HAND GESTURE RECOGNITION USING DEEP LEARNING TECHNIQUES

A PROJECT REPORT

submitted by

BABY RUKHIYA
(Reg. No. TKM20EE1107)

to

the APJ Abdul Kalam Technological University
in partial fulfillment of the requirements for the award of the Degree

of

Master of Technology
in
Electrical and Electronics Engineering

with specialisation in

Industrial Instrumentation and Control

Department of Electrical and Electronics Engineering
TKM College of Engineering
Kollam - 691005
KERALA

JULY 2022

DECLARATION

I undersigned hereby declare that the project report entitled "**Hand Gesture Recognition Using Deep Learning Techniques**", submitted for partial fulfillment of the requirements for the award of degree of Master of Technology in Electrical and Electronics Engineering with specialisation in Industrial Instrumentation and Control, of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of *Prof. Sumayya Jaleel*, Asst. Professor, Department of Electrical and Electronics Engineering. This submission represents my ideas in my own words and where ideas or words of others have been included. I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Kollam
July 01, 2022

BABY RUKHIYA

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

TKM COLLEGE OF ENGINEERING

KOLLAM - 691005

CERTIFICATE

This is to certify that the report entitled " **Hand Gesture Recognition Using Deep Learning Techniques** " submitted by **BABY RUKHIYA** , **(Reg. No. TKM20EEII07)** of fourth semester to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electrical and Electronics Engineering with specialisation in Industrial Instrumentation and Control, is a bonafide record of the project work done by her under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Prof. Sumayya Jaleel
Project Supervisor
Assistant Professor
Department of EEE
TKM College of Engineering

Prof. Sumayya Jaleel
Project Coordinator
Assistant Professor
Department of EEE
TKM College of Engineering

Prof. Shanavas T N
PG Coordinator
Associate Professor
Department of EEE
TKM College of Engineering

Dr. Sabeena Beevi K
Head of the Department
Associate Professor
Department of EEE
TKM College of Engineering

Acknowledgement

A lot of effort and hard work has been put into this project in course of its presentation. However, it would not have been possible without the kind support and help of many individuals and other sources. I would like to extend my sincere thanks to all of them. I take this opportunity to express my deep sense of gratitude and sincere thanks to all who helped me to complete this project report successfully.

I express my sincere thanks to *Dr. T A Shahul Hameed*, Principal, TKM College of Engineering for his encouragement in the completion of my project.

I thank *Dr. Sabeena Beevi K*, Head of the Department, Department of Electrical and Electronics Engineering, *Dr. Imthias Ahamad T P*, Professor, Department of Electrical and Electronics Engineering and *Prof. Shanavas T N*, PG Coordinator, Department of Electrical and Electronics Engineering for their support and cooperation.

I am greatly thankful to my Project Supervisor *Prof. Sumayya Jaleel* , Assistant Professor, Department of Electrical and Electronics Engineering for her supervision, assistance and helpful suggestions.

I am deeply indebted to *Prof. Sumayya Jaleel*, Assistant Professor, Project Coordinator, Department of Electrical and Electronics Engineering and *Prof. Amal A*, Assistant Professor, Department of Electrical and Electronics Engineering, for their excellent guidance, positive criticism and valuable comments.

Finally I thank my parents, friends, near and dear ones who directly and indirectly contributed to the successful completion of my project.

BABY RUKHIYA

Abstract

Deep learning algorithms have attracted more attention recently as a result of their unrivalled ability to automatically learn distinguishing features from enormous amounts of data. Surface Electromyography(sEMG) signals can be used to extract hidden characteristics, here seven different gestures are classified by building a deep neural network model. Analysis of the sEMG can be used to forecast the human's intended motion. However, there are frequently a lot of parameters in the models that researchers have lately presented. As a result, the classification accuracy of LSTM model is improved, compared to a simple Convolutional Neural Network(CNN). The Myo Dataset were used to validate the proposed framework. The gesture recognition system produced high classification accuracy. The comparison study on numerous parameters conclude that LSTM model showed the best performance with overall accuracy of 94.78%.

Contents

List of Tables

List of Figures

Abbreviations

CNN	Convolutional Neural Network
CWT	Continuous Wavelet Transform
DT	Decision Tree
KNN	K-Nearest Neighbors
LDA	Linear Discriminant Analysis
LSTM	Long Short Term Memory
MLP	Multi Layer Perceptron
QDA	Quadratic Discriminant Analysis
RMS	Root Mean Square
sEMG	Surface Electromyography
STFT	Short Term Fourier Transform

Notations

C_t	Cell state
f_t	Forget gate
h_t	Hidden state
o_t	Output gate
r_t	Reset gate
x_t	Input gate

Chapter 1

INTRODUCTION

1.1 Overview

Surface Electromyography (sEMG) signals have gained widespread use in a number of disciplines includes the operation of robotic limbs, medical equipment, and human computer interface. The purpose of human hand gestures can now be determined using robotics and artificial intelligence technology, obtained by analyzing the sEMG signals obtained from the patient using an artificial intelligence method. Artificial intelligence and robotics can be used to assist the disabled people to independently carry out some fundamental everyday tasks very effectively. The non stationary sEMG signals, represent the total number of subcutaneous athletic action potentials created through a contraction of the muscles.A smart algorithm uses these signals as one of its primary physical signals to determine the purpose of motion.

Differentiating sEMG signals obtained from various gestures is crucial for related applications that use sEMG signals. The main focus of the literature on gesture detection or artificial limb control by sEMG signals is currently on the temporal and frequency domain feature extraction of sEMG signals, which tries to identify sEMG signals by feature recognition.

Deep learning algorithms have recently been developed by academics, changing the focus from feature engineering to feature learning. The enhancement of the classifier's robustness is the ultimate objective, regardless of the technique used. The quantity of training data that is available is one of the key elements for precise predictions, particularly when using deep learning algorithms. A single user cannot reasonably be expected to produce tens of thousands of samples in one session in the particular setting that hand gesture recognition creates. The main focus of the literature on gesture detection or artificial limb control by sEMG signals is currently on the temporal and frequency domain feature extraction of sEMG signals, which tries to identify sEMG signals by feature recognition.

After years of research, we have identified some efficient frequency and time domain feature combinations. domainhave been proposed, and their respective datasets have yielded some promising results. Given that different movements can be discriminated using conventional approaches, feature extraction is especially crucial. However in traditional approaches, find it challenging to improve the efficiency of sEMG-based gesture recognition. The process of developing and choosing features can be challenging, and there are many different ways to combine features, which increases workload and delivers unsatisfactory results

Researchers have suggested using deep neural networks to separate sEMG signals. Powerful feature extraction is available on CNN. To differentiate between various EMG signals, several studies have employed CNN. The majority of convolutional neural network processing is done on image data.

