit (ReLU), which is favored for its faster training speed. The capacity of the neural networks to approximate any functions, especially non-convex, is directly the result of the non-linear activation functions. Every kind of activation function takes a vector and performs a certain fixed point-wise operation on it. Some of the activation functions explained in the following subsection. [35]

2.2.5.6.Rectified Linear Unit (ReLu)

The ReLU is the most used activation function in the world right now. Since it is used in almost all the convolutional neural networks or deep learning[31][35]**.** It does not suffer from 19

the vanishing or exploding gradient. Another advantage is that it involves cheap operations compared to the expensive exponentials.

The ReLU has the following mathematical form

$f(x) = \max(0, x)$

2.2.5.7. Sigmoid and Soft max

There are two most common activation functions that are used in deep learning approaches, which are soft max and Sigmoid. A **Sigmoid function** is a mathematical function having a characteristic "S"-shaped curve or **sigmoid curve**. Often, we use a sigmoid function when our problem is solved with binary classification, which means our result lies between zero and one(**0 & 1**) [21][35] [36]. It takes a real value and squashes it between 0 and 1. However, when the neuron‘s activation saturates at either a tail of 0 or 1, the gradient at these regions is almost zero. Thus, the back propagation algorithm fails at modifying its parameters and the parameters of the preceding neural layers. The sigmoid activation function is the most suitable activation function for binary classifications. Soft max is different because they are most effective in multi class classifications.

2.2.5.8. Over fitting and Under fitting

Over fitting and under fitting are the two most critical problems in neural network models and these problems happen because of different reasons

2.2.5.8.1 Over fitting

Over fitting happens when a model learns effectively on the training data but performs poorly on validation(hold out data).[37]

2.2.5.8.2.Under fitting

Under fitting happens when the model fails to effectively learn the problem and performs poorly on training data and also does not perform well on validation data. Therefore if we see a model that performs badly both on training and validation data, it should be known that the model suffers from under fitting and need to try to change our approach or by updating our parameters. [37]

2.2.5.9 Dropout

Dropout is a technique that helps to avoid over fitting by setting the outputs of hidden neurons to zero with a pre-set probability. The neurons which are set to zero do not contribute to the forward pass or back-propagation, i.e. they are ‘dropped out‘. Due to these ‘dropped-out‘ neurons, the

network samples a different architecture every time. This forces neurons to learn more robust features as they cannot rely on the presence of other neurons. A dropout ratio of 0.2 doubles the number of iterations required to converge but reduces over fitting.[38]

2.2.5.10. Batch Normalization

This layer quickly became very popular mostly because it helps to converge faster. It adds a normalization step (shifting inputs to zero-mean and unit variance) to make the inputs of each trainable layer comparable across features. By doing this it ensures a high learning rate while keeping the network learning. In addition to that, it also allows activations functions such as Sigmoid to not get stuck in the saturation mode (e.g. gradient equal to 0). [21]

2.2.5.11.Tools and Techniques.

2.2.5.11.1.Python

Python is an interpreted high-level, general-purpose programming language. It is an easy and powerful programming language that is dynamic and easy to catch up with. As it‘s a lot of built-in libraries for Artificial intelligence and Machine Learning, well documented, simple enough to catch up with, python is preferred in this study[39]. Some of the libraries are Tensor Flow, Keras, and sci-kit–learn. In this study, to produce the result, python 3.7.3 is used and runs on Windows 10, 64- bit operating system, with 4 GB RAM and 3.20 GHz of processor.

2.2.5.11.2.Tensor flow

Tensor Flow is a Python library for fast numerical computing created and released by Google. It is a foundation library that can be used to create Deep Learning models and it‘s an open-source library for fast numerical computing, which is an end to end open-source machine learning(ML) framework. Tensor Flow was designed for use both in research and development and in production systems. It can run on single CPU systems, GPUs as well as mobile devices and large scale distributed systems on hundreds of machines. So we can say Tensor Flow is a brilliant tool, with lots of power and flexibility. However, for quick prototyping work, it can be a bit verbose. [40]

2.2.5.11.3. Keras

Keras is a high-level neural network, API written in Python and capable of running on top of Tensor Flow, CNTK or Theano. It was developed with a focus on enabling fast experimentation. It is a model-level library providing high-level building blocks for developing deep learning architectures. It does not allow low-level operations like tensor manipulation and differentiation, Instead of that, it uses a specialized, well-optimized tensor library, serves as a backend engine for

Keras. Keras allows for easy and fast prototyping through user-friendliness, modularity, and extensibility.[41]

2.3 summary of related work

Sahel et al(2021), proposed deep learning pre trained model. In this paper, RCNN, FRCNN and retinanet for logo recognition. Faster R-CNN model is composed of RPN(Region Proposal Network) and a firm R-CNN with communal convolutional feature layer. The used flickr logos-32 common public dataset. Hardware specifications were CPU AMDryzen, GPU 2080, RAM 16GB, 2.3GB for RCNN, 1.5GB for FR-CNN and 3.1GDB for Retinanet hardware consumed. Faster R-CNN model of previous research results by 12% in mAP. They achieved accuracy training dataset 99.8 for FR-CNN and 95.2 with retinanet.

Steven C-h. hoi et al(2015) proposed logonet for large scale logo detection and brand recognition. Logonet has two dataset logo18 and logo160. Logo18 has 18 logo classes,10 brands and 16043 logo objects and logo160 has 160 logo classes, 100 brands and 130,608 logo objects. For logo 18 50% for training, 20% for validation and 30% for testing and logo 160 20% for training, 20% for validation and 60% for testing they used. DRCN technique include RCNN,FRNN and SPPNet. RCNN(caffenet with fine tuning) obtained best logo detection and brand recognition and FRCN(VGG16) obtained second best result. Caffenet accuracy is 95.2.

Karim et al(2019) propose logo recognition using deep CNN. Pre-trained are employed for feature extraction and classification using SVM classifiers. Pre-trained are modified for logo recognition. They used voting algorithm for training and testing of fine tuned deep models. Logo 32 plus they used for training and testing and logo 32 plus has 32 brands and 12312 logo instances. Resnet DCNN achieve best recognition 97.2 of accuracy. Resnet, googlenet and VGG16 using voting algorithm highest accuracy 98.4 in the proposed method.

Alsheikhy et al(2020), proposed authomatic logo recognition using CNN. In this paper, DCNN ensure logo recognition without using huge computation resources. Densnet model was trained and tested on Flicker logo 32 dataset. Flicker logo 32 public available dataset has 8240 logos with 32 classes. Proposed model accuracy was 92.8.

Ping shi et al(2016), proposed region based CNN for TV logo classification. In this paper, maximally stable extermal Region(MSER) used as method of generating candidate boxes per image. CNN used for TV logo classifying. TV logo dataset was collected from video streaming and some augmentation method. 120000 logo images collected from video frame for training and

validation and 1200 photo graphs for testing. CNN achieve high accuracy when compare to SIFT and HOG. Implementation code was written in keras.

Yuanyuan li(2017), proposed graphic logo detection using CNN. They use Flicker logo 32 dataset and flicker logo 32 has 32 classes and each classes 70 images, 8240 logo images and 6000 non logos images. This detector consists two stages Region proposal Network(RPN) and Fast R-CNN. RPN proposes object candidates and Fast R-CNN entracts feature for each proposal by ROI-pooling and bounding box regression. Faster R-CNN was trained end to end for 50k iteration with learning rate 0.001 and for 20k iteration 0.0001

Authors and years	Title	Models	Result	Remark
Sahel et al(2021)[17]	Logo Detection Using Deep Learning with Pre trained CNN Models	Three different models applied namely RCNN, FRCNN and retinanet	accuracy training 95.2 for retinanet.	It performs 95.2 accuracy
Alsheikhy et al(2020),[50]	Logo Recognition with the Use of Deep Convolutional Neural Networks	. Densenet model	Proposed model accuracy was 92.8.	Accuracy perform less than other model
Karim et al(2019) [49]	Logo recognition by combining deep convolutional models in a parallel structure	three different model applied Resnet,googlenet and VGG16	Resnet DCNN achieve best recognition 97.2 of accuracy.	
Yuanyuan li(2017),[52]	Graphic Logo Detection With Deep Region Based CNN	This detector consists two stages Region proposal Network(RPN) and Fast R-CNN.	Faster R-CNN learning rate 0.001	
Ping shi et al(2016)[51]	TV Logo Classification Based on CNN	CNN Model	CNN achieve high accuracy	Ony classifying TV logos
Steven C-h. hoi et al(2015)[48]	Large scale deep logo detection and brand recognition with deep region based convolutional network	Four different model applied: RCNN,FRNN and SPPNet. RCNN	Caffenet accuracy is 95.2.	Good for large dataset training and testing

Table:-literature Review summary

Chapter Three Research Methodology

3.1 General Approaches

Here are the general procedures I have followed to develop my own Logo and trademark model.

- The has been necessary data collected to train the model from different government and non-government offices
- The necessary tools used to develop the models are downloaded from their sites and set up on the computer to be trained on.
- Data Augmentation is used to generate a large amount of training data from a small amount of data since it is a must condition to have a lot of data in Deep Learning.
- Images used to train the models are preprocessed using different preprocessing techniques like resizing the images to a uniform standard so that the model can get normalized and uniform data to train and validate the data set.
- After preprocessing, the images are segmented by using the color code, which is proposed in this study.
- The Logo and trademark model is designed and implemented using the segmented images.

3.2 Training from Scratch

3.2.1 Initialization

Choosing the proper initialization for each layer is the key to achieving high performance. Initializing the weights in an appropriate way can be significantly influential to how easily the network learns from the training (as it effectively preselects the initial position of the model parameters with respect to the loss function to optimize in designing the model to experiment with the network parameters are initialized with Xavier sometimes called Glorot initialization. The key idea behind this initialization scheme is to make it easier for a signal to pass through the layer during forward as well as backward propagation, for a linear activation this also works nicely for sigmoid activations because of the interval where they are unsaturated is roughly linear as well. It draws weights from a probability distribution (uniform or normal) with variance.

