NATIONAL FLAG DETECTION USING CNN

BY

SUMAIYA SULTANA
ID: 151-15-4949

RAKIBUL ISLAM
ID: 151-15-5131
AND

ABU SHAIF KHAN
ID: 151-15-5207

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Md Zahid Hasan

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised By

Abdus Sattar

Senior Lecturer

Department of CSE

Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2018

APPROVAL

This Project/internship titled **"National Flag Detection Using CNN**", submitted by SUMAIYA SULTANA, ID No: 151-15-4949, RAKIBUL ISLAM, ID No:151-15-5131 and ABU SHAIF KHAN ID No:151-15-5207 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 09 December,2018.

BOARD OF EXAMINERS

____

Dr. Syed Akhter Hossain **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Narayan Ranjan Chakraborty **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Md. Tarek Habib **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Dr. Mohammad Shorif Uddin **External Examiner**
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Md Zahid Hasan, Assistant Professor of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Md Zahid Hasan
Assistant Professor
Department of CSE
Daffodil International University

Co- Supervised by:

Abdus Sattar
Senior Lecturer
Department of CSE
Daffodil International University

Submitted by:

Sumaiya Sultana
ID: 151-15-4949
Department of CSE
Daffodil International University

Rakibul Islam
ID: 151-15-5131
Department of CSE
Daffodil International University

Abu Shaif Khan
ID: 151-15-5207
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Md Zahid Hasan, Assistant Professor,** Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Deep Learning*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to the Almighty Allah and Head**,** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

This research project focus on "**National Flag Detection Using Convolutional Neural Network**". This is a kind of image processing work with the help of convolutional neural network. In this project we solve National Flag classification problem, where our goal will be to tell which flag the input image belongs to. We are achieved it by training an artificial neural network on few thousand images of national flag. We make the Neural Network learn to predict which class the image belongs to. Finally, we achieve our goal for detection of national flag using Artificial Neural Network when corresponding national flag is provided. We work with more than 5,000 images. Finally, after developing the project we are able to obtained a satisfactory accuracy that is 98%.

TABLE OF CONTENTS

CONTENTS	PAGE NO

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Image Classification has been an interesting issue for a long while. It's overwhelming applications in modern technologies and daily life have been the motivation for continuous research. Last couple of years, bunches of work are being done in dependent on image processing and image classification. The present world is depending on data. Images are one kind of data. An exceptional type of application but related to our day to day life is detecting some image. For an example, if we see a flag of a country, we can't able to recognize it and similarly our machine can't also detect any country's flag. Regardless of where you go, you can simply discover various types of flags and you will see some flags from company's corporations. Flags have their own meanings and different country's flags hold history and memory of significant people and events. So identifying national flag has enormous importance to every people.

We know that technology is getting exponentially developed. One gift of the ongoing vast development of computational power has empowered the use of Artificial Intelligence. In particular, Neural Network can be conveyed to make national flag detectable.

Flags are not designed to be used basically with automatic identification purposes. Normally flags have that attributes those are used for identification on a national basis. This implies, there is typically no simplified algorithm by which these attributes can be consolidated to identify which nation belongs the flag.

In contrast to previous scenario, to teach an algorithm how to recognize objects in images, we utilize a particular sort of Artificial Neural Network that is Convolutional Neural Network (CNN) which does not utilize raw data straightforwardly, rather gathers the essential element at first and after that procedure it. Convolutional Neural Network can be successfully used to classify unstructured image data. The convolution layer has an extremely significant impact to distinguish the critical features out of the image. Then the max-pooling layer saves the useful data while lessening the size. In our project we have to handle multiple classes. For Handling multiple classes we use Softmax function. Convolutional Neural Networks are enlivened by the brain. Research during the 1960s by D.H Hubel and T.N Wiesel on the brain of mammals recommended another model for how people see the world outwardly [26]. Convolutional Neural Networks have a different design in comparison to normal Neural Networks. Usually in Neural each layer is comprised of an arrangement of neurons, where each layer is completely associated with all neurons in the layer previously.

Before the usage of the proposed network a few contemplations must be taken including quality of the images taken and number of it. As without satisfying these two conditions it is impossible to achieve the desired accuracy.

1.2 Motivation

With the ongoing progress of the computational power, the frontiers of newest technologies are running relatively haywire. This computational power development is often described by Moore's Law. Alongside this open the door that Artificial Intelligence has turned out to be overwhelming and is being utilized in relatively every sector. So why not utilize this opportunity to take care of image detection issues which are extremely urgent in our current manageability. From the facial recognition in the Facebook to medical imaging in the health sector the image classification is becoming dormant day by day. Because of this present situation it is fundamental that the exploration in image detection must go as one.

Flags are national symbols. Every country has got a specific flag as their national symbol. There is plethora of benefits to detecting flag automatically. For travelers, it can be helpful to quickly recognize fellow countrymen, embassies and ships. Moreover, when attending some international conference, it can be essential to know which countries are participating in the conference by detecting their national flag. Similarly, if we talk about the team identification using their national flag in Fifa World Cup or International Cricket Tournament it’s going to be very easier using national flag detection [11]. For children it can be very helpful to gain knowledge about different nation's flag. So, these are reasons behind we are motivated in the topic of national flag detection.

1.3 Rationale of the study

Despite the fact that one can think there are lots of cluster of study and methods are available which can solve a lot of problems, the real consideration is, this will be enough for the future? As before it is mentioned that, the technology is getting advanced but with the exploring technology difficulties are also increasing. So now this time proper image recognition is an absolute necessity to guarantee smooth manner and security.

Presently the principle challenges are the formation of some common architectures which can solve the general problems. It is a tremendous task as real life scenario can be unique kind of and in the meantime paradoxical. For instance, images can be sometimes difficult to recognize still for human. So it completely makes sense to implement an efficient neural network architecture which can mimic the human brain.

Then again, the current image processing techniques must be optimized and adjusted with the goal that generate the outcomes we looked after. Specifically, the national flag detection requires lots of study to classify. All these things demands that the current research must go on to find the solutions of the newest problems and along with it to enhance the existing technologies. The methods presented here will also need further research as both artificial neural network and convolutional neural network are still in the initial stage. The hyper parameters tuning and an ideal design will never end up conceivable except if all of studies are finished.

1.4 Research Questions

- What kind of machine learning technique should be used for the solution of the problem?
- Which model we have to choose?
- How can we reduce overfitting problem?

- What are the challenges?

1.5 Expected Outputs

From the study in this paper, two efficient solutions from the existing technologies will be found. In terms national flag detection, the dataset will be made structured from the raw image data. A proper exploratory analysis can reveal the underneath structure among the data itself. Then with a successful implementation of the Artificial Neural Network through python codes and libraries will distinguish one specific countries flag among several countries flag. Detecting the national flag of various countries is becoming easier and as before it was quiet impossible to detect all national flags manually. It will be no more impossible because of the blessing of Artificial Neural Network. The efficiency of the classification will depend on the input dataset. If the dataset contains great amount of noise, it may make some mismatch [13]. Other than that the classification should be pretty impressive as the Artificial Neural Network is very efficient to classify the structured dataset. Though it might give us an excellent result, we should be always careful to cross check our results in case of any discrepancies among the dataset.