The amount of the training data has a direct bearing on the correctness of the final test for deep learning systems. Image data is processed using convolutional neural networks the most frequently. First, we employ EMGNet, a small deep CNN for gesture identification. The sEMG signal's temporal information cannot be used by CNN to classify the signal. So that it can't fulfil the goal of inferring gestures directly from the sEMG data, the signals need to be preprocessed by basic filtering. We developed the LSTM model with the above issue in view. The LSTM model addresses the long-term reliance issue by storing important data about past states in a memory cell.The Long Short-Term Memory (LSTM) was chosen for this investigation because it can store important details about prior states and use memory cells to take advantage of the temporal intervals between the data.

1.2 Objectives

1. To create a powerful hand gesture recognition system using CNN and LSTM deep learning algorithms.

2. To simulate and display the outcomes of the suggested models.

3. To compare the accurate model against other deep learning networks.

1.3 Organisation of the report

The previous works related to the thesis are discussed in Chapter 2. The hand gesture recognition is addressed in Chapter 3. The evaluation data set is presented in chapter 4. Chapter 5 discusses deep neural networks Chapter 6 describes the implemented model.Proposed model is detailed in chapter 7. Result and analysis presented in chapter 8 and the conclusion and future scope of the thesis is given in Chapter 9.

Chapter 2

LITERATURE REVIEW

2.1 Overview

This chapter describes literature review of hand gesture recognition. The traditional and advanced methods are discussed.

2.2 Literature survey

The most recent advancement in bio robotics is gesture recognition. Certain arm muscle groups are triggered by gesture patterns. The human arm develops a bio potential as a result of this triggering. The Myo armband records these signals [1].Finding the bioelectric potentials of a human muscle group is done using electromyography [1].The term "Electromyogram" refers to the tool used to record the bioelectric potentials of the muscle group. Surface electromyography (sEMG), one of the electrophysiological signals, is highlighted in these applications. Since the sEMG is one of the most valuable electrophysiological signals and may be used to understand how the human body behaves, it is a relevant alternative as an input signal in human-machine interfaces[2].

The concept of multiple channel acquisition is helpful for hand gesture recognition. Devices used in sEMG processing, such as armbands, help put the electrode on users. Armbands employ signals from muscle groups in the forearm to recognize gestures without precise placement of electrodes on muscles. Armbands are utilized in human-machine interfaces and are a new trend in sEMG acquisition due to their ability to be worn on the upper limbs. This benefit comes at the expense of accuracy when compared to the conventional way of placing precise electrodes. Armbands typically combine acquisition and processing into one unit. In order to avoid low accuracy and enhance the classification time and performance on sEMG armbands, processing strategies are investigated. Additionally, a large quantity of inputs in the pattern recognition step may make it difficult for classifiers to function well while handling sEMG data.

Additionally, feature extraction is a crucial stage in the sEMG processing. To get a high classifier performance, selecting the best feature set is crucial, and each problem has the best feature set. To determine which feature should be extracted, methodologies for feature selection in classification are required. The set ought to be used to a certain classification procedure. In this paper, we contrast two methods for choosing the optimum feature set for each classifier and utilizing dimensionality. classification came before reduction methods. A method for selecting features is combinatorial analysis of characteristics. The combinatorial analysis of all features high is a process that demands computational effort, though, given to the sheer quantity of full sEMG characteristics that are available [2].

Electronic gloves or sensors are used in sensor-based recognition to record the motion and position of hand gestures. There are numerous sorts of sensors, including multi-touch screens, vision-based sensors, and mount bases. We can simply detect the location of the palm and fingers by using the gloves' sensors. Due to the requirement for the user to be linked to the electronic Device, this strategy is rather challenging [3].An example of a sensor-based technique is the data glove. The advantages of this strategy include high accuracy and quick reaction times. The armband's ability to catch muscle groups in the arm and forearm region makes it ideal for signal acquisition.

The system for data collection and processing is created using the Lab VIEW platform. An Artificial Neural Network called a Multi-Layer Perceptron (MLP), K-nearest neighbour (k-NN), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Decision Tree (DT), and Nave Bayes were used to classify gestures (NB).The proposed use of a multiple channel armband for the classification of hand gestures serves to justify the current study. Additionally, while working with the EMG signal, stages can be modularized by using data gathering and conditioning systems through Lab VIEW [4].

In general, it is possible to intuitively frame the recognition of hand motions using instantaneous sEMG images as a problem of image classification that can be resolved by conventional Supervised learning. We develop a classifier to predict the desired hand gesture for each incoming sEMG image using a training set of recorded instantaneous sEMG images labeled with executed hand gestures. We used deep learning [5] to address the sEMG picture classification challenge since its deep convolutional neural network (ConvNet)-based computational model is trained end to end, from initial pixels to final categories, without the need for additional data or manually designed feature extractors.

Thus, this strategy satisfies our needs for experimental verification. We used the deep-learning framework to computationally decode the patterns in the instantaneous sEMG images and identify hand gestures from sEMG photos. An offline training phase and an online recognition step make up this method. An image classifier is taught to determine which hand gesture each sEMG image belongs to during the training phase using the sEMG images and their associated gesture labels. The learned image classifier is used to identify hand motions from sEMG images during the recognition phase[6].

For medical purposes, a notion akin to this known as sEMG topography or sEMG map was presented. An sEMG map is a time-averaged 2D intensity map of sEMG signals, in which each pixel reflects the root mean square (RMS) value of a specific channel in a time window. Recently, sEMG maps have also been used to recognize hand motions .The idea of an instantaneous sEMG image is offered to solve per-frame gesture recognition, which is the main distinction between our instantaneous sEMG image and the sEMG map [7]. Our immediate sEMG picture is generated directly from the raw sEMG signals, but the sEMG map is formed from rectified sEMG signals. Only in the extreme scenario, when the window length of the sEMG map is reduced to one frame, does the sEMG map resemble our instantaneous sEMG image.

For recognizing dynamic hand motions, a deep learning-based model that combines 3D-CNN and LSTM has been developed[8]. For the purpose of accurately identifying and tracking hand postures, hand gesture recognition methods have been developed. In recent years, a variety of techniques for hand gesture recognition have been suggested. Real-time hand gesture identification method based on computer vision. Hand areas are recognized using motion detection and adaptive skin color. Utilizing the Histograms of Oriented Gradients, hand pictures are extracted (HOG). Hand gestures are characterized by the distinctive local distribution of edges and gradients of intensity. The recovered HOG features are projected into a low-dimensional subspace using PCA-LDA. These attributes are further categorized using K-Nearest-Neighbors (KNN).With a 91 percent accuracy rate, eleven different gestures are classified. A CNN-based technique for gesture recognition was proposed by Chung et al. [9]. Using a camera, the method recognizes hand gestures. To distinguish gestures from a complicated background, color space and several morphological operations are applied. Kernel correlation filters are employed to follow the motion of the gesture. The VGG-Net and the Alex Net are two separate models that are given the processed images. Compared to Alex Net, the VGG-Net got a higher recognition rate. The VGG-Net has a 95.61 percent recognition accuracy rate.