3.3 Data Collection

In deep learning, data is the most crucial thing that we need to take care of. Deep Learning algorithms are different from other traditional machine learning algorithms in the size of the data they use, that they need a huge amount of data for training. In this study we have used three classifications of data sets for implementing our model which are training set , validation set and test set. The Image data collected totally are 3622 original images which are augmented to increase the size which is discussed in the next section.

3.4 Data Augmentation

Over fitting happens when we have too few samples of training data, and Data Augmentation is the best way to fight this over fitting [21] when we work on image data. It is a method of boosting the size of the training set so that the model cannot memorize all of it. This can take several forms depending on the dataset. For instance, if the objects are supposed to be invariant to rotation such as galaxies or planktons, it is well suited to apply different kinds of rotations to the original images. In this study there were totally 3622 original images and these images are augmented to 14488 images.

Data augmentation techniques in computer vision are the following

- ✓ Rotation
- ✓ Width shift
- ✓ Height shift
- ✓ Rescaling
- ✓ Shear
- ✓ Zoom
- ✓ Horizontal flip
- ✓ Fill mode
- ✓ Data format
- ✓ Brightness

3.5 Pre–Processing to resize Images

Images needs to be resized before they are fed to the model, so that the model can get a uniform image size format and will have the a uniform response time during the real world implementations. There is no a standard set to resize our images but we can resize it based on the

computational resources we have, because as the size of image increased, it needs more computational resources to be processed. In this study we have resized out images into 256x256 height and width dimension.

3.6 Optimization Algorithms

The loss function of a convolutional neural network is highly non-convex. Hopefully, the latter is also fully derivable so that gradient-based optimization algorithms can be applied. However, convolutional neural networks are usually made of millions of parameters. Thus, only the first-order derivatives are used in practice most widely used first-order optimization algorithm is Gradient descent. These algorithms tell us whether the function is decreasing or increasing at a particular point basically it gives a tangential line to a point on its error surface. The second derivative which is also called Hessian is a Matrix of second-order partial Derivatives. Since the second derivate are costly in term of memory and computational effort are not used much.

3.6.1 Gradient Descent

Gradient descent[22] is a way to minimize an objective function (θ) parameterized by a model's parameters θ E $R2$ by updating the parameters in the opposite direction of the gradient of the objective function $\nabla\theta$ (θ) with respect to the parameters. The learning rate η determines the size of the steps we take to reach a (local) minimum. In other words, we follow the direction of the slope of the surface created by the objective function downhill until we reach a valley.

3.6.2 Rmsprop Algorithm

Rmsprop is an unpublished[23][24] optimizer but, a kind of optimizer that became the most popular and effective. Rmsprop is similar to gradient descent with momentum. It increases our learning rates by restricting the vertical oscillations, and converging the horizontal oscillations faster so that our model can learn fast and effectively.

3.7 Performance Measurement Methods

3.7.1 Loss Function (objective function)

The loss function is the crucial element[25] in artificial neural networks, which is used to measure the inconsistency between predicted value (y^) and the actual label (y). It is a non-negative value, where the robustness of the model increases along with the decrease of the value of loss function. The loss function is the hardcore of empirical risk function as well as a significant component of structural risk function.

3.7.2 Cross-Entropy

Cross-entropy loss, or log loss, measure the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss increases as the predicted probability diverge from the actual label. The Mathematical computation behind[42] cross-entropy is tough, but fortunately, it is easy to understand and implement it.

3.8 Regularization Approaches

Deep and large enough neural networks can memorize any data. During training, their accuracy on the train set typically converges towards perfection while it degrades on the test set. This phenomenon is called over fitting. Thus different regularization approaches used to handle this over fitting problem.

3.8.1 Dropout

Dropout is one of the most effective and most commonly used regularization techniques for neural networks, developed by Geoff Hinton and his students at the University of Toronto. Applied to a layer, consists of randomly dropping out or setting to 0, a number of output features of the layer during training. In Logo and trademark model Dropout is used to minimize the over fitting that may happen during the model training.

3.8.2 Early Stopping

It consists of stopping the training before the model begins to over fit the training set. In practice, it is used a lot during the training of neural networks, and it is a means of halting model training before it starts to over fit. Early stopping is another regularization technique we have used in this study to terminate our model training before it starts to over fit after the model learned enough features from the images.

3.9 Evaluation Matrices

To predict the performance of the Logo and trademark model we need to use accuracy, specificity, sensitivity, confusion matrix precision, recall, F1.We calculate true positive(TP), True Negative(TN), false Positive (FP) and false Negative (FN).

3.9.1 Accuracy

Accuracy means the algorithm capability to predict the class of the dataset correctly. It determine of how close or near the predicted value is to the actual value[43][44].

$$\text{Accuracy} = \frac{TP+TN}{}$$

$$\frac{}{TP+FP+TN+FN}$$

3.9.2 Precision

Precision measure the true values correctly predicted from the total predicted values in the actual class[45]. Precision quantifies the ability of the classifiers to not label a **negative** example as **positive.**

$$\text{Precision} = \frac{TP}{TP+FP}$$

3.9.3 Recall

Recall determine the rate of the total positive values that are correctly predicted. Recall answers what percentage of actual Positives is correctly classified[46].

$$\text{Recall} = \frac{TP}{TP+FN}$$

3.9.4 F-measure

F-measure is also called F1-score is the harmonic mean between recall and precision. It takes both false positive value and false negatives values into account.

$$F1_{score} = 2 * \frac{\text{Precision} * \text{recall}}{\text{Precision} + \text{recall}}$$

3.9.5 Sensitivity

Sensitivity (SN) is as well as True Positive Rate (TPR). Sensitivity is the mean proportion of actual true positives that are correctly identified[47].

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

3.9.6 Specificity

Specificity (SP) is as well as True Negative Rate (TNR), is calculated the fraction of negative values that are correctly classified.

$$\text{Sensitivity} = \frac{TN}{TN+FP}$$

Chapter Four Proposed System

4.1 Proposed System

This topic focuses on the design of the proposed work and its experimental parameters. Specifically, the design of the proposed model and descriptions, how features are extracted, and classification are performed in the proposed model.

4.2 Model Design

A EthioLogoNet is a Deep Learning calculation that can accept a picture as information, relegate significance (learnable loads and predispositions) to unmistakable viewpoints/objects in the picture, and recognize one from the other. The design of a EthioLogoNet is roused by the association of the Visual Cortex and is comparable to the available example of Neurons in the Human Brain. Singular neurons can just react to boosts in a little space of the visual field called the Receptive Field. A few comparative fields can be stacked on top of one another to traverse the full visual field. When contrasted with other grouping techniques, the measure of pre-handling needed by a EthioLogoNet is altogether less.

The vital benefit of using the CNN calculation utilized in the EthioLogoNet model for identifying and classifying logo and trademark fakeness is that it is progressively robotized than customary AI calculations. The need to plan various calculations for various issues in the traditional AI calculation requires the use of more hand-tailored calculations, while on CNN of this model, we simply need to plan one calculation for all issues. EthioLogoNet is simply the convolutional neural organization model named without help from anyone else. This model is made from scratch utilizing an aggregate of 14,488 increased information pictures which are classified into fifteen classes. These classes are AASTU, ASTU, Arsi University ,Arki Water, Coca Cola, Mirinda, Sprite, Bahardar University, Gondar University, Fanta, One Water, Pepsi, Tena Oil, Top Water, Yes Water. The all-out gathered dataset is separated into fifteen training sets, approval, and test informational collection. Diverse division strategies are utilized to part the dataset into subsets. For instance, 70%, 15%, and 15 %, for training, approval, and test separately.

Experimentation with this information index is done to track down the best-performing model. The misfortune capacity of the double cross-entropy is utilized with the sigmoid activation work 33 for the yield layer, and the effective Adam inclination plummet procedure is used to become familiar with the loads utilizing the most regular and proficient learning paces of 0.001 and 0.0001. The primary layer is a convolution layer called a Conv2D. The layer has 4 element maps, every one of which has a size of 3x3 and a rectifier initiation work. This is the info layer, which expects

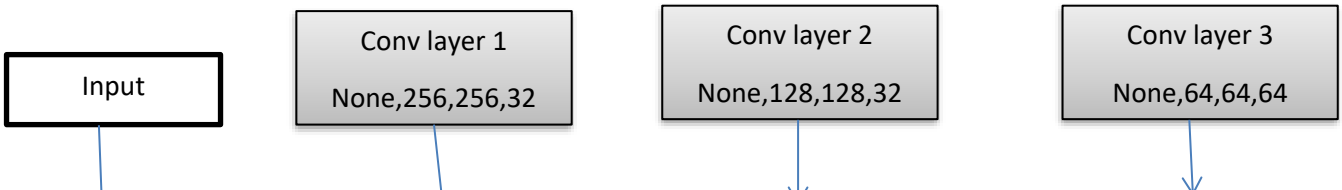
pictures looking like (pixels, width, and height), which I gave picture sizes of 256X256 pixels. The accompanying layer involves a pooling layer named MaxPooling2D that takes the greatest worth. It is set up with a 2x2 foot-size pool.

A convolutional layer with 4 element guides of 3x3 sizes and a rectifier actuation work called ReLU, just as a pooling layer of 2x2. The subsequent layer incorporates a convolutional layer 7 component maps every one of which has a size of 3x3 and a rectifier enactment work, trailed by a maximum pooling layer with a 2x2 pool size.

The third layer has 16 element maps, every one of which has a size 3x3 and a rectifier actuation work, trailed by a maximum pooling layer with a 2x2 pool size. The fourth layer has 32 element maps, every one of which has a size 3x3 and a rectifier initiation work, trailed by a maximum pooling layer with a 2x2 pool size. The fifth layer has 32 element maps as the fourth layer yet in this layer, each element map has a size 5x5 and a rectifier initiation work, trailed by a maximum pooling layer with a 2x2 pool size. The 6th layer has 64 component maps, every one of which has a size 3x3 and a rectifier enactment work, trailed by a maximum pooling layer with a 2x2 pool size.

The seventh layer has 64 element maps as the fourth layer yet in this layer, each component map has a size 5x5 and a rectifier actuation work, trailed by a maximum pooling layer with a 2x2 pool size. The eighth layer has 150 component maps, every one of which has a size 3x3 and a rectifier initiation work, trailed by a maximum pooling layer with a 1x1 pool size. every one of which has a size 1x1 and a rectifier actuation work, trailed by a maximum pooling layer with a 1x1 pool size. The following layer changes the 2D network information over to a vector called to straighten it permits the yield to be handled by standard completely associated layers streamed by a rectifier actuation work.