1.6 Report Layout

This report will initially begin with the hypothetical background of Artificial Neural Network and Convolutional Neural Network and it’s important to this research. Then a brief review of the current and relevant works will be discussed after that a thorough approach will be exhibited along with detailed description how we perform the entire procedure. At first national flag detection will be addressed. Consequently, the subsequent discoveries will be appeared in graph and a detailed discussions of the results will be presented. In the end, further improvement and potential proposals will be suggested. Along with, short summary and reference will be mentioned.

Chapter 2

Background

2.1 Theoretical Background

Machine learning has been picking up force over a decade ago: self-driving cars, effective web search, image recognition. The victories gradually propagate into our day to day life. Machine learning is a class of artificial intelligence methods, which allows the PC to work in a self-learning mode, without being explicitly programmed.

If you've spent any time reading about artificial intelligence, you'll almost certainly have heard about artificial neural networks. Artificial Neural Network (ANN) are computing systems vaguely inspired by the biological neural networks that constitute animal brains. The neural network itself is not an algorithm, but rather a framework for many different machine learning algorithms to work together and process complex data inputs.

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information [20]. The key element of this paradigm is the modern structure of the data processing system. It is composed of a large number of highly interconnected processing elements working in agreement to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. This idea emerged trying to recreate the procedures happening in the brain by Warren McCulloch and Walter Pitts in 1943[23]. Neural networks comprise of individual units called neurons. Neurons are situated in a progression of groups—layers. Neurons in each layer are associated with neurons of the following layer. Data comes from the input layer to the output layer along these compounds. Each individual node performs a simple mathematical calculation. Then it transmits its data to all the nodes it is connected to. Artificial neural networks are one of the main tools used in machine learning. As the "neural" part of their name suggests, they are brain-inspired systems which are intended to replicate the way that we humans learn. Neural networks consist of input and output layers, as well as a hidden layer consisting of units that transform the input into something that the output layer can use. They are excellent tools for finding patterns which are far too complex or numerous for a human programmer to extract and teach the machine to recognize.

To analyze the data from recent vast growth, the proper computation platform is needed. This is an important decision to make which network should we use. The networks will vary depending upon the choice we make. As we wanted to have flexibility in allowing the user to show multiple types of flags at one go, we have decided to go with image classification. Image classification will give us the localized bounding box of each type of flag. Moreover, to pick between the correct libraries to deal with the calculation can be abiding as well. Whatever the platform or libraries it will require a lot of calculation and pass different critical stages. How these computations can be done efficiently in terms of time and space is equally important. There are two choices, TransorFlow or Keras. Tensorflow is the most famous library used in production for deep learning models [15]. It has a very large and awesome community. However, TensorFlow is not that easy to use. On the other hand, Keras is a high level API built on TensorFlow (and can be used on top of Theano too). It is more user-friendly and easy to use as compared to TensorFlow [15]**.** Here in our project we use Keras library. On the other hand, image identification problem will be implemented through finding the proper identification of flag from a set of given dataset and many other issues. This problem is unique in the sense that this dataset is completely different. How to relate between the dependent and independent variables and at the same time with efficacy is the ultimate part of the solution. Indeed, even with the correct decision among these alternatives, the architecture of design will dependably require thought.

These problems require an effective neural network. The proper activation functions are essential part of designing a neural network. Although some pretty useful algorithms are already in existence, the accuracy and efficiency can always get better with the appropriate selection of neural network architecture [8].

Finally, from the discoveries of the outcomes a long term study ought to be led to choose between the calculation stage, parameters tuning and inherent mathematical strategies. Additionally, the association between the studies may bring some new thoughts which can be gainful for future.

The last influx of neural networks came regarding the expansion in computing. That brought Deep learning, where innovative structures of neural networks have turned out to be more intricate and ready to tackle an extensive variety of tasks that couldn't be adequately solved previously. Image classification is a prominent example. For an essential thought of how a deep learning neural network learns, imagine a factory line. After the data set as input, they are then passed down the conveyer belt, with each subsequent step or layer extracting an alternate arrangement of high-level features. If the network is intended to recognize an object, the primary layer may examine the brightness of its pixels.

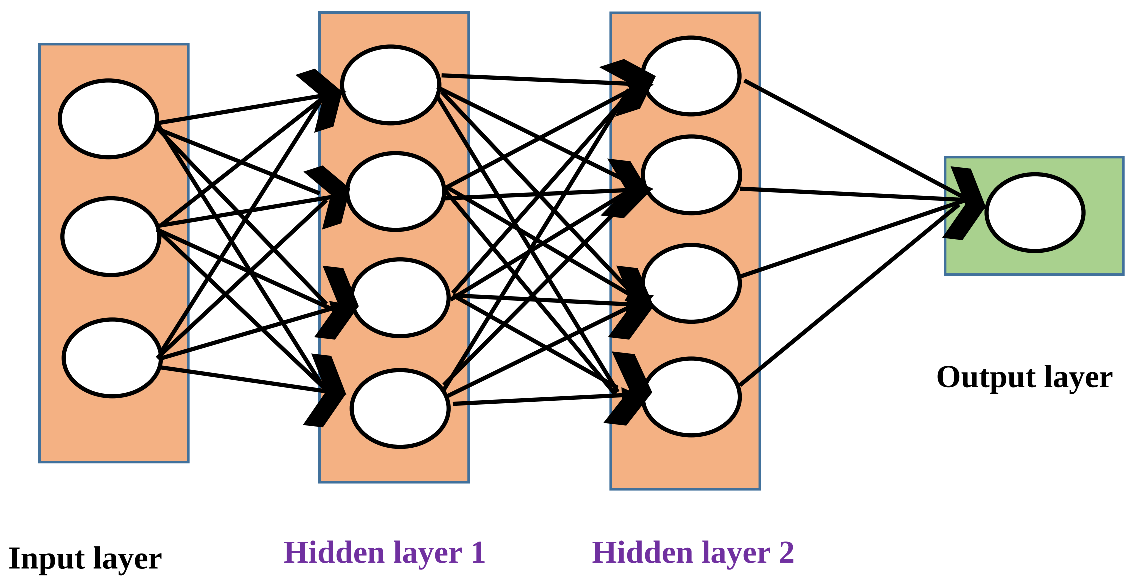

Figure 2.1: Basic Architecture of Artificial Network

There are multiple types of neural network, each of which come with their own specific use cases and levels of complexity. The most popular type of neural network is something called a feedforward neural network, recurrent neural network, convolutional neural networks and so on. However, in this research the focus is on convolutional neural networks.