Without first preprocessing the region of interest or segmenting the hand gestures contained in the images, a nine-layer CNN is utilized to categorizes the hand gestures. The method that is being used can instantly categorize seven different kinds of hand movements. When the background of the photos was simple, the system's accuracy was 97.1 percent; when the background was complex, it was 85.3 percent. For the purpose of recognizing gestures, Neethu et al. [10] also used a CNN-based classification algorithm. The hand, which is the area of interest, is first separated from the backdrop, and then the input image's contrast is improved with adaptive histogram equalization. Additionally, connected component analysis is employed to segment fingers. The CNN is fed the segmented finger tips in order to categorize various hand movements. The proposed method successfully recognizes gestures against complex backgrounds with an accuracy of 96,2 percent.

Currently, some researchers have investigated numerous efficient network frameworks and effectively applied deep learning to the classification of sEMG signals [11].Literature [6] used CNN to classify sEMG signals, using the raw signals as the input space. Short-Time Fourier Transform (STFT) was used to create the spectrograms of the unprocessed sEMG signals, which were then fed into the convolutional network (ConvNets) [12].ConvNets were employed in literature [7] to categorize the characterizations of the short-time Fourier transform-based spectrogram and Continuous Wavelet Transform-extracted sEMG signals (CWT).We suggested classifying the sEMG signal using a combination of CNN and Long Short-Term Memory (LSTM) from our earlier work [5] since sEMG signals correspond to the timing signal. The ability of CNN to extract features is used while the temporal information in the signal is kept.

By combining CNNs and LSTMs into a single, integrated design, we benefit from their complementary nature. EMG signals can be directly fed into the network using the LCNN and CNN-LSTM models [5].We were able to demonstrate in practice that the LCNN model outperforms CNN-LSTM in terms of performance. The model is optimized with the aid of ADAM. Currently, some researchers have investigated numerous efficient network frameworks and effectively applied deep learning to the classification of sEMG signals [13]. Literature [6] used

CNN to classify sEMG signals, using the raw signals as the input space. Short-Time Fourier Transform (STFT) was used to create the spectrograms of the unprocessed sEMG signals, which were then fed into the convolutional network (ConvNets) [12]. ConvNets were employed in literature [14] to categorize the characterizations of the short-time Fourier transform-based spectrogram and Continuous Wavelet Transform-extracted sEMG signals (CWT).

We suggested classifying the sEMG signal using a combination of CNN and Long Short-Term Memory (LSTM) from our earlier work [15] since sEMG signals match to the timing signal. The ability of CNN to extract features is used while the temporal information in the signal is kept. By combining CNNs and LSTMs into a single, integrated design, we benefit from their complementary nature. The CNN-LSTM and LCNN models, which may directly input pre-processed EMG signals into the network [15], are being studied as a possible alternative to the LSTM. We tested this in practice and found that the LCNN model performs better than CNN-LSTM. As a non-linear activation function, we employ PReLU [12]. However, the LSTM model was introduced in [16] since the ConvNets model in [14] was too complex, which resulted in expensive computation for gesture recognition.

As a result, a new network model was suggested in this research, and it was demonstrated through experimentation that this model not only increases recognition accuracy but also simplifies the network model[17]. In terms of modeling skills, LSTMs and CNNs are complementary because LSTMs excel at temporal modeling and CNNs excel at feature extraction. By combining CNNs and LSTMs into a single, cohesive architecture that we refer to as LCNN, we are able to benefit from their complementary nature [17]. Signal timing information is extracted using the LSTMs model. The CNNs model has the ability to classify signals and extract secondary features. Additionally, we examine the impact of the literary-inspired CNN that is included before the LSTM [18].Compared to CNN, the LSTM model achieves better outcomes.

2.3 Concluding Remarks

This chapter deals with literature review on hand gestures recognition using electromyogrphy signals and computation by various techniques.

Chapter 3

HAND GESTURE RECOGNITION

3.1 Overview

In this chapter, various hand gesture recognition techniques and eight different types of hand gestures included in our dataset are described .

3.2 Experimental and Simulation Motor Parameters

Human-robot interaction (HRI) uses hand gesture recognition to provide intuitive and natural user interfaces. Wearable sensors like data gloves and external sensors like cameras are used for hand gesture identification. Data gloves can offer precise measures of hand attitude and movement, but they frequently cost a lot of money, involve considerable calibration, and limit natural hand mobility. In order to locate the hands and separate them from the background in an image sequence, which is a non-trivial task when there are occlusions, lighting changes, rapid motion, or other skin-colored objects in a scene (for reviews of video-based methods), video-based gesture recognition addresses these issues but also introduces a new problem. Since ancient times, gestures have been used as the main mode of communication in addition to speech. These gesture recognition systems' main objective is to enable elderly and disabled people to carry out daily duties like communicating basic information or operating machinery with the use of simple hand gestures.

The system of communication can help to create human relationships. It makes it simple to communicate with other people and easily share thoughts, sentiments, and affection. The importance of communication is crucial. If we are chatting, walking, playing, sitting, or even sleeping, a message is being created and sent to the person or people. Verbal and nonverbal forms of communication are both possible. Speaking and listening combine to form verbal or spoken communication. People can express their opinions via this communication through face-to-face interactions, the telephone, radio, television, and other media. In non-verbal communication, a person can convey their feelings by body language, gestures (such as swearing, grinning, nodding, or stooping forward), and posture (such as standing in a particular way to convey emotions). Examples of eye contact and smiles and frowns on the face. As an illustration of non-verbal communication, consider the gunshot. The term "gesture" refers to hand, arm, body, head, or face movements that are overly expressive of thoughts, opinions, feelings, etc. The term "recognition" refers to the process of finding and recognising things in a digital media.

This essay aims to Recognition of hand gestures The recognition of gestures is both static and dynamic in style. As the name implies, static gesture recognition emphasizes that the hand position is still consistent during the gesture duration. A dynamic gesture is one in which the hand's position varies continuously during the gesturing period, while a static gesture is dependent on the hand's shape and textural angles. Dynamic gesture based on hand movements, orientations, and shapes in addition to the angles of the fingers' textures. The use of gesture technology enables users to operate or control equipment in a more natural way, similar to how a TV, microwave, or air conditioner remote control works. Gestures feature a variety of hand movements that are created by the hand. While "posture" refers to the contour of the hand, "gesture" refers to the motion of the hand. Robotics, television interface, and sign language recognition are all common applications employed in hand gesture recognition.