The following layer is a completely associated layer with 64 neurons with a rectifier initiation work. The yield layer has fifteen neurons for every one of the fifteen classes, just as a Softmax/sigmoid initiation work produces likelihood-like expectations for each class. The model is prepared with the ADAM angle drop procedure and diverse learning rates utilizing two fold cross-entropy as the misfortune work.



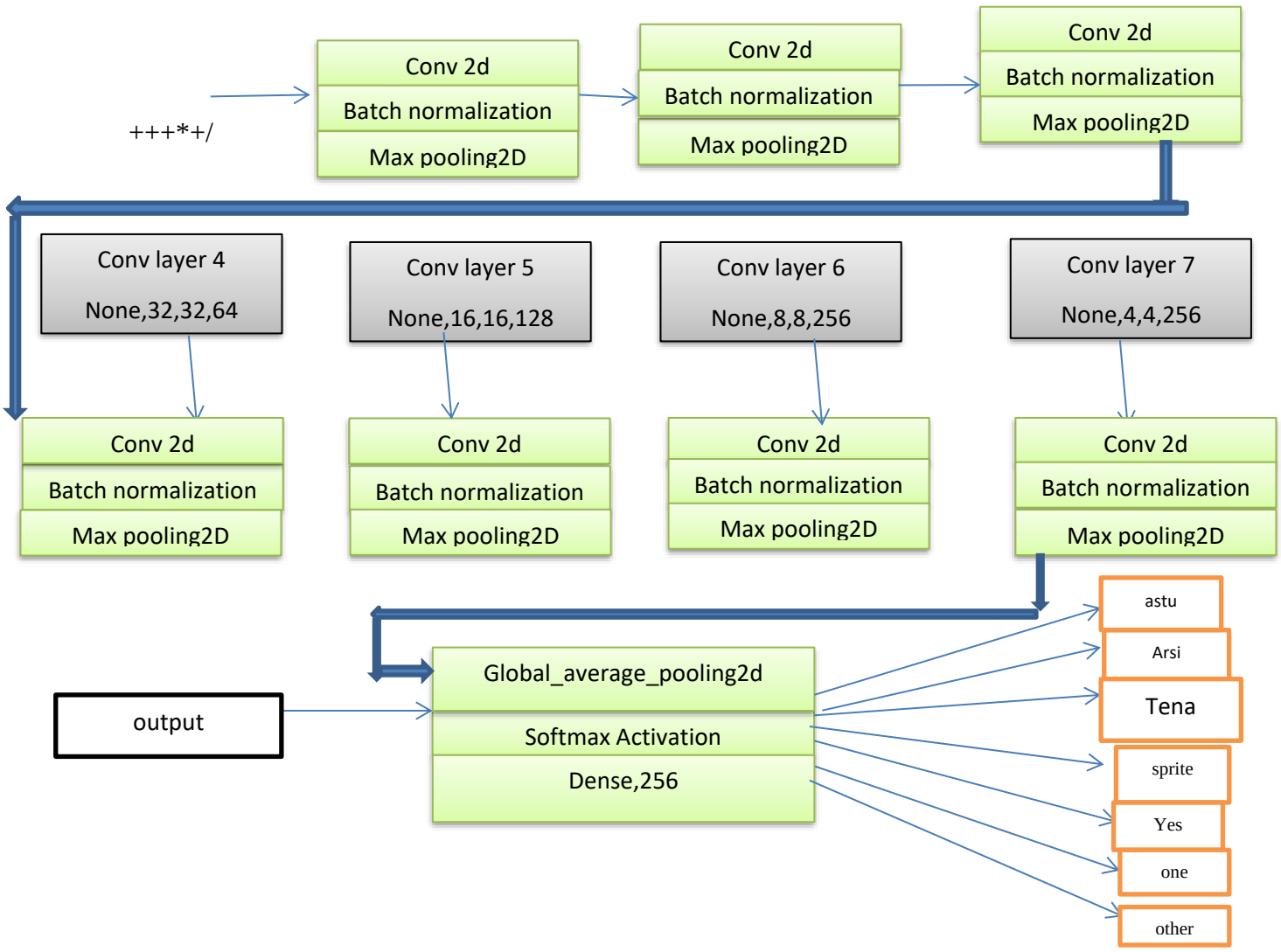


Figure 5:- EthiologoNet Model Architectures

Input Layer

The input layer of our EthioLogoNet model accepts grayscale and RGB images of size 256*256*3 with fifteen different classes are AASTU, ASTU, Arsi University, Arki Water, Coca Cola,

Mirinda, Sprite, Bahardar University, Gondar University,Fanta, One Water, Pepsi,Tena, Top Water, Yes Water.This layer only passes the input to the first convolution layer without any computation. Therefore, there are no learnable features and the number of parameters in this layer is 0.

Convolutional Layer

The convolutional layer is the main and first layer of the CNN architecture. It is used to extract the characters from the input image. The convolution layer maintains the association connects pixels by learning image features from input data. The mathematical operation takes two input image matrices and 3x3 kernel filters. The 5x5 image pixel values and the convolution filter of 3x3. The CNN adds and divides the result by the total number of pixel and then create a map and put the value of the filter at that place. After that, it moves the feature to every other position of the image and obtains the output of the matrix, and repeats the steps for the other filters. This layer moves the filter to every possible position on the image.

Pooling Layer

This layer compresses the image into a smaller size by taking the maximum value from the filtered image and obtains a matrix out of it. It also controls overfitting. In the EthiologoNet model, there are three max-pooling layers.

Fully Connected (FC) Layer

In the EthiologoNet model, there are two fully connected layers including the output layer. The first fully connected layers have 64 neurons each and the final layer which is the output layer of the model has only one neuron. The first FC layer accepts the output of the sevens Convolution layer after converting the 3D volume of data into a vector value (Flattening). This layer computes the class score and the number of neurons in the layer predefined during the development of the model. It is the same as ordinary neural networks and as the name implies, each neuron in this layer is connected to all the numbers of layers.

Output Layer

The output layer is the last (the third FC layer) of the model and it has 1 neuron with a SoftMax activation function. Because the model is designed to classify fifteen classes called categorical classification.

```
conv2d (Conv2D)              (None, 256, 256, 16)      448

max_pooling2d (MaxPooling2D  (None, 128, 128, 16)      0
)

conv2d_1 (Conv2D)            (None, 128, 128, 32)      4640

max_pooling2d_1 (MaxPooling  (None, 64, 64, 32)        0
2D)

conv2d_2 (Conv2D)            (None, 64, 64, 64)        18496

max_pooling2d_2 (MaxPooling  (None, 32, 32, 64)        0
2D)

flatten (Flatten)            (None, 65536)             0

dense (Dense)                (None, 128)               8388736

dense_1 (Dense)              (None, 15)                1935

=================================================================
Total params: 8,414,255
Trainable params: 8,414,255
Non-trainable params: 0
```

Figure 6:- parameters

As we can see in the figures 6 the EthioLogoNet model have 8,414,255 small parameters when we compare to the other deep learning architectures such as AlexNet which have 60 million parameters, VGG 138 million parameters, and GoogLeNet 4 million parameters; therefore, they need a huge computational power to train those models from scratch and they need a very large amount of data. But the EthioLogoNet model is trained with a minimum amount of resources and data and it is overall performance was well.

Feature Extraction

Feature extraction is the main stage of image classification. The important features that are used to classify the image are extracted by the CNN algorithm. In the existing system using some features parameters like color feature, this feature is considered because their visual color difference identifies whether the crop is infected by the disease or not by a human vision in the previous system. In this proposed model it automatically extracts features from sunflower leaf.

Deep learning enables machines to learn from experience and understand. Deep learning learns pathogens and classifiers naturally not like machine learning.

Dataset

The data has been collected from a different place as discussed in a methodology.

The followinga are the sample data we collected from websites.

Figure 7:-sample image from dataset

Optimization algorithms

EthiologoNet model is prepared to utilize a slope plunge advancement calculation to limit the mistake rate and the back-proliferation calculation is utilized to refresh the loads. Slope plunge is the most famous and the most generally utilized enhancement calculation in profound learning investigates. Simultaneously, every cutting-edge Deep Learning library contains executions of angle plummet enhancement calculations like Keras (utilized in this postulation). It refreshes the heaviness of the model and tunes boundaries, accordingly, limit the misfortune work. The Adaptive Moment Estimation (Adam) streamlining agent is utilized to advance the slope drop. The most well-known advancement calculation utilized today for preparing profound neural organizations is Adam to processes versatile learning rates.

Activation function

Experiments are conducted by using two different activation functions. Softmax and Sigmoid are used the EthioLogoNet model. Especially, in the output layer of the model the sigmoid activation functions.

Epochs

It is the number of times the whole training data is shown to the network while training. It is the number of iterations the entire training dataset passes forwarded and back warded through the model or the network. This model was trained by using different epochs starting from 20,50,75 and 100. This was done to get the best optimal epoch.

Batch size

It is the total number of training input passed into training. The single set is divided into small subsets called mini-batches to update the parameters. Though, to prevent any loss of accuracy and keep the generality of the model, the model is trained in the whole dataset.

Learning rate

Back-propagation was used to train the EthiologoNet model the learning rate is used during weight updates. It controls the amount of weight to update during back-propagation. The challenging part was to choose the proper learning rate during the experiment. The learning rates with a value of too small take longer to train than a value of larger. But when given a smaller value the model is more optimal than a model with a learning rate of larger. The experiment was done by using a learning rate of 0.01, 0.001, and 0.0001. This was done to pick the best optimal learning rate.

Table 2 summary of hyper parameters used training model

Parameter	Value
Activation function	Softmax, sigmoid,Relu
Optimization algorithm	Adam
Loss function	BCE
Epoch	50,75,100
Batch size	64
Learning rate	0.0001

Hyper-parameters for EthiologoNet Model

All the convolutional neural network models were employed using various parameters. These parameters make so many possible changes in the architecture. Training them with the huge dataset needed quite a long time. Performing hyper parameter optimization is used to makes efficient

computing with available resources. Thus, we utilized some common hyper parameter values and looked further into techniques to achieve a better model evaluation. The following are hyper-parameters that are choosen for the EthioLogoNet model.