In neural networks, Convolutional neural network is one of the primary type of neural networks to do images recognition, images classifications, objects detections, recognition faces and so on. These are the areas where CNNs are broadly used. CNN image classifications takes an input image, process it and characterize it under specific classes. Computers sees an input image as exhibit of pixels and it depends on the image resolution. In fact, deep learning CNN models to train and test, each input image will go through series of convolution layers along with filters, Pooling and apply ReLU and Softmax functions [12].

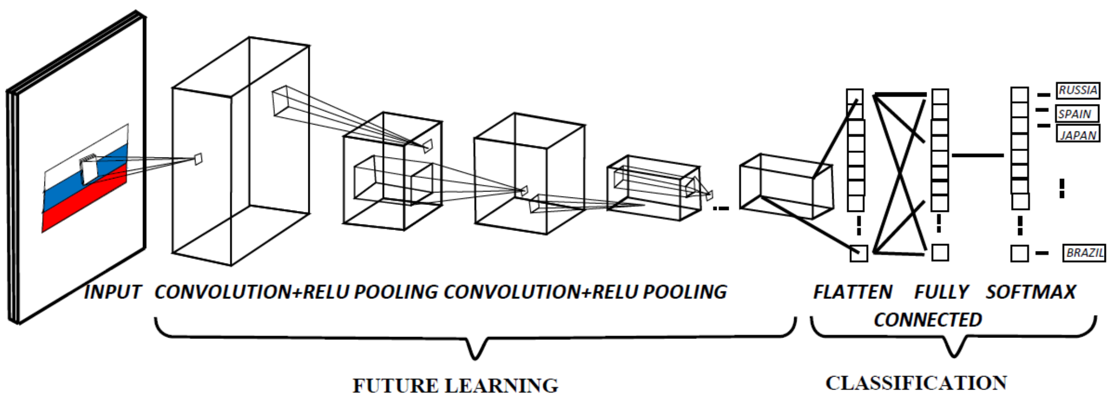

Figure 2.2: Convolution Layers in sequence

Convolution is the first layer to separate features from an input image. Convolution protects the connection between pixels by learning image features utilizing little squares of input data. It is a mathematical task that takes two inputs, for example, image matrix and a filter or kernel. These are convolution layers that will deal with our input images, which are seen as 2-dimensional matrices. We perform the convolution operation by sliding this filter over the input. At every location, we do element-wise matrix multiplication and sum the result. This sum goes into the feature map [12].

A pooling layer is another building block of a CNN [3]. Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network. Pooling layer operates on each feature map independently. The most common approach used in pooling is max pooling which just takes the max value in the pooling window.

After the convolution and pooling layers, our classification part consists of a few fully connected layers. However, these fully connected layers can only accept 1 Dimensional data. To convert our 3D data to 1D, we use the function flatten in Python. This essentially arranges our 3D volume into a 1D vector.

Activation is the activation function for the layer. This activation function has been proven to work well in neural networks. The ReLU function is the Rectified linear unit. It is the most widely used activation function [21]. It is defined as: f(x)=max (0, x). Another function that is softmax function would squeeze the outputs for each class between 0 and 1 and would also divide by the sum of the outputs. This essentially gives the probability of the input being in a particular class. For example, to classify an object with probabilistic values between 0 and 1. It can be defined as –

$$\sigma(Z)_j = \frac{e^{zj}}{\sum_{k=1}^{K} e^{zk}} \ for\ j = 1, \ldots., K.$$

Kernel size is the size of the filter matrix for our convolution [16]. So a kernel size of 3 means we will have a 3x3 filter matrix. In a CNN context, people sometimes use "kernel size" to mean the size of a convolutional filter, and likewise a "kernel" is the filter itself. Deep neural networks, more concretely convolutional neural networks, are basically a stack of layers which are defined by the action of a number of filters on the input. Those filters are usually called kernels. For example, the kernels in the convolutional layer, are the convolutional filters. The kernel size here refers to the width x height of the filter mask.

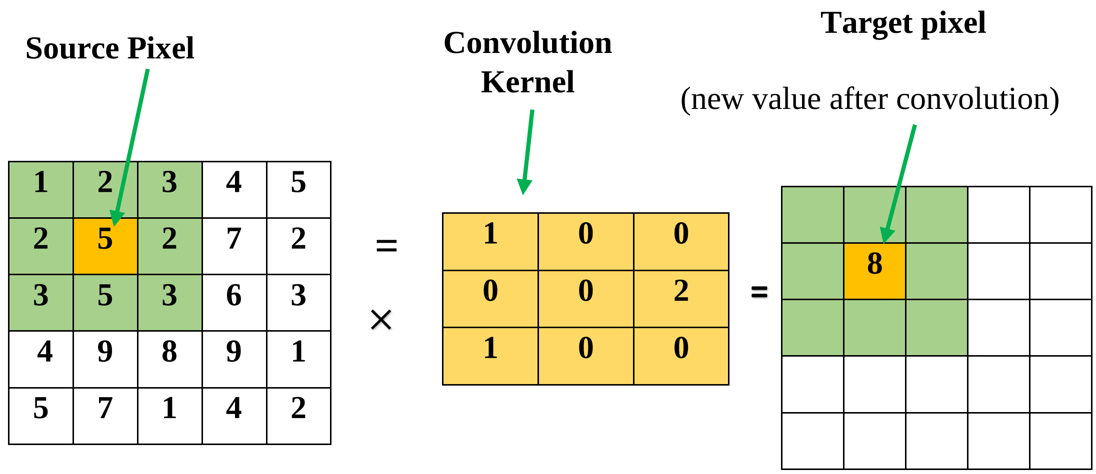

Figure 2.3: A kernel operates on one pixel.

2.2 Related Works

Artificial Intelligence Simple Perceptron was first created in 1957 at the Cornell Aeronautical Laboratory by Frank Rosenblatt [23]. Rosenblatt demonstrated how a binary classification problem can be automated with the use of simple perceptron. From that lots of research have been directed in the field of artificial intelligence. Neural Network turned out to be exceptionally well known in 1980's the point at which the famous godfather of neural network Geoffrey Hinton discovered a few achievements [24]. Alongside him some different researchers made some wonderful contributions but the idea while envisioned at first slowly died off in the 1990’s. As neural network requires expansive dataset to train the input data to maximize the outcome, the technology at that did not allow the huge scale usage of neural network. But as the recent ongoing development in the computational power, the idea of neural network has been revived and has made distinction in a very short span of time.

However, the latest research is on mostly combining multiple neural network framework to create a hybrid platform that can perform complex tasks which otherwise seems impossible. Convolutional neural network is one of the major deep learning architectures that is successfully used in many different applications and continuously evolving field of study.

There is enormous image classification research done by Convolutional neural network. As example, Nationality identification, Digit identification, Cat and dog identification, Vehicle recognition and also National Flag detection. But national flag detection related researches are so rare. One or two researches are worked with National Flag detection but by those research, satisfactory accuracy is not obtained and the researchers are work with very less number of data. In 2017 Hoang Huu Duc and Keechul Jung work on this using Tensorflow [1]. In this paper the authors work with limited data and they find the accuracy 93%. And also work using Keras 2018 Fifa World cup round of 16 flag detection using CNN [11]. Our goal is to work with huge data and attain an optimal accuracy and also increasing accuracy.