When performing the same task, various people will still produce distinct signals, even when a large number of precise control electrodes are used to sense the signals. As a result, it can be difficult to recognize sEMG signals. Deep learning has seen a lot of success in a number of fields, including speech recognition and image categorization. Photos may be accurately categorized by teaching the neural network model to recognize their characteristics. Network architecture is now a topic of deep learning research. Several academics are now investigating a variety of effective network topologies and successfully using deep learning to classify sEMG signals. Using CNN to identify sEMG signals with the raw signals as the input space

Once the location and shape of a hand are known, a classification algorithm that gives predictions of a gesture being made or a posture being held can be used as an input.Utilizing Long Short Term Memory and Convolutional Neural Network, ten different motion types were classified.

The following list of hand motions is classified:

1. Hibernation
2. Flexion
3. Extension
4. Radial deviation
5. Ulnar deviation
6. Pronation
7. Supination
8. Fist

Figure 3.1: Different Gestures

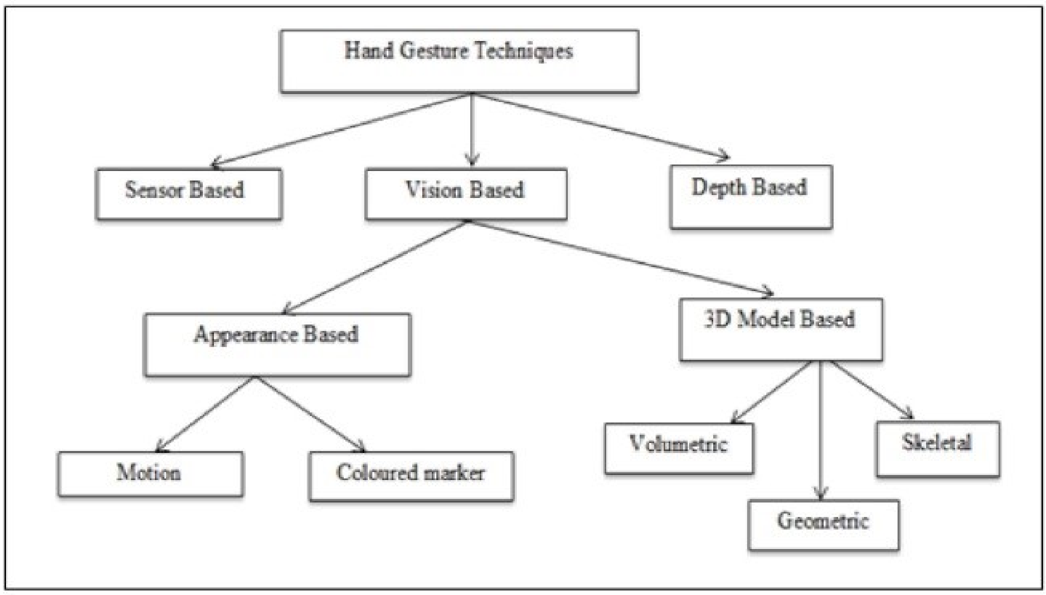

Figure 3.2: Techniques of hand gesture recognition

3.3 Armband For sEMG Signal Extraction

The gesture dataset was collected using the Myo armband.The Myo bracelet is non-intrusive because users may put it on by themselves without any extra preparation thanks to the dry electrodes.For the best possible contact between the subject's skin and the electrodes when using gel-based electrodes, the skin must first be shaved and washed.

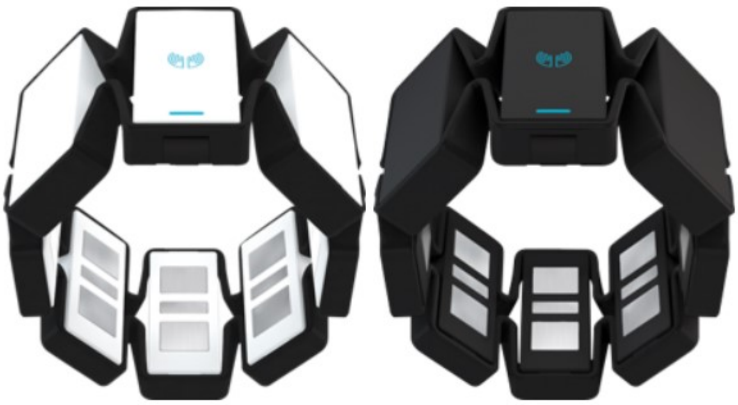

Figure 3.3: Arm band with sensors

The myo armband is typically positioned in the human wrist bulge. Myo Armband is an electromyography readings dataset for wrist fist movements, wrist flexion, wrist extension, radial deviation, ulnar deviation, and hibernation. The thickest region of the right forearm is where the Myo armband is attached, with the LED facing the dorsal (back) side of the hand. Each file should be about 12 000 lines long and comprise one minute's worth of recordings (200 Hz divided by 60 s equals 12 000).Every five seconds, the labels at the ends of the lines switch between hibernation (0) and the gesture identified by the file name. The hibernation gesture is an exception, in which the file only has hibernation labels ((0)) at the ends of the lines.

3.4 Concluding Remarks

This chapter discussed about the differenf hand gestures and armband used to collect data of gestures. Next chapter shows the evaluation data set.

Chapter 4

EVALUATION DATA SET

4.1 Overview

The type of data set used for the validation of sEMG signals, electromyography, suface electromyography and also data preprocessind is detailed.

4.2 Evaluation

There are numerous uses for gesture recognition and categorization in the creation of interfaces for controlling robot systems and human-machine interfaces.The use of the superficial electromyography signal (sEMG) as the system's input has been studied.The bioelectrical processes that give rise to sEMG are created by physiologic changes in the membranes of muscle fibres.To obtain high accuracy during the categorization, the EMG signal's capture, conditioning, and processing steps must be carried out correctly. In addition to EMG signals, other types of input signals that do not require the EMG signal are also employed for categorization, such as video analysis and sensors in instrumented gloves.However, in these circumstances, there is a need for equipment with high resolution in addition to favourable ambient lighting conditions, making the categorization more effective.Given that EMG signals may be obtained in any context, its inclusion in this classification thus shows to be a pertinent suggestion.

The acquisition is often carried out with a small number of channels, looking for the specific location of the channels in the key muscles that are in charge of the referred movements. However, users of common apps do not necessarily know the precise position of each muscle. As a result, armbands make it easier for people to use them by assisting in the search for patterns across muscle groups. The commercial armband Myo, used for mounting databases and analyzing categorization methods, is one of the featured cases. The fact that Myo is a commercial wrist band has the drawback that its circuits and acquisition systems are thought of as "black boxes" because there is no information available regarding the design of the device. Then, it is not possible to change setups for EMG signal acquisition when using Myo.

The EMG signal with a small number of classifiers, concentrating on use and signal preprocessing. The majority of works don't compare different classifiers. This study uses a multiple channel technique to categorize eleven actions (hand gestures) involving the hand and wrist, including wrist flexion, wrist extension, wrist flexion to the left, wrist extension to the right, supination, and pronation. These six actions, depicted in Fig. 1, were chosen because, in addition to being human movements, they may be performed by any healthy person with no issues in the arm or forearm region. The armband that has been suggested is employed for signal acquisition since it has the ability to record muscle groups in the forearm and arm regions.