Deep learning architecture

- EthiologoNet

Training mechanism

- Training from Scratch

Training testing set distribution

- Train: 80%, Test: 10%, valid: 10%
- Train: 70%, Test: 15%, valid 15%
- Train: 60% Test: 20% , valid 20%

Data set type

- RGB Original, gray scale and Augmented image

5.1 INTRODUCTION

This topic describes how the entire implementation was done. In this chapter, all the process passed to get the research result is described in detail. It contains data preparation, data augmentation, data split, and image processing, and building the model.

5.2 Data preparation

Since the images had different proportions, their size was modified to a 256x256 pixel proportion. This resolution was the same satisfactory for both maintaining the image features, as well as keeping computational time to a minimum. All images were originally saved in an RGB 3-channel. To achieve the efficient model as required the first thing that must be done is sort outing the data set. Before train the model it is important to well a proper sufficient data set. Originally the collected images from the fields are not enough to use for deep learning. So, the first thing done to increase the data set size is using a Data Augmentation. This is used to enhance the size of the original data. For this study, the collected original data is a total of 3622 images. This data is augmented to around 14,488 images by applying 4 (four) feature factors.

These original data images are multiplied using different features through augmentation processes. Sample of augmented images are the following

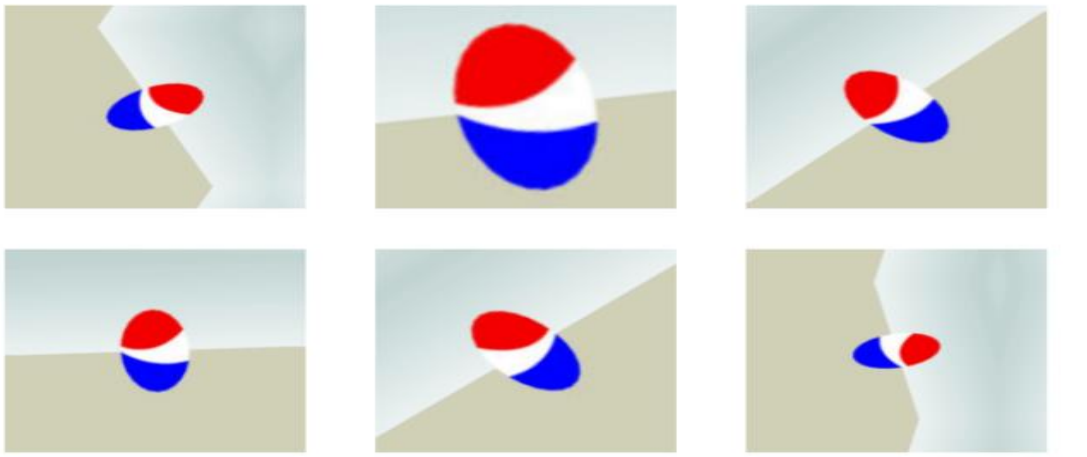


Figure 8:- Sample of augmented image

Sample code of data augmentation

```
datagen = ImageDataGenerator(
        rotation_range=40,
        width_shift_range=0.2,
        height_shift_range=0.2,
        shear_range=0.2,
        zoom_range=0.2,
        horizontal_flip=True,
        rescale=1./255,
    data_format='channels_last',
     brightness_range=[0.5,0.5],
        fill_mode='nearest')
```

5.3 Image Preprocessing

All pictures in the preparation information, they have an inadequate nature with various measurement size, hence it needs to standardize to a similar picture measurement, which is making the preparation information normalized to a similar picture measurement size. Subsequently, I have utilized an element of 256X256 for all pictures in the preparation set. Pictures are stacked recursively dependent on their pathname. Contingent upon the prefix of their filename, a name worth '0' or '1' is affixed to each stacked picture. Shading change Two capacity for shading transformation were carried out in this proposal a transformation from the BGR to RGB shading space, where the request for the picture's shading space, where the pictures lose all the shading data and keep up with just the splendor and immersion for every pixel. Information standardization is accomplished by discovering the distinction of the mean from each pixel, after that isolating the result by the standard deviation.

The dataset values are standardized inside the (0, 1) territory to guarantee that the misfortune work, which is typically not curved, tracks down the worldwide least as effectively as could be expected. Diminishing the scope of contribution for preparing values additionally helps the union of the back-proliferation calculation. Dataset rearranging The dataset is being rearranged by python's haphazardly Shuffle strategy, which is a request rearranging calculation dependent on an arbitrary number generator. After the application, the request for the pictures is initially was in sequential order, presently blended all through the dataset.

Dataset parting the dataset is being parted in the accompanying design First 15% of the all-out number of the dataset is held out and set as a testing dataset, second 15% of the dataset is held out and set the approval set, then, at that point, the leftover dataset is parting again the 80/20 style,

where the 20% piece is utilized as the approval dataset. The rest is utilized for preparing the dataset calculation. The preparation and approval sets are utilized for the preparation of the calculation, while the testing dataset is utilized solely after the classifier is prepared, to make forecasts.

Data Split

The dataset was split into three sections A training portion dedicated to training the classifier a validation portion that would allow the training process to improve and a testing portion that is completely hidden from the training process and will be used for validating the classifier on unknown data. All split create that contain 80/20 ratio of the class of logo image to avoid the unnecessary bias that would incur if a dataset would contain images from only one category

The first data split takes place by removing 10% of the total dataset and saving it for later uses as testing. The remaining 90% of the dataset is then split again into 80/20 ratio, resulting in the following portion

Table 3 dataset splitting percentage

Dataset	Total	Training	Testing	Validation
Percentage of Total dataset	100%	80%	10%	10%
Total number images	14,488	11,590	1,449	1,449

Table 4 dataset splitting percentage

Dataset	Total	Training	Testing	Validation
Percentage of Total dataset	100%	70%	15%	15%
Total number images	14,488	10142	2173	2173

Table 5 dataset splitting percentage

Dataset	Total	Training	Testing	Validation

Percentage of Total dataset	100%	60%	20%	20%
Total number images	14,488	8693	2897	2897

5.4 Build EthiologoNet Model

5.4.1 Parameters for layers

Convolutional layer filter size filters the number of filters should depend on the complexity of the dataset and the depth of the neural network. A common setting to start with is [256, 128, 64, 32, 16, 8, 4] for seven layers. Kernel size number of filters a small window of pixels at a time (3x3) which will be moved until the enter image is scanned. If the image is smaller than 256X256, work with a smaller filter of 1x1; the Width and Height of images were already providing. 2D convolutional layers take a three-dimensional input, typically an image with three color channels such as RGB Max pooling is used which further reduces the size of the convolved image and prevents over fitting. It is the application of a moving window across a 2D input space, where the maximum value within that window is the output 2x2.

5.4.2 Parameters to fit the model

Epoch is the number of passes of the entire training data set when an entire dataset is passed forward and backward through the neural network. It describes the number of images (items) and the algorithm sees the entire data set and the algorithm has seen all samples in the training dataset. The number of epochs used is 100. This number of epochs is enough scope for the loss function to converge if it is possible. This also means that we are giving enough time to the architecture to learn. It takes around 2 minutes per epoch which is 2*100=200 minutes. A single number of epochs are too big to feed to the computer at once so divide it into several smaller batches. In the experiment, it tried in different batches to come up with the best learning.

EthiologoNet or CNN is a class of deep neural networks, commonly applied to analyzing visual imagery. Here is the EthiologoNet model's example of CNN with few layers using Conv2D -2D convolutional layer, Batch Normalization, and MaxPooling2D application of a moving window across 2D input space. Hyper parameters are set before training they represent the variables that determine the neural network structure and how it is trained. Conv2D with hyper parameters

mentioned above Conv2D (kernel size, (filters, and filters), input shape (256, 256, and 3) with activation function for each layer as a ReLU.

- MaxPooling2D layer to reduce the spatial size of the incoming features
- 2D input space MaxPooling2D ((2, 2)) and ((2, 2))
- Do the same increasing the kernel size 4-8-16-32-64-128-256
- Provide the last fully connected layer which specifies the number of classes of EthiologoNets.
- Print the summary of the model.

Chapter Six Result And Discussion

6.1 Experiment Result

Afterward performing all the evaluation on the Grayscale and RGB segmented dataset on the EthiologoNet model, experiments with top results are discussed below. The EthiologoNet model is evaluated using different parameters that can affect the efficiency of the model. These parameters are several epochs; train test split ratio, learning rate, and dropout ratio which are explained concisely in previous chapters. Different parameter ratios are used for the evaluation purpose and the most successful ratios are discussed in this chapter.

In this thesis, three questions stated in the problem statement part are tried to be answered. To select the best model to solve the stated problem, various experiments are conducted using the CNN method. Using this method, the EthiologoNet model is build using 10,141 (ten thousand and one hundred and fourth one) training samples from 14488(fourth thousand and four hundred and eighty eight) total data set. This is 70% of the total dataset. Model fit is implemented using 4347(four thousand and three hundred and fourth seven)

validation (testing) dataset from 14488(fourth thousand and four hundred and eighty eight) total dataset. This is 15% of the total dataset. In another experiment, the data was divided to 80% for training samples from the total dataset, 20% for test and validation data samples. To come up with the best model the third try of the experiment was dividing the dataset into 60% for training samples and 40 % testing and validation data samples. The lack of a globally acknowledged standard ratio for separating the training and test data sets is the reason behind this. The EthiologoNet model was built using RGB image training datasets and grayscale image datasets. Performances of both models are evaluated against validation datasets. The different scenarios are used in an experiment in addition to dividing datasets into training and testing subsamples. The result obtained from the experiment done using RGB images is discussed as follows in different perspectives. The experiment was done in different data splits as mentioned above. These are 60-40, 70-30, and 80-20 the experiments are done accordingly. Using this different data partition, the result obtained is also different as listed in the following table

Table 5 expermental Result

No	Training samples	Testinga and validation samples	accuracy	Loss	epoch
1	60%	40%	96.5	0.01	100
2	70%	30%	96.99	0.21	100
3	80%	20%	98.14	0.06	100

6.2 Result of Experiment done on RGB data images

Multi-channel images are displayed using the colors in RGB images (24-bit with 8-bits for each red, green, and blue channel). Colors have been chosen to reflect natural hues (i.e., the green in an RGB image reflects the green color in the specimen). As stated previously, this thesis is based on RGB and gray scale data images. The experiment is carried out utilizing various circumstances to improve the EthiologoNet model's performance.