2.3 Research Summary

The principle reason for this research is to implement two kinds of neural network which is Artificial Neural Network and Convolutional Neural Network and keeping in mind that doing as such effectively classify image data for the most part. After the effective implementation, a few difficulties and challenges for development are found. To address these issues while the research is being done and furthermore keeping those for the future works is eventual.

2.4 Scope of the Problem

Both of our issues are exceptionally shortsighted in nature, at least considering how much complexity a neural network can contain. So the fundamental problem is collecting proper data, proper data means noise free images along with clear attributes of the image.

Secondly, the target is to classify structured image data in Artificial Neural Network and unstructured image data information in Convolutional Neural network. After that investigating the outcomes from the graphs and accuracy are also considered with same importance. Furthermore, what changes and traps can be used to make the networks more productive and powerful and moreover getting expected outcomes after analyzing those aspects are tended to. However, developing new algorithms or mathematical equations are not in the in the scope of this research.

2.5 Challenges

The main challenges associated with the research are given below-

- Getting good quality datasets with needed attributes.
- Preprocessing of the datasets for the effective analysis.
- Corresponding exploratory data analysis to visualize the nature of the datasets.
- Choosing a proper programming platform and corresponding libraries.
- Incorporating GPU computing along with CPU computing to reduce time complexity.
- Choosing proper initial hyper parameters and tweaking them while programming.
- Finding the optimal number of convolutional layers in CNN implementation.
- Getting enough data to train the neural network properly.
- To choose the correct activation functions in both hidden and output layers.
- To avoid over fitting in Convolutional Neural Network.

CHAPTER 3
Research Methodology

3.1 Introduction

This chapter is about the process in which the work of National Flag Detection using CNN was done. The chapter also include the techniques and reason of using these techniques. This chapter also describe that how well our research model and why some other model are not used in the national flag detection. This chapter includes from data collection to model train using CNN. We provide a simple introduction on dataset, we give about the data preprocessing then we discuss the model installation and introduce about the train model.

3.2 Research subject and instrumentation

Our research subject is on Machine learning on the topic of National Flag Detection using CNN. CNN stands for convolution neural network which is discussed in the previous chapter (chapter 2). This work also following some basic steps and some techniques with the basic steps. We make a flowchart of out topic so that we can briefly understand how our project work from data collection to result evaluation. A flowchart is a type of diagram that represents a working process or workflow.

The flowchart shows the steps of boxes and their steps by connecting the boxes with arrows. Flowcharts [Figure 3.1] are used in analyzing, designing or managing a process. The following diagrammatic representation illustrates a solution model to our system.

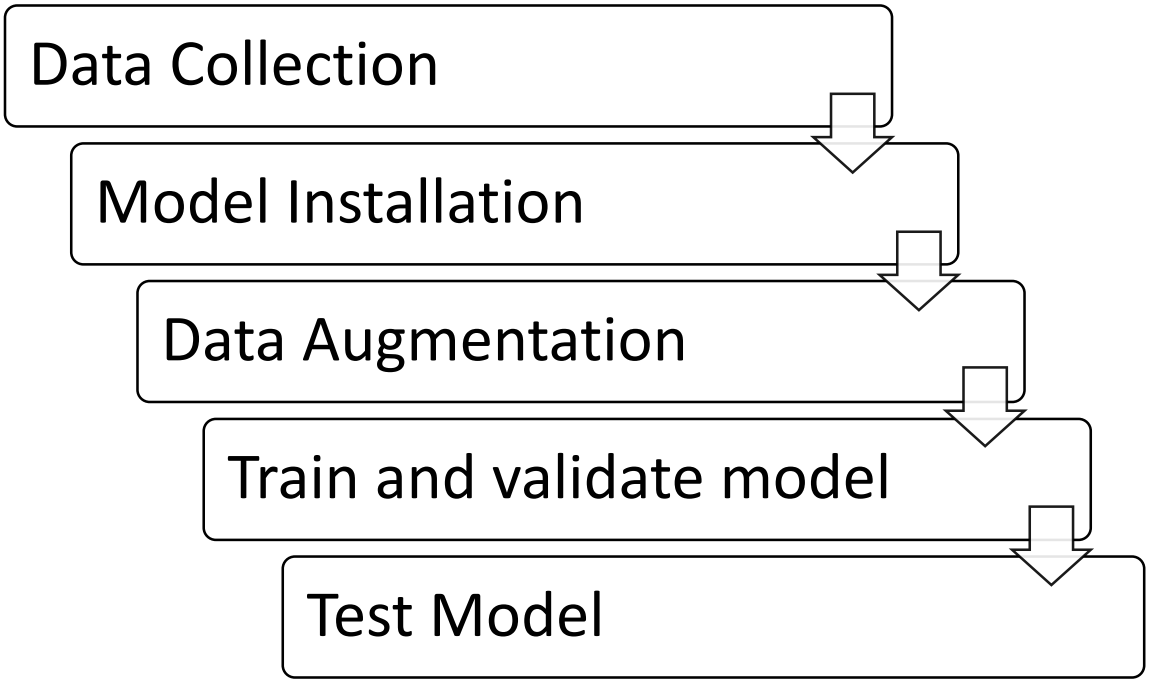

Figure 3.1: Flow chart of the system model

3.3 Data collection procedure

3.3.1 Data collection

There are lots of countries in the world and also lots of distinct flag. There are similarities of some few countries national flag. For detecting national flag, we have collected 5000 images of national flag of ten different countries for our experiment. They are Argentina, Australia, Brazil, China, Germany, Japan, Mexico, Russia, Saudi Aribia and Spain. We have collected different kinds of flag like straight flag, waiving flag, zoomed flag, shaded flag and lots of different angle flag. We have collected the national flag image from search engine (Google), Facebook and some of few images we made using Photoshop using different texture, contrast and different shade. All the images are the JPG format. The challenges of collecting image from online is that images are not properly found because some of the images have some written some text in front of the flag. So we have to negate those images because that can be create massive problem when we train in the model that create the error for test evaluation and accuracy will be less.

3.3.2 Process the directory

After collecting all the 10 country images now we have to divide into 80%-20% rule of the images into train and test part because we have to validate the images to the corresponding countries in the train model so that we can find the accuracy level of the flag of distinct countries. So, we take 20% of data of the different countries randomly in country wise folder. And we use the number code for each image so that we train the images the Neural Network understand which country is stand for. The image codes are:

Table 3.1: Country names and their identification codes

Country	Codes
Argentina	0
Australia	1
Brazil	2
China	3
Germany	4
Japan	5
Mexico	6
Russia	7
Saudi Aribia	8
Spain	9

Figure 3.2 National flag with the code names

3.4 Implementation Requirements

3.4.1 Keras

Keras is an API intended for individuals, not machines. Keras pursues best practices for diminishing intellectual load [4]: it offers steady and basic APIs, it limits the quantity of client activities required for normal utilize cases, and it gives clear and noteworthy input upon client mistake. This makes Keras simple to learn and simple to utilize. As a Keras client, you are more gainful, enabling you to attempt a larger number of thoughts than your opposition, quicker - which thusly causes you win machine learning rivalries. This convenience does not come at the expense of diminished adaptability: since Keras incorporates with lower-level profound learning dialects (specifically TensorFlow), it empowers to execute anything you could have worked in the base dialect. Specifically, as tf.keras, the Keras API incorporates flawlessly with your TensorFlow work processes[19].