4.3 Electromyograph

The procedure of electromyography involves identifying the bioelectric potentials of each human muscle group. The term "Electromyogram" refers to the tool used to record the bioelectric potentials of the muscle group. The myo arm band, which is utilized in this experiment to gather the human EMG signals produced after each gesture, is the suggested apparatus. There is a bioelectric potential created in the muscle groups whenever the human arm makes any gesture. Using a myo arm band, these bioelectric potentials are recorded.

4.3.1 Myo Dataset

It was also utilized as the pre-training dataset in the dataset's training and testing portions. The former is primarily used to establish, validate, and enhance the categorization model, which consists of 19 subjects. The latter is only used for validation and final training and has 17 participants.

The Myo Dataset contains 7 different forms of motion, and there are significant variances between them.

4.3.2 Data Preprocessing

The human arm's EMG biopotentials are made up of a variety of noise and other signals.To have clean, processed data that aids in increasing the classification algorithm's accuracy, this noise and other signals should be eliminated.

Data acquisition is the initial step in locating and capturing sEMG signals, processing them to eliminate noise and undesirable signal components.The foundation for processing and analysing sEMG signals is pattern recognition.The pattern recognition system is mostly used to gather motions of the muscular activity to move the helping devices, such as robotic arms or prosthetic limbs.

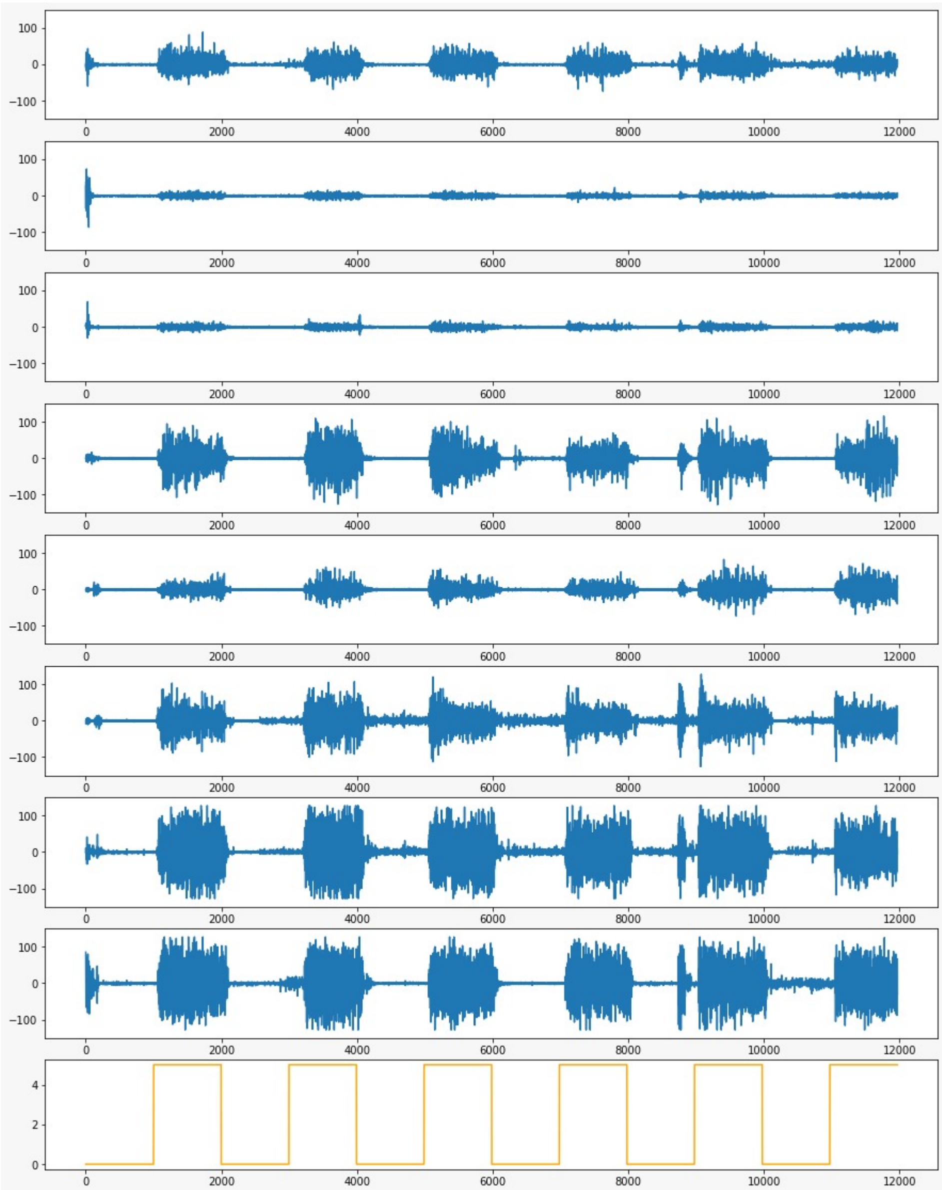

Figure 4.1: Visualization of data

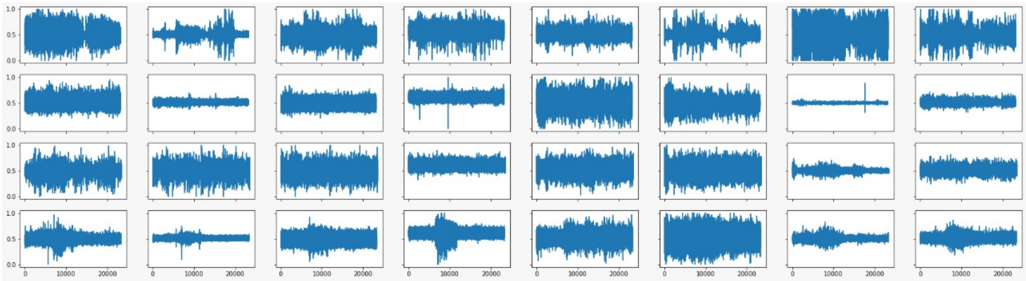

Figure 4.2: Data Representation

4.4 Concluding Remarks

This chapter discussed about the surface emg signals and myo dataset. Next chapter shows the implemented model.

Chapter 5

DEEP NEURAL NETWORKS

5.1 Overview

The basics of Recurrent Neural Network (RNN), Long Short Term Memory (LSTM) and Convolutional Neural Network (CNN) are covered in this chapter.

5.2 Recurrent Neural Network

A class of neural networks called recurrent neural networks (RNN) is useful for simulating sequence data. RNNs, which are derived from feedforward networks, behave in a manner like to that of human brains. In sequential data, recurrent neural networks alone can provide predictions that other algorithms cannot. RNNs are able to accurately forecast what will happen next because of their internal memory, which helps them to retain key details about the input they received. For sequential data such as time series, speech, text, financial data, audio, video, weather, and many more types, they are the algorithm of choice. Compared to other algorithms, recurrent neural networks can develop a far deeper grasp of a sequence and its environment.