6.3 Experiment result using output Activation Function

The Sigmoid Activation Function is a mathematical function that has a distinct "S" curve. It is mostly used for logistic regression and the creation of simple neural networks. This function is used separately for each element's raw output in this thesis experiment. Softmax is a technique for mapping non-normalized data output to a probability distribution for output classes. In neural network-based classifiers, it is used in the final layers. They are trained under either the log-loss or cross-entropy regime. Softmax is often used in artificial and convolutional neural networks. These activation functions are used in the EthioLogoNet model.

6.4 Experiment result using various Learning Rates

The learning rate is a hyper-parameter that governs how much the weights of our network are adjusted about the loss gradient. A value too little may result in a long training process that becomes stuck, whereas a value too large may result in learning a sub-optimal set of weights too quickly or an unstable training process. The stochastic gradient descent algorithm estimates the error gradient for the current state of the model using examples from the training dataset, then No Training samples Test and validation samples Accuracy Loss Epochs 60% 40% 95.8% 0.12 100 70% 30% 96.99% 0.011 100 80% 20% 98.14% 0.07 100 updates the weights of the model using the back-propagation of errors process, sometimes known as simply back-propagation. The step size, often known as the learning rate is the amount by which the weights are changed during training. The learning rate is an adjustable hyper parameter that has a modest positive value, usually between 0.0 and 1.0 and is used in the training of neural networks. This experiment was done by using a learning rate of 0.1, 0.01, 0.001 and 0.0001.

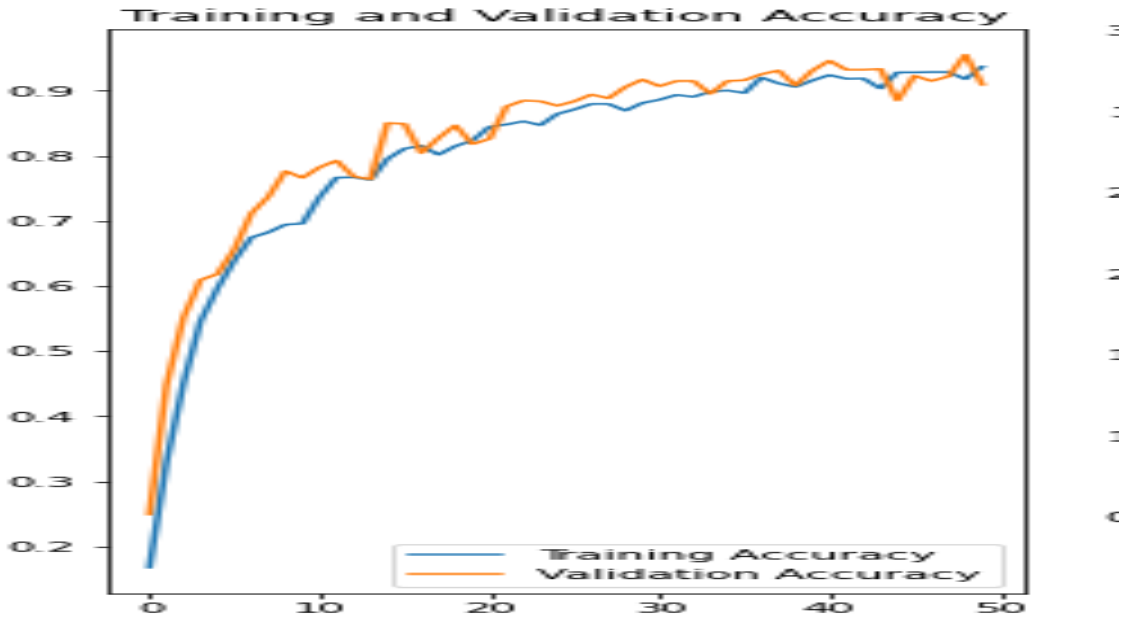


Figure 9:- Training And Validation Accuracy Analysis for 70:30% data split ratio 50 epochs

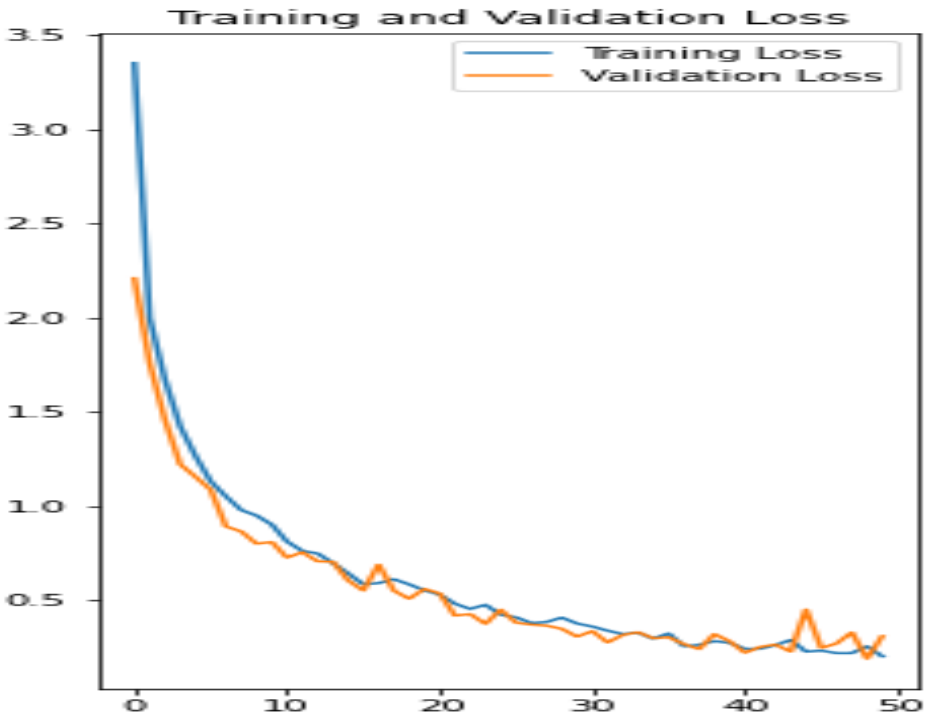


Figure 10:- Training And Validation Loss Analysis for 70:30% data split ratio 50 epochs

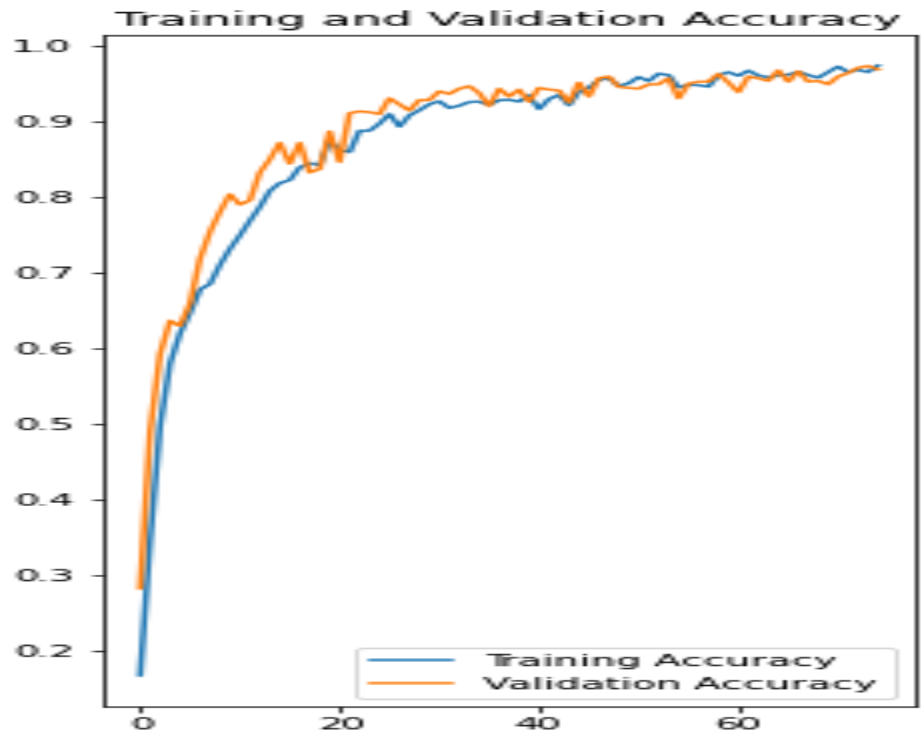


Figure 11:- Training And Validation Accuracy Analysis for 70:30% data split ratio 75 epochs

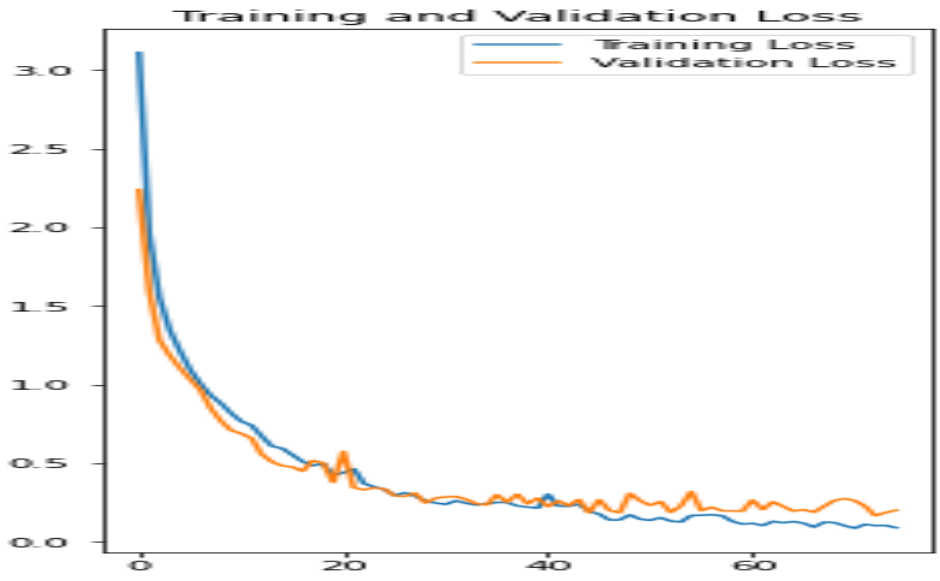


Figure 12:- Training And Validation Loss Analysis for 70:30% data split ratio 75 epochs