3.4.2 Model installation

This experiment is based on the sequential API model of Keras platform. The processor is 2.70GHz Intel core i5, memory 4GB 1600MHz DDR3 RAM, System type: 64-bit Linux operating system and x-64 based processor.

First of all, we have to import Keras. Then import sequential API, where we have just to add one layer at a time, beginning from the input.

The CNN can confine include that are extremely valuable in wherever from these changed pictures that is highlight maps. The mix of convolutional and pooling layers, CNN can join neighborhood includes and can take in more worldwide highlights of the picture. Dropout is a regularization technique, where an extent of hubs in the layer are arbitrarily disregarded (setting their weights to zero) for each preparation test. This drops arbitrarily a proportion of the system and powers the system to learn includes distributed. This system likewise enhances speculation and lessens the overfitting.

The Flatten layer is use to change over the last element maps into a one single 1D vector. This straightening step is required with the goal that you can make utilization of completely associated layers after some convolutional/maxpool layers. It consolidates all the discovered nearby highlights of the past convolutional layers.

The activation function use for layer. The activation function we are using for first 2 layers is the ReLU or Rectified Linear Activation [17]. The ReLU function is the nonlinear function, which means it can easily back propagate the errors and have multiple layer of neurons are activated. The main advantage of the ReLU function that it does not activate all the neuros at a time. Because the function if the input is negative it will changes to zero and that’s why the neurons are not get activated only few neurons are activated. There are other nonlinear function like tanH or sigmoid but rest of them ReLU function is best all of them because the function’s complexity is less than other and work very precisely.

At last we utilized the highlights in two completely associated (Dense) layers which is simply fake a neural systems (ANN) classifier. In the last layer (Dense (10,activation="softmax")) the net yields dissemination of likelihood of each class. Softmax function is basically used for classification the given images [18].

3.4.3 Hyper parameters

When we using CNN, we have decided the three hyper parameters they are:

I. Kernel size
II. Stride
III. padding

kernel size use for musk in individual picture on every pixel. Here we use 3X3 kernel size. Stride in the experiment means the step of convolution operation. Or stride is the number of pixels go over or shift over the input image/ matrix if the stride is 1 then the filter moves to 1 pixel at a time. For example, if we de valid convolution in two sequence of length 20 and 16, in general we get an output which length is 5(20 - 16 + 1). That’s means the sequence 2 moves step by step besides sequence 1. Here we use strides=(2, 2). Sometimes filter are not fit as good as in the input matrix. So we can have two choice. That is the padding the picture with a zero padding so that it fits and another is drop the portion of image where the filter does not fit. That is called valid padding which keeps only the valid part of the input images/matrix.

```
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 32, 32, 64)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 32, 32, 512)       295424
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 32, 32, 512)       2359808
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 16, 16, 512)       0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 16, 16, 256)       1179904
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 16, 16, 256)       590080
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 16, 16, 256)       590080
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 8, 8, 256)         0
_________________________________________________________________
conv2d_8 (Conv2D)            (None, 8, 8, 128)         295040
_________________________________________________________________
conv2d_9 (Conv2D)            (None, 8, 8, 128)         147584
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 4, 4, 128)         0
_________________________________________________________________
flatten_1 (Flatten)          (None, 2048)              0
_________________________________________________________________
dense_1 (Dense)              (None, 100)               204900
_________________________________________________________________
dropout_1 (Dropout)          (None, 100)               0
_________________________________________________________________
dense_2 (Dense)              (None, 10)                1010
=================================================================
Total params: 5,702,550
Trainable params: 5,702,550
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.3: Model summary of out Model Which is the number of parameters of the whole model

3.4.4 Optimizer

Optimizer is basically use for minimize the cost function to approach the minimum point using Gradient Descent Algorithms. Once our layer are added to the model, we need to set up a score function. We characterize the misfortune capacity to quantify how inadequately our model performs on pictures with known names. It is the loss rate between the observed names and the predicted names. We utilize a particular shape for categorical classifications called the "categorical_crossentropy".

This optimizer function will iteratively enhance parameters (filter kernel values, weights and predisposition of neurons) with the end goal to limit the error/loss. we choose Adam optimizer which is the best Gradient Descent optimizer, this is utilized to perform enhancement and is extraordinary compared to other streamlining agent at present.

Metric function "accuracy" is utilized is to assess the execution our model. This metric function is like the loss function, with the exception of that the outcomes from the metric assessment are not utilized when preparing the model.

3.4.5 Annealer

With the end goal to make the optimizer quicker and nearest to the global minimum of the loss function, we utilized a annealing technique for the learning rate (LR). The LR is the progression by which the optimizer strolls through the loss function. The higher LR, the greater are the means and the speedier is the convergence. Anyway, the sampling is extremely poor with a high LR and the optimizer could likely fall into a local minima.

It's better to have a diminishing learning rate amid the preparation to reach productively the global minimum of the loss function. To keep the upside of the quick calculation time with a high LR, we diminished the LR significantly every X epochs depending on need if accuracy is not increasing.

With the ReduceLROnPlateau work from Keras.callbacks, we diminish the LR by half significantly if the exactness isn't enhanced after 3 epochs.

3.4.6 Data Augmentation

With the end goal to abstain from overfitting issue, we have to grow Artificially our National Flag dataset. We can make your current dataset considerably bigger [19].

Methodologies that adjust the training dataset in manners that change the cluster array representation while keeping the name the equivalent are known as data augmentation systems. We use Some famous augmentation utilize are Rotation, Horizontal flips, Width shift, Height Shift, Rescale, Shear and Zoom range.

By applying this augmentation technique to our training dataset, we can without much of a stretch Double or Triple the quantity of training examples and make an exceptionally robust model.

3.4.7 Train model

After finishing all this and end of the data augmentation we train our model for 50 Epochs. We use 4283 real training images by 142 channels have 30 images per channel with reference of Validation Data or test data of 1049 images. Using also callback function of Learning rate reduction if accuracy not increase every 3 steps/epochs. We will discuss our result and accuracy in the next chapter (Chapter 5).

Chapter 4

Experimental Results and Discussion

4.1 Introduction

In this chapter we discuss our experimental results and discussion about our research project on National Flag Detection using CNN. We trained our model and all the implementation process are described in previous chapter (Chapter 3). In this part we will show our training accuracy with respect to validation data or test images and also see some statistical graph of training vs validation accuracy and loss.

4.2 Experimental Results

After trained our model we find our accuracy 98% for our experiment. Which is the highest accuracy of previous work on National Flag Detection using CNN till now. Our model reaches approximately 98% accuracy on validation dataset after 50 epochs.