You must be familiar with "regular" feed-forward neural networks and sequential data in order to comprehend RNNs effectively. The way that RNNs and feed-forward neural networks channel information gives them their names. Information only flows in one direction in a feed-forward neural network—from the input layer to the output layer, via the hidden layers. Never touching a node more than once, the information travels in a straight line through the network .Feed-forward neural networks are poor at making predictions because they have little recall of the information they receive .A feed-forward network has no concept of time order because it

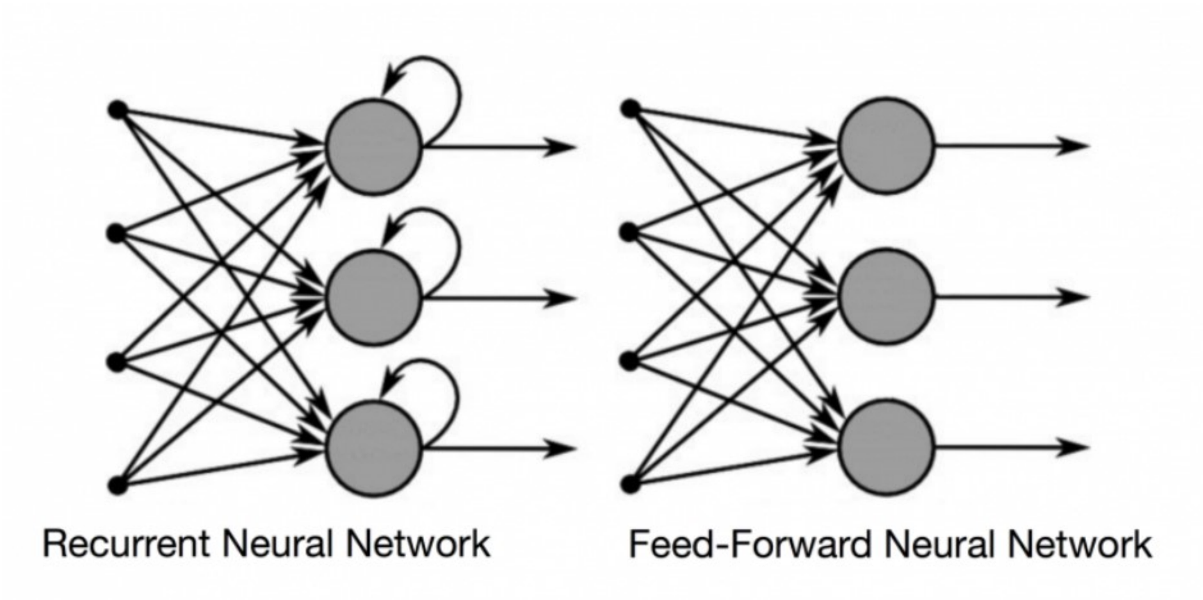

Figure 5.1: RNN vs Feed-forward Network

simply takes into account the current input. It just isn't able to recall anything from the past outside its schooling.

The information flow between an RNN and a feed-forward neural network is depicted differently in the above Fig.5.1.

RNN generates output, duplicates it, and feeds the copy back into the network. Recurrent neural networks integrate the present with the recent past. But because of its inherent memory, a recurrent neural network can recall such characters. Therefore, the present and recent past are the two inputs of an RNN. This is significant because an RNN can perform tasks that other algorithms are unable to, because the data sequence conveys essential information about what will happen next. Short-term memory is a feature of a typical RNN. They have a long-term memory in addition to an LSTM.

5.3 Long Short Term Memory

RNNs with longer memory spans are known as Long Short-Term Memory (LSTM) networks. Building pieces for an RNN's layers are called LSTM. By giving data "weights," LSTMs enable RNNs to either accept new information, forget it, or give it enough weight to affect the result. The layers of an RNN, which is frequently referred to as an LSTM network, are constructed using the units of an LSTM.RNNs can recall inputs for a long time because to LSTMs.

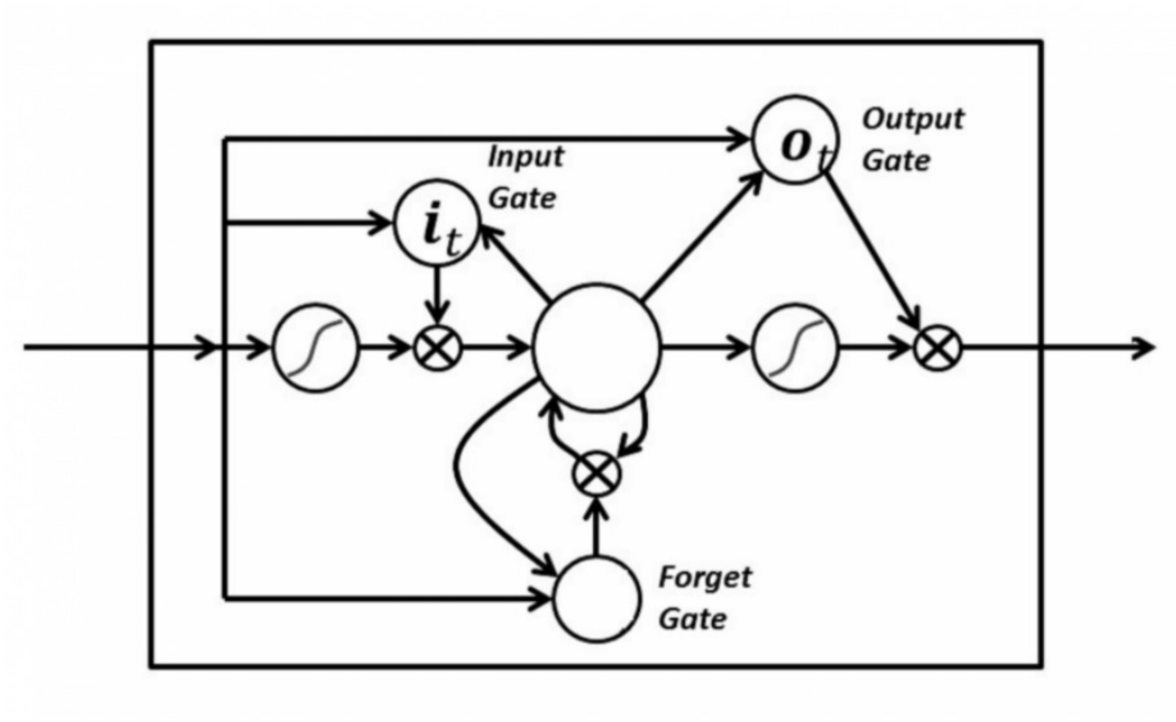

Figure 5.2: Structure

This is due to the fact that LSTMs have a memory that stores information, much like a computer's memory. The LSTM has the ability to read, write, and delete data from its memory.