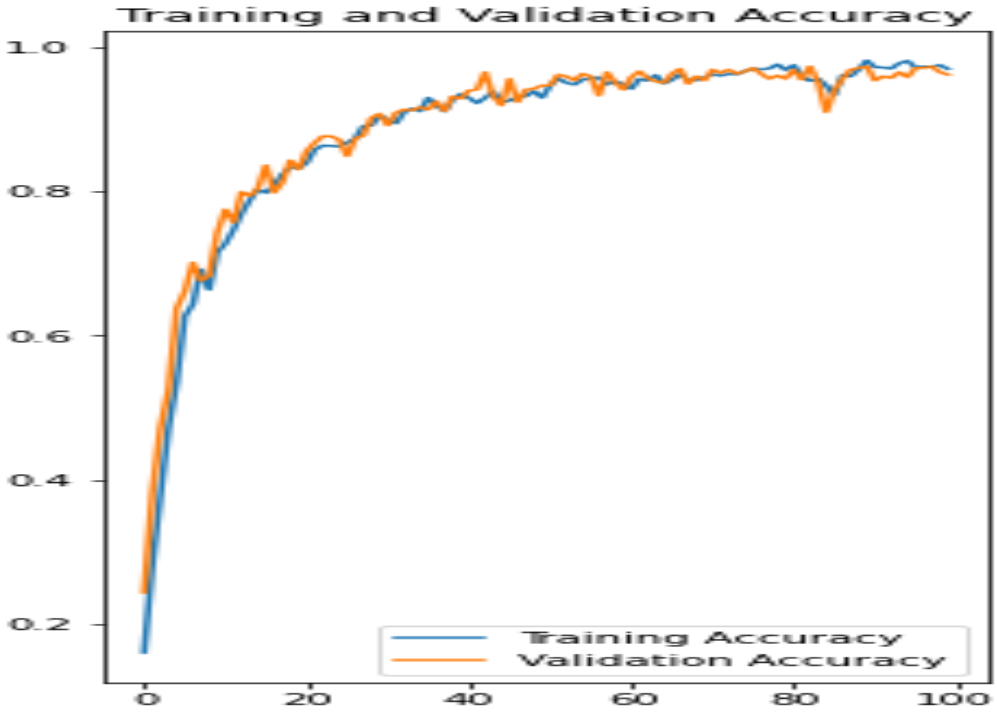


Figure 13:- accuracy and loss Analysis for 70:30% data split ratio 100 epochs

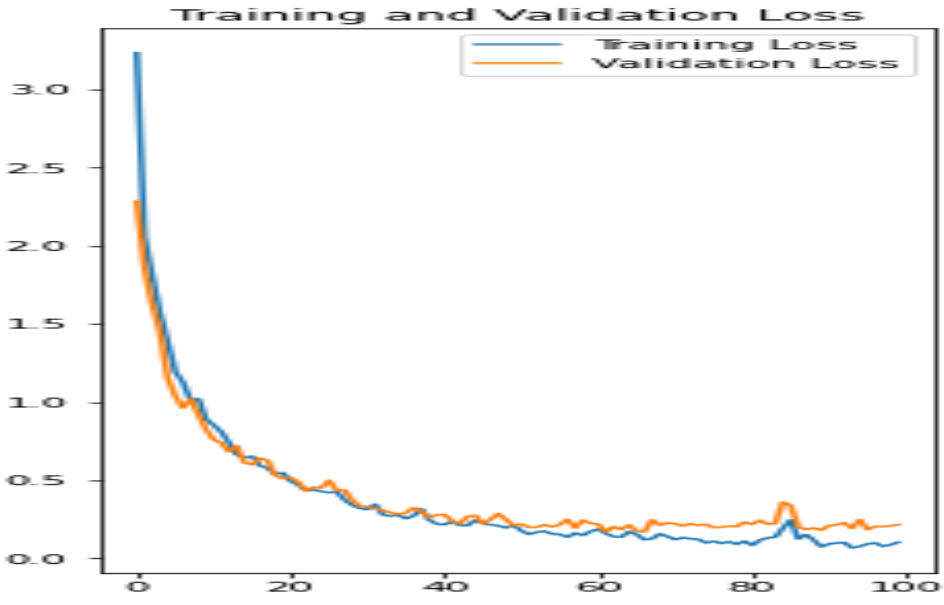


Figure 14:- Training And Validation Loss Analysis for 70:30% data split ratio 100 epochs

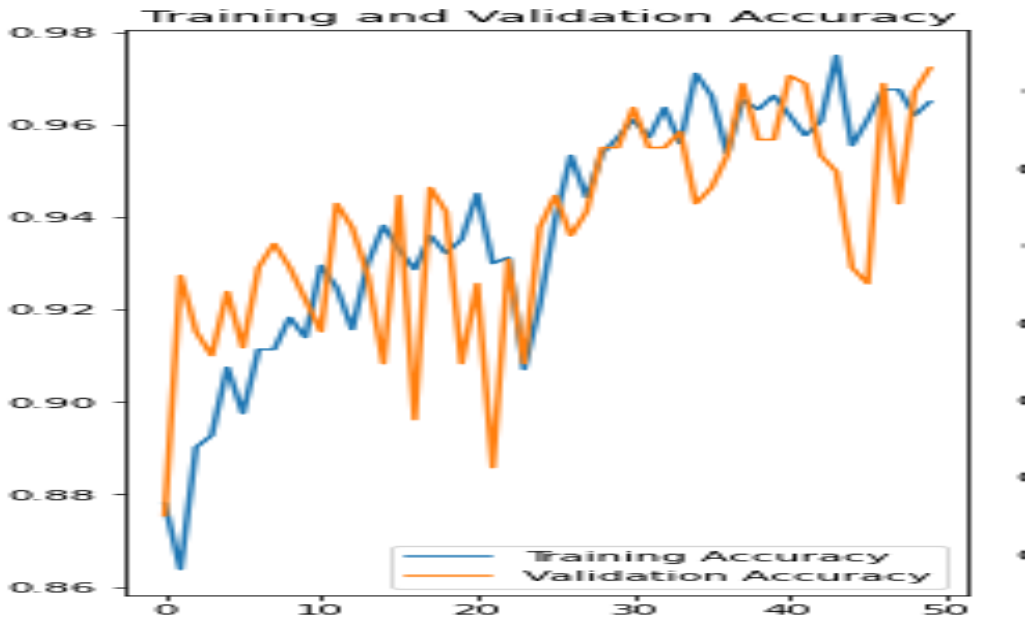


Figure 15:- Training And Validation Accuracy Analysis for 80:20% data split ratio 50 epochs

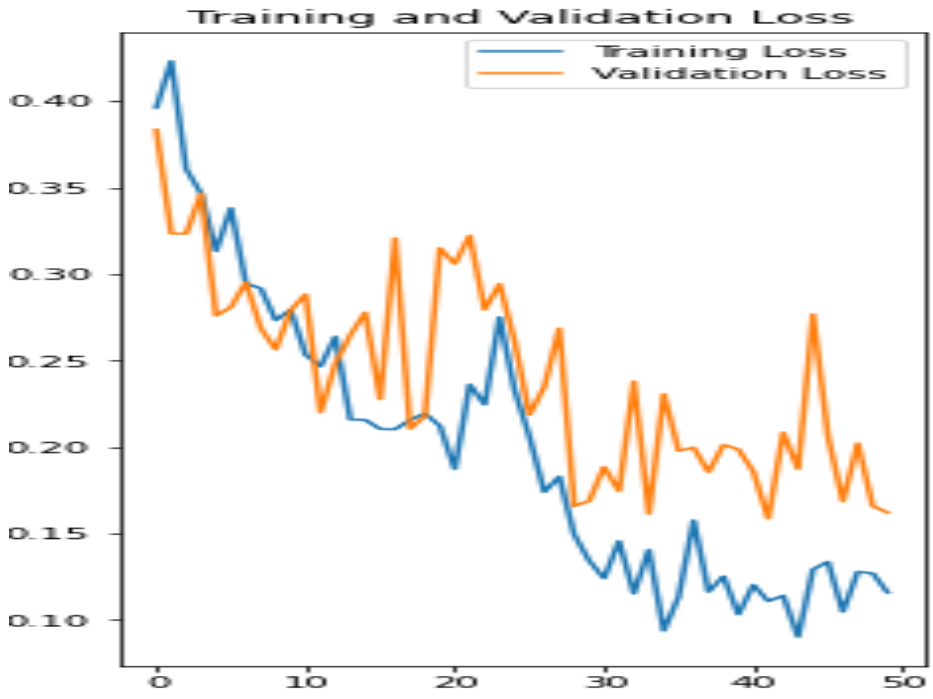


Figure 16:- Training And Validation loss Analysis for 80:20% data split ratio 50 epochs

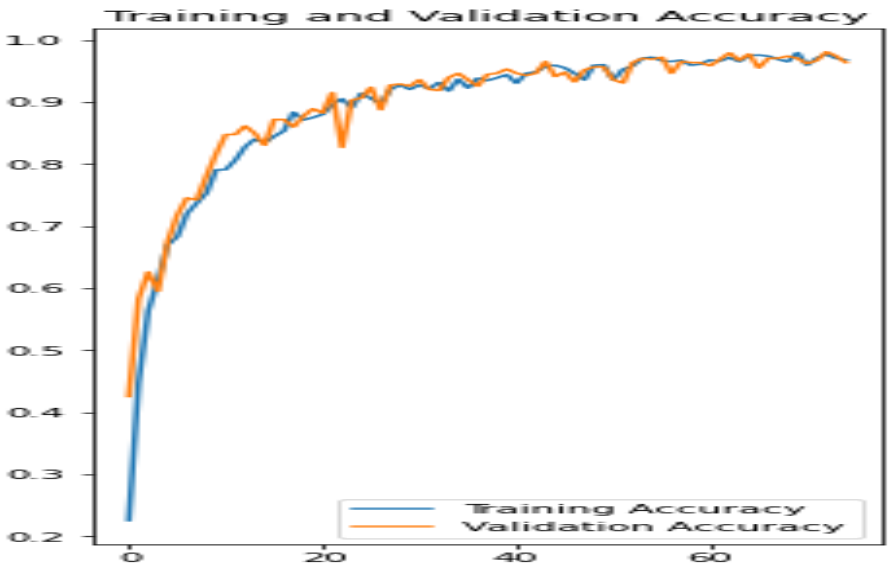


Figure 17:- Training And Validation Accuracy Analysis for 80:20% data split ratio 75 epochs