```
Epoch 1/50
143/142 [==============================] - 879s 6s/step - loss: 2.3048 - acc: 0.1193 - val_loss: 2.2995 - val_acc: 0.1268
Epoch 2/50
143/142 [==============================] - 839s 6s/step - loss: 2.1362 - acc: 0.1867 - val_loss: 1.5417 - val_acc: 0.3695
Epoch 3/50
143/142 [==============================] - 840s 6s/step - loss: 1.4297 - acc: 0.4363 - val_loss: 1.0274 - val_acc: 0.5772
Epoch 4/50
143/142 [==============================] - 833s 6s/step - loss: 1.1225 - acc: 0.5690 - val_loss: 0.7746 - val_acc: 0.6725
Epoch 5/50
143/142 [==============================] - 836s 6s/step - loss: 0.8629 - acc: 0.7135 - val_loss: 0.5625 - val_acc: 0.7993
Epoch 6/50
143/142 [==============================] - 840s 6s/step - loss: 0.6761 - acc: 0.7669 - val_loss: 0.5744 - val_acc: 0.8305
Epoch 7/50
143/142 [==============================] - 840s 6s/step - loss: 0.5425 - acc: 0.8084 - val_loss: 0.4893 - val_acc: 0.8303
Epoch 8/50
143/142 [==============================] - 848s 6s/step - loss: 0.4673 - acc: 0.8340 - val_loss: 0.3731 - val_acc: 0.8468
```

Figure 4.1: Training accuracy after 8 Epoch

```
Epoch 00018: ReduceLROnPlateau reducing learning rate to 0.0005000000237487257.
Epoch 19/50
143/142 [==============================] - 837s 6s/step - loss: 0.1427 - acc: 0.9601 - val_loss: 0.1463 - val_acc: 0.9641
Epoch 20/50
143/142 [==============================] - 838s 6s/step - loss: 0.1263 - acc: 0.9618 - val_loss: 0.1180 - val_acc: 0.9715
Epoch 21/50
143/142 [==============================] - 834s 6s/step - loss: 0.1147 - acc: 0.9690 - val_loss: 0.1137 - val_acc: 0.9697
Epoch 22/50
143/142 [==============================] - 842s 6s/step - loss: 0.1250 - acc: 0.9678 - val_loss: 0.1104 - val_acc: 0.9736
Epoch 23/50
143/142 [==============================] - 838s 6s/step - loss: 0.1131 - acc: 0.9706 - val_loss: 0.1312 - val_acc: 0.9661
Epoch 24/50
143/142 [==============================] - 847s 6s/step - loss: 0.1037 - acc: 0.9723 - val_loss: 0.1237 - val_acc: 0.9710
Epoch 25/50
143/142 [==============================] - 838s 6s/step - loss: 0.1041 - acc: 0.9713 - val_loss: 0.0907 - val_acc: 0.9769
Epoch 26/50
143/142 [==============================] - 836s 6s/step - loss: 0.1141 - acc: 0.9711 - val_loss: 0.0912 - val_acc: 0.9772
```

Figure 4.2: Training accuracy after 26 Epoch

```
Epoch 44/50
143/142 [==============================] - 838s 6s/step - loss: 0.0393 - acc: 0.9883 - val_loss: 0.0829 - val_acc: 0.9878
Epoch 45/50
143/142 [==============================] - 839s 6s/step - loss: 0.0369 - acc: 0.9895 - val_loss: 0.0898 - val_acc: 0.9876
Epoch 46/50
143/142 [==============================] - 832s 6s/step - loss: 0.0411 - acc: 0.9886 - val_loss: 0.1081 - val_acc: 0.9810
Epoch 47/50
143/142 [==============================] - 833s 6s/step - loss: 0.0343 - acc: 0.9888 - val_loss: 0.0959 - val_acc: 0.9800

Epoch 00047: ReduceLROnPlateau reducing learning rate to 6.25000029685907e-05.
Epoch 48/50
143/142 [==============================] - 845s 6s/step - loss: 0.0283 - acc: 0.9914 - val_loss: 0.0941 - val_acc: 0.9810
Epoch 49/50
143/142 [==============================] - 896s 6s/step - loss: 0.0380 - acc: 0.9888 - val_loss: 0.1120 - val_acc: 0.9869
Epoch 50/50
143/142 [==============================] - 919s 6s/step - loss: 0.0327 - acc: 0.9909 - val_loss: 0.0620 - val_acc: 0.9851