This memory can be thought of as a gated cell, where the cell determines on the basis of the value it assigns to the information whether or not to store or erase information (i.e., whether it opens the gates or not).Weights, which the algorithm also learns, are used to determine importance. This merely indicates that it gradually comes to understand what information is crucial and what is not. The input, forget, and output gates are the three gates in an LSTM. These gates decide whether to allow additional input (input gate), erase the data because it is unimportant (forget gate), or allow the information to affect the output at the current timestep (output gate).An example of an LSTM with its three gates is shown below:

The analogue sigmoid that make up an LSTM's gates have a range of zero to one. They can perform backpropagation since they are analogue. The issue of disappearing gradients is resolved by LSTM because it maintains gradients that are sufficiently steep, which keeps training time low and accuracy good.

5.4 Convolutional Neural Network

Convolutional neural networks (CNN/ConvNet) are a class of deep neural networks used most frequently to interpret visual data in deep learning. Normally, matrix multiplications come to mind when we think of a neural network, but that is not the case with ConvNet. It makes use of a unique method called convolution . Artificial neurons are arranged in numerous layers to form convolutional neural networks. Artificial neurons are mathematical functions that compute the weighted sum of several inputs and output an activation value, roughly imitating their biological counterparts .Each layer of a ConvNet creates a number of activation functions that are passed on to the following layer when an image is entered. Typically, the first layer extracts fundamental features like edges that run horizontally or diagonally.

The following layer receives this output and detects more intricate features like corners or multiple edges. The network may recognise increasingly more complex elements, including objects, faces, etc., as we go further into it.The classification layer generates a series of confidence ratings (numbers between 0 and 1) that indicate how likely it is for the picture to be a member of a "class," based on the activation map of the final convolution layer. The Pooling layer, like the Convolutional Layer, is in charge of shrinking the Convolved Feature's spatial size. By lowering the dimensions, this will lower the amount of CPU power needed to process the data.

5.5 Concluding Remarks

This chapter provided explanations for recurrent neural networks (RNN), long short term memory (LSTM), convolutional neural network (CNN) neural networks. It is also discussed how each method's cell structure .

Chapter 6

IMPLEMENTED MODEL

6.1 Overview

The model implemented is convolutional neural network, the different layers and activation functions are detailed.

6.2 Convolutional Neural Network

A neural network type called a convolutional neural network, or CNN or ConvNet, is particularly adept at processing input with a grid-like architecture, like an image. A binary representation of visual data is a digital image. It is made up of a grid-like arrangement of pixels, each of which has a pixel value to indicate how bright and what colour it should be. CNNs, often referred to as convolutional neural networks, are a particular kind of neural network that are typically made up of three layers: a convolutional layer, a pooling layer, and a fully connected layer.

6.2.1 Convolution layer

The foundational component of the CNN is the convolution layer. It carries the majority of the computational load on the network. This layer creates a dot product between two matrices, one of which is the kernel—a collection of learnable parameters—and the other of which is the constrained area of the receptive field. Compared to a picture, the kernel is smaller in space but deeper. This indicates that the kernel height and width will be spatially small if the image

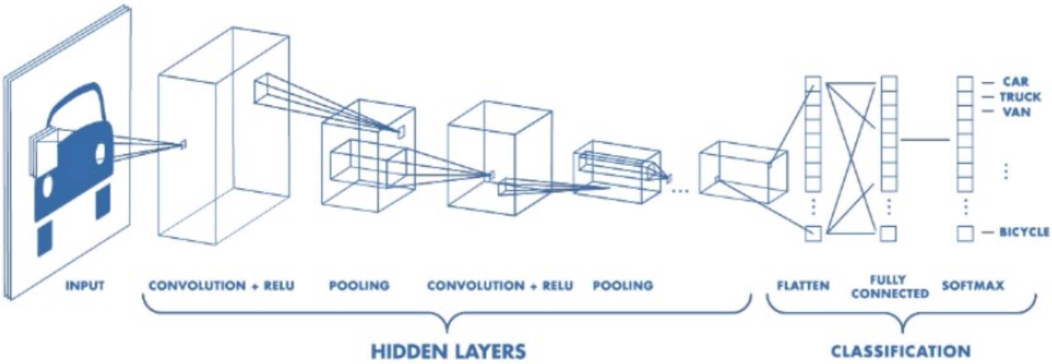

Figure 6.1: Basic structure of CNN

consists of three (RGB) channels, but the depth will go up to all three channels.The output volume may be calculated using the formula below if we have an input of size $W \times W \times D$, Dout number of kernels with a spatial size of F, stride S, and amount of padding P.

$$\mathrm{W_{out}} = \frac{W - F + 2P}{S} + 1 \tag{6.1}$$

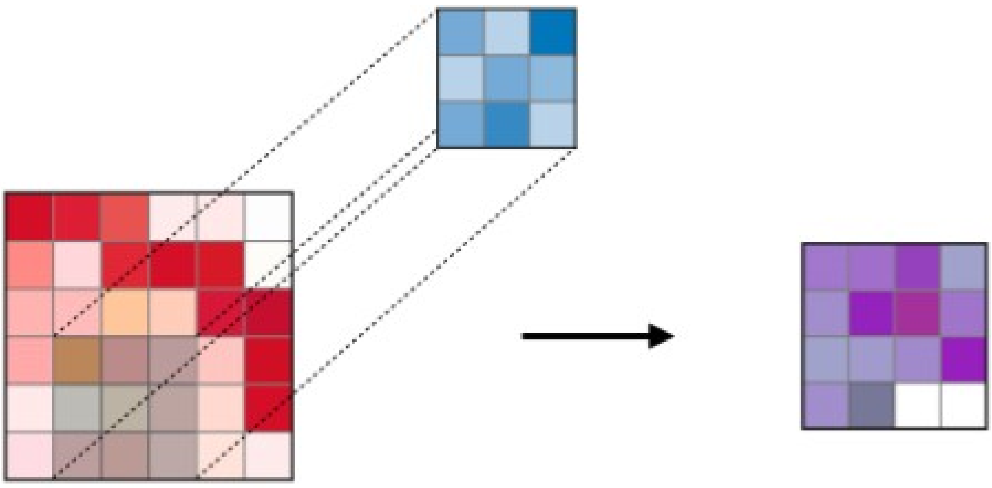

Figure 6.2: Convolution

6.2.2 Pooling Layer

By calculating an aggregate statistic from the surrounding outputs, the pooling layer substitutes for the network's output at specific locations. This aids in shrinking the representation's spatial size, which lowers the amount of computation and weights needed.Each slice of the representation is subjected to the pooling operation separately.