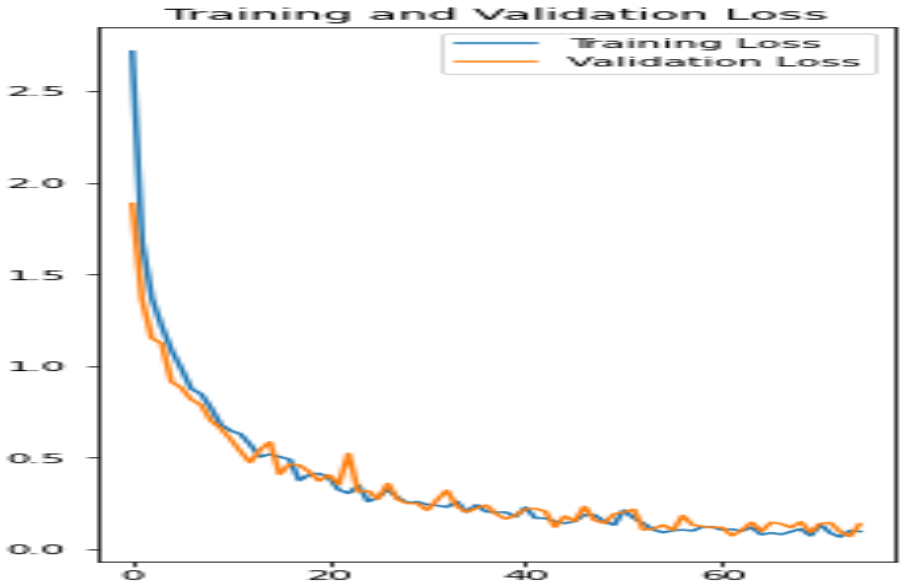


Figure 18:- Training And Validation loss Analysis for 80:20% data split ratio 75 epochs

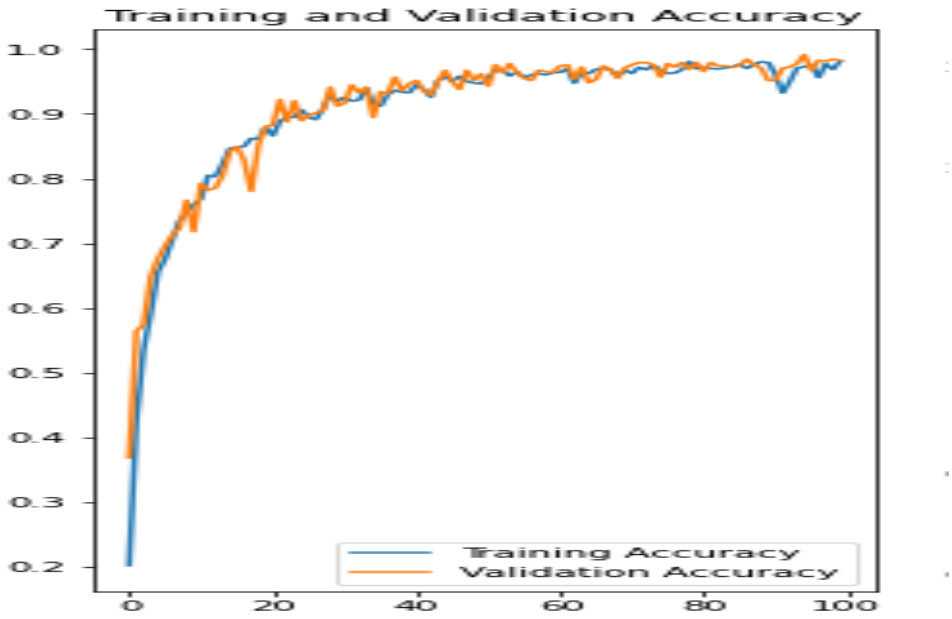


Figure 19:- Training And Validation Accuracy Analysis for 80:20% data split ratio 100 epochs

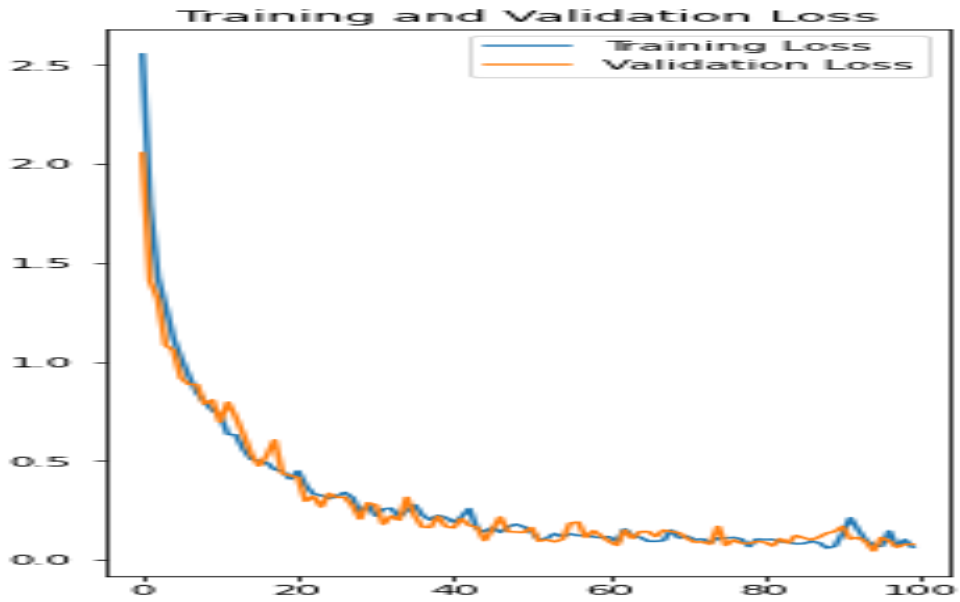


Figure 20:- Training And Validation loss Analysis for 80:20% data split ratio 100 epochs

Table 5 Result of an experiment by using different training and validation dataset ratio

No	Ratio	Loss	Accuracy	Val-loss	Val-accuracy	epochs

1	60:40%	0.01	95.1	0.16	95.42	50
2	60:40%	0.1	95.6	0.11	95.5	75
3	60:40%	0.12	95.8	0.1	96	100
4	70:30%	0.2	93.68	0.3	90.89	50
5	70:30%	0.08	97.43	0.2	96.89	75
6	70:30%	0.1	96.99	0.21	96.19	100
7	80:20%	0.11	96.1	0.08	95.87	50
8	80:20%	0.09	96.8	0.13	96.37	75
9	80:20%	0.06	98.14	0.07	98.27	100

CHAPTER SEVEN CONCLUSION AND FUTURE WORKS

7.1.Conclusions

Logo classification is fundamental in many application domains. The problem is that may prepared fake logos specially in poor country like Ethiopia. To make early identification of the fake logo and trademark, we have EthioLogoNet and implemented a deep learning approach by using the Convolutional Neural Network and image processing algorithm. We have presented a EthioLogoNet model to detect and classify different logo and trademark for classifying fake or genuine images of the logos as an input. The EthioLogoNet model can be used as a tool to detect fake logos.

This study discussed the CNN model by applying different important factors that can affect the model designed in the study. Three dataset types are used in the study to conduct the experiments for the EthioLogoNet model, and this data set types are: RGB data set, a data set with threechannel images Grayscale image data set a data set which only contains one channel images, and augmented image dataset a data set which is RGB and applying different augmented technics such as rotation, zoom, width shift, height shift, shear range, fill mode, brightness range, vertical flip, and horizontal flip. EthioLogoNet model has achieved an accuracy of 98.14% with 100 epochs. 0.0001 learning rate using grayscale image data set. This result is improved when the model is trained on the RGB data image data set, which climbed to an accuracy of 98.14% finally after the augmented image dataset is archived belter performance than the other archived 98.14% with 100 training epochs, the learning rate of 0.0001.

Finally, the most important factors that have impacts on the efficiency and accuracy of the model are several training epochs and learning rate the number of epochs is the number of iterations the model learns the whole data set. It is nor fixed that is how many epochs a model should be trained, but it obvious that the model learns more as it is trained more and more having a limit to stop somewhere and depends on the model architecture, type, and size of the data set. This study is archived with the highest accuracy with 100 epochs for the EthioLogoNet model.

7.2. future works

In the future the progress of identifying the fake logo and trademark has been further modified to analyse a multiple leafs and various part of the logo and trademark.

- Reduce the computation and the size of deep model for small machine.

- Further improvement can be made to the neural network system to factor dataset scalability and more environmental diversities.

- Finally, it would be interesting to use the neural network system to segment contours a given image of our dataset in the future work more images of different logo and trademark needs to be collected.

References

[1] R. Boia, A. Bandrabur, and C. Florea, “Local description using multiscale complete rank transform for improved logo recognition,” in 2014 10th International Conference on Communications (COMM), May 2014, pp. 1–4, doi: 10.1109/ICComm.2014.6866723.

[2] S. Bianco, M. Buzzelli, D. Mazzini, and R. Schettini, “Deep Learning for Logo Recognition,” Neurocomputing, Jan. 2017, doi: 10.1016/j.neucom.2017.03.051.

[3] F. N. Iandola, A. Shen, P. Gao, and K. Keutzer, “DeepLogo: Hitting Logo Recognition with the Deep Neural Network Hammer,” arXiv:1510.02131 [cs], Oct. 2015, Accessed: Aug. 12, 2020. [Online]. Available: http://arxiv.org/abs/1510.02131.