Epoch 00050: ReduceLROnPlateau reducing learning rate to 3.125000148429535e-05.
```

Figure 4.3: Training accuracy after 50 Epoch

4.3 Result analysis

4.3.1 Statistical Analysis

Figure 4.4 and Figure 4.5 show the variation in accuracy and loss rate based on our training dataset. From the generating figure the difference between validation and training accuracy approximately same in every time while the training out model. That means that our model is not over fit the training dataset. So our model is very good well trained. In the figure 4.4 the red line represents validation loss and blue line Training loss. In figure 4.5 the red line represents the validation accuracy and blue line represents the training accuracy

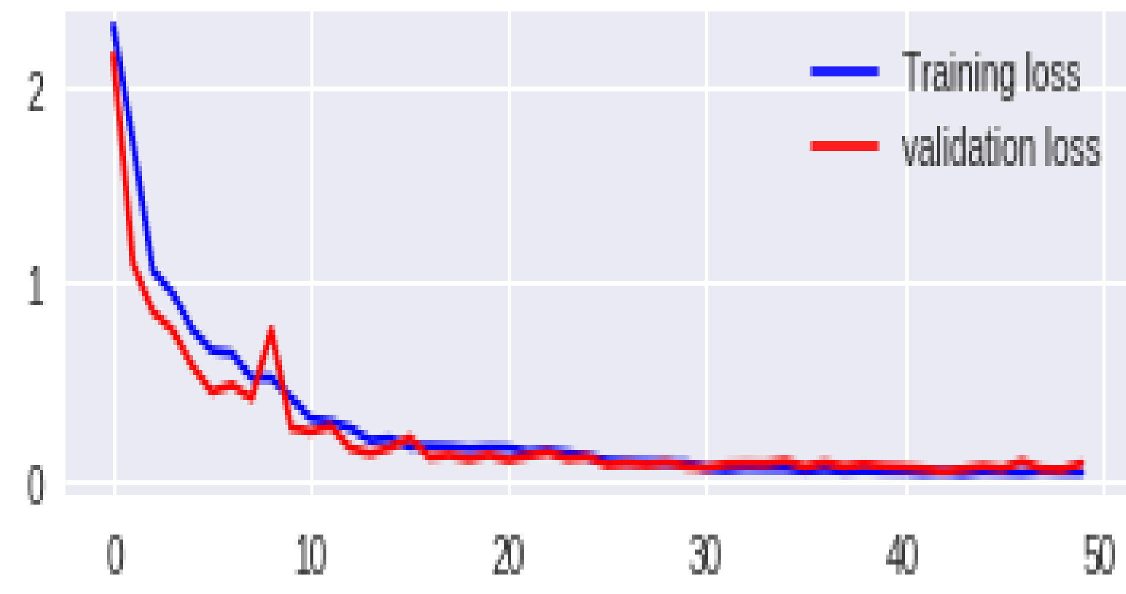

Figure 4.4 Training loss and validation loss on every steps of 50 epochs

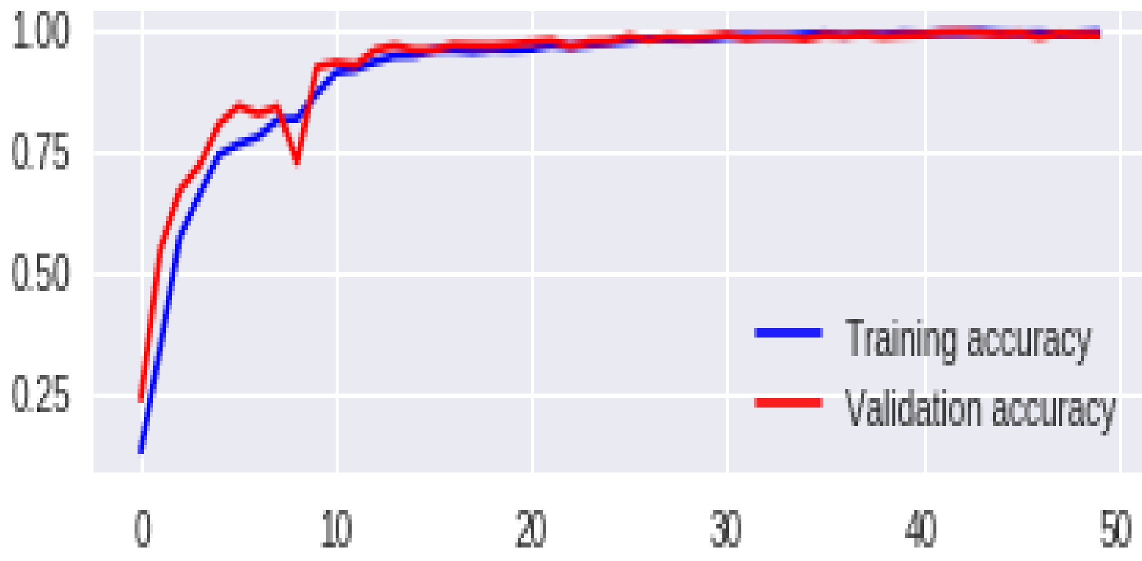

Figure 4.5 Training Accuracy and Validation Accuracy on every steps on 50 epochs

Table 4.1: Table of the Description of the two figures

Datasets	**Index**	**Performance**
Datasets	The accuracy of the training set	98%
	The accuracy of the validation set	98%
	The loss of the training set	0.033
	The loss of the validation set	0.062

4.3.2 Test the New Image

After all this process we give a new image in the test directory or any image path in computer the experiment shows us the correct country in a list/array. In the array it shows us 1 in the correct country position and the rest of the position is 0. For example, we take an image of Brazil and it shows us 1 in Brazil Position and rest of is all Zeros 0.

```
[[0. 0. 1. 0. 0. 0. 0. 0. 0. 0.]]
{'Argentina': 0, 'Australia': 1, 'Brazil': 2, 'China': 3, 'Germany': 4, 'Japan': 5, 'Mexico': 6, 'Russia': 7, 'Saudi Arabia': 8, 'Spain': 9}
```

Figure 4.6 Test a new image

4.3.3 Confusion Matrix

Confusion matrix basically gives us an idea about how well our classifier has performed, with respect to performance on individual classes. So typically a confusion matrix is filled up based on the test set whose true labels is known. The test data is passed through the classifier and predictions are noted. A table of predicted labels vs true labels is then filled out [27].

The way you fill out the confusion matrix is simple, if the predicted label matches its true label then count one on the diagonal element of the corresponding class. If it is wrongly classified count one on the (predicted class, target class) element. This process is repeated for all elements of the set under consideration.