[4] Anil K. Jain, Robert P.W. Duin, (2004). "Introduction to pattern recognition". The Oxford Companion to the mind, Second Edition, Oxford University Press, Oxford, UK.(pp 698- 703).

[5] Dutt. V, Chaudhry. V, Khan. I, (2012). "Pattern Recognition: an Overview ". American Journal of Intelligent Systems .(pp 23-27).

[6] Editorial, (2005). "Advances in Pattern Recognition" ,Pattern Recognition Letters 26, (pp 395-398).

[7] Gonzalez,R.C.Thomas,M.G . (1978). "Syntatic Pattern Recognition: an Introduction", Addison Wesley,Reading,MA,

[8] Watanabe , (1985)." Pattern Recognition: Human and Mechanical". Wiley, New York .

[9] K. Fukunaga. (1990). "Introduction to statistical pattern recognition". Academic Press, Boston .

[10] RJ Schalkoff ,(1992)."Pattern Recognition: Statistical, Structural and Neural Approaches". John Wiley & Sons

[11] Srihari, S.N., (1993) . "Covindaraju, Pattern recognition", Chapman &Hall, London, 1034-1041.

[12] B. Ripley, (1996)," Pattern Recognition and Neural Networks" , Cambridge University Press, Cambridge,

[13] Robert P.W. Duin, (2002). "Structural, Syntactic, and Statistical andPattern Recogition" : Joint Iapr International Workshops Sspr 2002 and Spr 2002, Windsor, Ontario, Canada, August 6-9,2002 Proceedings

[14] Sergios Theodoridis, Konstantinos Koutroumbas, (1982)." Pattern recognition " ,Elsevier(USA).

[15] Rafael C. Gonzalez, Richard E. Wood, (2002), "Digital Image Processing", Third Edition.

[16] "Introduction to Image Processing". http://www.engineersgarage.com/articles/image-processing-tutorialapplications

[17] Sahel et al(2021), “Logo Detection Using Deep Learning with Pretrained CNN Models”

[18] https://medium.com/analytics-vidhya/image-classification-techniques-83fd87011cac

[19] K. Simonyan and A. Zisserman,(2014). “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556,

[20]VGG16 – Convolutional Network for Classification and Detection"https://neurohive.io/en/popular-networks/vgg16/"

[21] P. A. Miceli, W. D. Blair, and M. M. Brown, *Isolating Random and Bias Covariances in Tracks*. 2018.

[22] Vitaly Bushaev, ―Understanding RMSprop — faster neural network learning,‖ *Towards Data Science*, 2018. [Online]. Available: https://towardsdatascience.com/understanding-rmsprop-faster-neural-network-learning-62e116fcf29a.

[23] Gandhi rohith, ―A Look at Gradient Descent and RMSprop Optimizers – Towards Data Science,‖ 2018. [Online]. Available: https://towardsdatascience.com/a-look-at-gradient-descent-and-rmsprop-optimizers-f77d483ef08b

[24] I. Changhau, ―Loss Functions in Neural Networks,‖ 2017. [Online]. Available: https://isaacchanghau.github.io/post/loss_functions/.

[25] B. J. Mccaffrey, ―Neural Network Cross Entropy Error,‖ 2014. [Online]. Available: https://visualstudiomagazine.com/articles/2014/04/01/neural-network-cross-entropy-error.aspx.

[26] A. K. Singh, B. Ganapathysubramanian, S. Sarkar, and A. Singh, ―Deep Learning for Plant Stress Phenotyping: Trends and Future Perspectives,‖ *Trends Plant Sci.*, vol. 23, no. 10, pp. 883–898, 2018.

[27] H. Wehle, ―Machine Learning – Artificial Intelligence- COGNITIVE,‖ July, 2017.

[28] Sambit Mahapatra, ―Why Deep Learning over Traditional Machine Learning?,‖ *Towards Data Science*, 2018. [Online]. Available: https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063.

[29] S. Albawi, T. A. Mohammed, and S. Al-Zawi, ―Understanding of a convolutional neural network,‖ *Proc. 2017 Int. Conf. Eng. Technol. ICET 2017*, vol. 2018–Janua, no. April 2018, pp. 1–6, 2018.

[30] M. Mishra, ―Convolutional Neural Networks,‖ *Medical Imaging and Information Sciences*, 2017. [Online]. Available: https://www.datascience.com/blog/convolutional-neural-network.

[31] R. Prabhu, ―Understanding of Convolutional Neural Network (CNN) Deep Learning,‖ *Medium.Com*, 2018. [Online]. Available: https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148.

[32] J. Brownlee, ―A Gentle Introduction to Pooling Layers for Convolutional Neural Networks,‖ *Deep Learning for Computer Vision*, 2019. [Online]. Available: https://machinelearningmastery.com/pooling-layers-for-convolutional-neural-networks/.

[33] V. Zhou, ―An Introduction to Convolutional Neural Networks,‖ 2019. [Online]. Available: https://victorzhou.com/blog/intro-to-cnns-part-1/.

[34] R. Venkatesan, B. Li, R. Venkatesan, and B. Li, ―Convolutional Neural Networks,‖ *Convolutional Neural Networks in Visual Computing*, 2018. [Online]. Available: https://divishd.wordpress.com/2017/08/24/convolutional-neural-networks-cnn-applications-in-nlp-for-sentence-classification/.

[35] N. C. Steele, C. R. Reeves, and E. I. Gaura, ―Activation Functions,‖ *Artificial Neural Nets and Genetic Algorithms*, 2001. [Online]. Available: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6.

[36] J. Brownlee, ―How to Avoid Overfitting in Deep Learning Neural Networks,‖ 2018. [Online]. Available: https://machinelearningmastery.com/introduction-to-regularization-to-reduce-overfitting-and-improve-generalization-error/.

[37] N. Schlüter, ―How to prevent Overfitting in your Deep Learning Models.‖ [Online]. Available: https://towardsdatascience.com/dont-overfit-how-to-prevent-overfitting-in-your-deep-learning-models-63274e552323.

[38] R. Zadeh and B. Ramsundar, ―Tensorflow for Deep Learning,‖ 2018. [Online]. Available: https://www.oreilly.com/library/view/tensorflow-for-deep/9781491980446/ch04.html.

[39] Tensorflow, ―Tensorflow.‖ [Online]. Available: https://www.tensorflow.org/tutorials/.

[40] Keras, ―Keras.‖ [Online]. Available: https://keras.io/.

[41] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, ―Deep Learning for Image-Based Cassava Disease Detection,‖ *Front. Plant Sci.*, vol. 8, no. 2002, pp. 1–10, 2017.

[42] Google, ―Classification: True vs. False and Positive vs. Negative,‖ *Machine Learning Crash Course Google Developers*, 2019. [Online]. Available: https://developers.google.com/machine-learning/crash-course/classification/true-false-positive-negative.

[43] G. Battineni, G. G. Sagaro, N. Chinatalapudi, and F. Amenta, “Applications of machine learning predictive models in the chronic disease diagnosis,” *J. Pers. Med.*, vol. 10, no. 2, 2020, doi: 10.3390/jpm10020021.

[44] A. Acharya, “Comparative Study of Machine Learning Algo- rithms for Heart Disease Prediction,” no. April, 2017.

[45] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

[46] A. Abdelaziz, M. Elhoseny, A. S. Salama, and A. M. Riad, “A machine learning model for improving healthcare services on cloud computing environment,” *Meas. J. Int. Meas. Confed.*, vol. 119, pp. 117–128, 2018, doi: 10.1016/j.measurement.2018.01.022.

[47] Y. Amirgaliyev, “Analysis of Chronic Kidney Disease Dataset by Applying Machine Learning Methods,” *2018 IEEE 12th Int. Conf. Appl. Inf. Commun. Technol.*, pp. 1–4, 2010.

[48] Steven C-h. hoi et al(2015), “Large scale deep logo detection and brand recognition with deep region based convolutional network” School of Information Systems, Singapore Management University, Singapore Alibaba Group, Hangzhou, China

[49] Karimi et al(2019) , “Logo recognition by combining deep convolutional models in a parallel structure” 2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA), 6 and 7 March, Tehran, Iran. 978-1-7281-1621-1/19/$31.00 ©2019 IEEE

[50] Alsheikhy et al(2020), “Logo Recognition with the Use of Deep Convolutional Neural Networks” Engineering, Technology & Applied Science Research Vol. 10, No. 5, 2020, 6191-6194

[51] Ping shi et al(2016), “TV Logo Classification Based on CNN” College of Information Engineering Communication University of China Beijing, China 978-1-5090-4102-2/16/$31.00 ©2016 IEEE

[52] Yuanyuan li(2017), “Graphic Logo Detection With Deep Region Based CNN” School of Communication and Information Engineering, Beijing University of Posts and Telecommunications, Beijing, China 978-1-5386-0462-5/17/$31.00 ©2017 IEEE

[53]https://www.google.com/search?q=cnn+architecture+diagram&tbm=isch&ved=2ahUKEwiUvKzgjpH4AhULqRoKHXY0CA0Q2-cCegQIABAA&oq=CNN+architecture&gs_lcp=CgNpbWcQARgBMgQIIxAnMgQIABBDMgQIABBDMgUIABCABDIFCAAQgAQyBQgAEIAEMgUIABCABDIFCAAQgAQyBQgAEIAEMgUIABCABFAAWABgiRdoAHAAeACAAfoBiAH6AZIBAzItMZgBAKoBC2d3cy13aXotaW1nwAEB&sclient=img&ei=--eZYtTSNYvSavbooGg&bih=804&biw=1708&client=firefox-b-d#imgrc=4P670hWCEN9fcM

[54]https://www.google.com/search?q=cnn+architecture+diagram&tbm=isch&hl=en&chips=q:cnn+architecture+diagram,online_chips:convolutional:SCwXHSMLhj8%3D&client=firefox-b-d&sa=X&ved=2ahUKEwiu5a-atZH4AhXD8IUKHZBaAekQ4lYoB3oECAEQKw&biw=1686&bih=804#imgrc=fuFS4foWKzn8LM

[55]https://www.simplilearn.com/image-processing-article

Appendix

A. Sample Dataset

Aastu

arki_water

Arsi_Uni

Astu

bahardar_Uni

Cocacola

Fanta

Pepsi

sprite

top_water