```
Confusion Matrix
[[93  0  0   0  0  0   1  1  0   0]
 [ 0 96  0   1  0  0   0  0  0   1]
 [ 0  0 99   1  1  0   0  0  0   1]
 [ 0  2  0 129  0  1   0  0  0   0]
 [ 0  1  1   0 93  0   0  0  0   0]
 [ 0  0  0   2  0 86   0  0  0   0]
 [ 0  0  0   1  0  1 124  0  1   1]
 [ 2  0  0   0  0  0   0 94  0   0]
 [ 0  0  0   0  0  0   2  0 86   0]
 [0   0  0   2  0  0   1  0  0 101]]
```

Figure 4.7: Confusion Matrix of Test datasets

TP= True positive, number of examples predicted positive that are actually positive.

Classified by the model as Will return and had in fact Returned in reality.

FP = False Positive, number of examples predicted positive, that are actually negative.

Classified by the model as Will return but actually Didn't return.

FN = False Negative, number of examples predicted negative that are actually positive.

Classified by the model as Won't return but had actually Returned in reality

TN = True Negative, Number of examples are predicted negative that are actually negative.
Classified by the model as Won't return and in fact Didn't return in reality.

True Positive Rate = $\frac{TP}{TP+FN}$

False Positive Rate = $\frac{FP}{FP+FN}$

False Negative Rate = $\frac{FN}{TP+FN}$

Tree Positive Rate = $\frac{TN}{TN+FP}$

Recall: recall is the number of true positives divided by the number of true positives plus the number of false negatives.

True positives are data point classified as positive by the model that actually are positive (meaning they are correct), and false negatives are data points the model identifies as negative that actually are positive (incorrect).

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

Precision = Precision is calculated as the number of correct positive predictions divided by the total number of positive predictions. It is also called positive predictive value. False positives are cases the model incorrectly labels as positive that are actually negative. While recall expresses the ability to find all relevant instances in a dataset, precision expresses the proportion of the data points our model says was relevant actually were relevant.

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

F1-score: The F_1 score (also F-score or F-measure) is a measure of a test's accuracy. This is a weighted average of the true positive rate (recall) and precision.

F1-score = $\frac{2*preision*recall}{precision+recall}$

Now, Calculating Precision, Recall and F1-Score for the flag of Argentina. In Argentina the TP is 93 and FP is 2 from the confusion matrix of the support of 95.

So the Precision = $\frac{TP}{TP+FP}$ = $\frac{93}{93+2}$ = 0.978 ≈ 0.98

For Recall = $\frac{TP}{TP+FN}$ = $\frac{93}{93+2}$ = 0.978 ≈ 0.98

F1-score = $\frac{2*preision*recall}{precision+recall}$ = $\frac{2*0.98*0.98}{0.98+0.98}$ = 0.98

By this way we can calculating other Flags Accuracy with the help of confusion matrix and the support. So Accuracy of other countries Australia 0.97, Brazil 0.98, China 0.98, Germany 0.97, Japan 0.98, Mexico 0.98, Russia 0.98, Saudi Arabia 0.98 and Spain 0.98. So our macro Average is 0.98. And the final Accuracy is 98%. we use the transfer learning technology to identify the national flag of ten countries based on our dataset. And we get the accuracy of the model is 98%.

4.4 Summary

After all the experiment based on CNN in Keras platform, we use the transfer learning technique to identify the nation flag of 10 countries based on our dataset. And we get the accuracy of the model is 98%.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

National flag images are very useful images. By the image of national flag, we can know any country's name. In this research we determined national flag's name among test information or other any image. Our work which is National-Flag-Detection, is belongs to the field of image processing. We have used convolutional neural network. We used ten countries national flag's images. For ten countries we collected 5000 images. We have to divide into 80%-20% rule of the images into train and test part. We take 20% of data of the different countries randomly in country wise folder. By using training an artificial neural network on few thousand images of national flags and make the NN (Neural Network) learn to predict which class the image belongs to.

In the beginning of the work we studied some works already done in this field and we found that their quantity of dataset is very low with low accuracy. So we increased our datasets and build our model so that the accuracy is increased. And our model is well trained. After trained our model we are able to obtained a satisfactory accuracy that is 98%. Which is the highest accuracy of previous work on National Flag Detection using CNN till now. Our model reaches approximately 98% accuracy on validation dataset after 50 epochs.

5.2 Future Scope

As our work is on National-Flag-Detection, which is a vast field of research. Lots of improvement and further developments can be performed on our method.

Now this study used 10 countries. In future we will work with almost all nations of the world. Besides we develop application. By this application we can use camera to take any national flag image and show the country name as an output instantly. Besides it shows countries details information from Wikipedia.

In future, we can easily plan a specific goal. From this research next time can be proposed a new enhancement and national flag detection techniques which will be better than present existing enhancement techniques. For instance, we are interested to work with R-CNN, Fast R-CNN or Faster R-CNN in future which will make National Flag Detection more efficient and faster.

REFERENCES

[1] *Hoang Huu Duc, Keechul Jung "Applying Tensorflow with Convolutional Neural Networks to Train Data and Recognize National Flags", Conference, 448, 6, 14 May 2017.*

[2] *Girshick R, Donahue J, Darrell T, Malik J. Region-based convolutional networks for accurate object detection and semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 2015*

[3] *Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)*

[4] *Sutskever, I., Nair, V.: Mimicking go experts with convolutional neural networks. In: Kůrková, V., Neruda, R., Koutník, J. (eds.) Artificial Neural Networks-ICANN 2008, pp. 101–110. Springer, Heidelberg (2008)*

[5] *K. He, X. Zhang, S. Ren, J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition", ECCV, (2014)*

[6] *Shen W, Zhou M, Yang F, Yang C, Tian J. Multi-scale convolutional neural networks for lung nodule classification. IPMI.,588–599 (2015)*

[7] *Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, Le-Cun Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. ICLR. 2014*

[8] Understanding of Convolutional Neural Network (CNN) — Deep Learning
In neural networks, Convol, available at <<https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning>>, last accessed on 05-05-2018 at 10.00pm.

[9]Convolutional Neural Networks, available at <<https://www.coursera.org/learn/convolutional-neural-networks>>, last accessed on 10-06-2018 at 12.00am.

[10] Beginner's guide to CNNs with keras {99.8%}, available at <<https://www.kaggle.com/moghazy/beginner-s-guide-to-cnns-with-keras-99-8>>, last accessed on 13-06-2018 at 09.00pm.

[11] Fifa world cup 2018 Round of 16 flag detection using CNN, available at <<https://www.python36.com/fifa-worldcup-flag-detection-using-cnn/>>, last accessed on 17-08-2018 at 7.00pm.

[12] A Beginner's Guide to Convolutional Neural Networks (CNN), available at <<https://skymind.ai/wiki/convolutional-network?fbclid=IwAR3Foi2yMBhy03EkP>>, last accessed on 27-08-2018 at 08.00pm.

[13] An intuitive guide to Convolutional Neural Networks, available at <<https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050?fbclid=IwAR3ia4ONQgGmw8Re1o>>, last accessed on 03-08-2018 at 11.00pm.

[14] Image Classification with Convolutional Neural Networks, available at <<https://medium.com/@ksusorokina/image-classification-with-convolutional-neural-networks-496815db12a8>>, last accessed on 16-08-2018 at 09.30pm.

[15] Create your first Image Recognition Classifier using CNN, Keras and Tensorflow backend, available at << https://medium.com/nybles/create-your-first-image-recognition-classifier-using-cnn-keras-and-tensorflow-backend-6eaab98d14dd?fbclid=IwAR1N7eFzo2OAfc>>, last accessed on 16-08-2018 at 10.00pm.

[16] Image Scaling using Deep Convolutional Neural Networks, available at <<https://engineering.flipboard.com/2015/05/scaling-convnets?fbclid=IwAR1L03UHc00gR0FfrjM7ra>>, last accessed on 18-08-2018 at 08.00pm.

[17] Activation Functions: Neural Networks, available at <<https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6?fbclid=IwAR2vqyoGYBZmsgqFZE3t>>, last accessed on 18-08-2018 at 08.30pm.

[18] Difference between soft-max function and sigmoid function, available at <<http://dataaspirant.com/2017/03/07/difference-between-softmax-function-and-sigmoid-function/?fbclid=IwAR0Q7muTefX>>, last accessed on 18-08-2018 at 09.30pm.

[19] Data Augmentation | How to use Deep Learning when you have Limited Data—Part 2, available at <<https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced?fbclid=IwAR22NOf2iL1jTEN0FjpI>>, last accessed on 25-08-2018 at 09.30pm.

[20] Artificial neural network, available at <<https://en.wikipedia.org/wiki/Artificial_neural_network>>, last accessed on 27-08-2018 at 12.00am.

[21] Understanding Activation Functions in Neural Networks, available at <<https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>>, last accessed on 18-08-2018 at 08.00pm.

[22] Kernel Size, available at <<https://en.wikipedia.org/wiki/Kernel_image_processing>>, last accessed on 27-08-2018 at 07.30pm.

[23] Perceptions, available at <<https://www.britannica.com/technology/perceptrons>>, last accessed on 27-08-2018 at 12000pm.

[24] Geoffrey Hinton | Achievements, available at<<https://en.wikipedia.org/wiki/Geoffrey_Hinton>>, last accessed on 29-08-2018 at 06.30pm.

[25] Keras Documentation, available at <<https://keras.io/>>, last accessed on 09-10-2018 at 12.00pm.

[26] Recounting the impact of Hubel and Wiesel, available at <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2718241/>>, last accessed on 10-09-2018 at 10.00pm.

[27] Simple guide to confusion matrix terminology, available at <<https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/>>, last accessed on 16-09-2018 at 8.00pm.