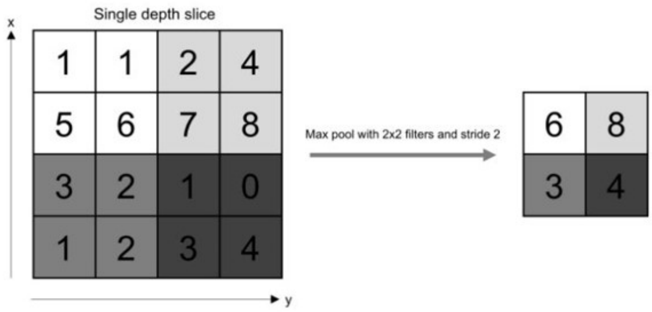

Figure 6.3: Pooling

$$W_{\text{out}} = \frac{W - F}{S} + 1 \tag{6.2}$$

The following formula can be used to calculate the size of the output volume if we have an activation map of size $W \times W \times D$, a pooling kernel of spatial size F, and stride S. An output volume of size $Wout \times Wout \times D$ will result from this.

6.2.3 Fully connected layer

As in a conventional FCNN, all of the neurons in this layer are fully connected to all of the neurons in the layer before and after.Because of this, it can be calculated using a matrix multiplication followed by a bias effect, as per usual.

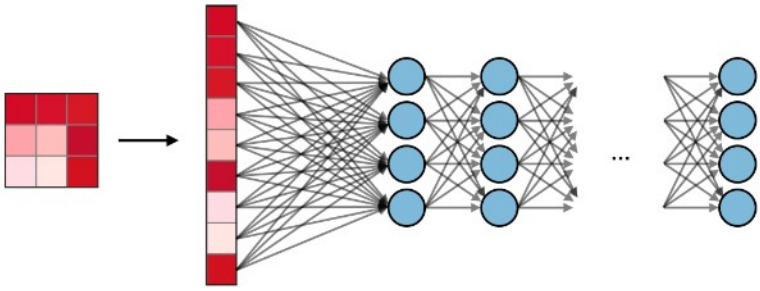

Figure 6.4: Fully Connected

6.2.4 Non linearity layers

Non-linearity layers are frequently included right after the convolutional layer to add non-linearity to the activation map because convolution is a linear operation and images are anything but linear. Non-linear operations come in a variety of forms, the most common ones being:

1. Sigmoid

   The sigmoid non-linearity has the mathematical form.It "squashes" a real-valued number into the range between 0 and 1.The gradient of a sigmoid becomes almost zero when the activation is at either tail, which is a very unfavorable sigmoid feature.

   Back propagation will effectively "kill" the gradient if the local gradient gets too small.Additionally, if the input to the neuron is exclusively positive, the output of the sigmoid will either be exclusively positive or exclusively negative, leading to a zigzag dynamic of gradient updates for weight.

2. Tanh

   A real-valued number is condensed by Tanh to the range [-1, 1]. Similar to sigmoid neurons, the activation saturates, but unlike them, its output is zero-centered.

3. ReLU

   In recent years, the Rectified Linear Unit (ReLU) has gained a lot of popularity. It performs the function ()=max (0,) computation. To put it another way, the activation just exists at zero threshold. ReLU speeds up convergence by six times and is more dependable than sigmoid and tanh. ReLU can, unfortunately, be brittle during training, which is a drawback. It can be updated by a strong gradient that prevents the neuron from ever updating further. However, by choosing an appropriate learning rate, we can make this work.

6.3 Concluding Remarks

This chaped discussed about convolutional neural network. Next chapter shows the proposed model.

Chapter 7

PROPOSED MODEL

7.1 Overview

In this chapter, we present a long short term memory network (LSTM), which can be trained using all available input information in the past and future of a certain time frame, to get around the drawbacks of the convolutional neural network model described in the preceding section.

7.2 Long Short Term Memory

The long-short term memory is a useful tool for dealing with time series prediction (LSTM).Recurrent neural networks (RNNs), which are common cyclic neural networks, come in a number of forms. The most crucial feature of RNNs is the hidden state, which keeps some information or past circumstances concerning the sequence. At its core, LSTM maintains data from inputs that have already been processed by it in a concealed state. It connects many RNN units or memory blocks, each of which has three gate units and one memory cell with recurrent connections. It builds gated cells that respond to input by blocking or passing information depending on the importance of the data element in order to address the increasing gradient problem.

In an LSTM network, information is transmitted through many of these LSTM units.An LSTM unit is composed of three fundamental components.The recurrent neural networks (RNNs), which are the ancestors of the long-term and short-term memory network (LSTM), are able to maintain learning data over the course of learning.The LSTM's internal structure and basic operation are shown in Fig. 7.1.

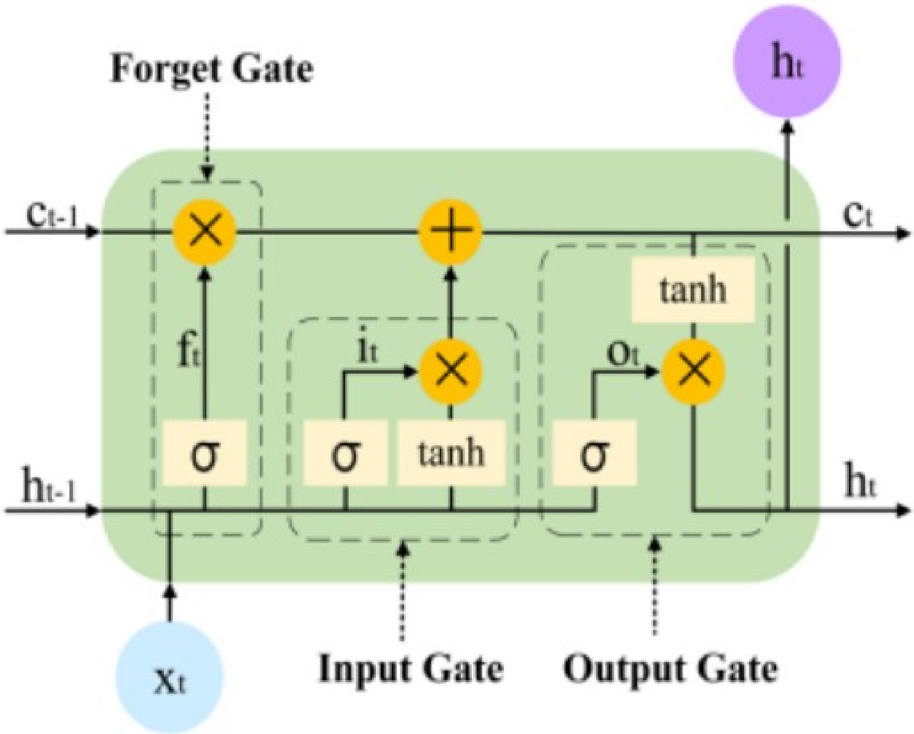

Figure 7.1: Basic structure of LSTM

7.2.1 LSTM model fundamentals

The long-term and short-term memory network (LSTM), which was originally created from recurrent neural networks (RNNs), is capable of preserving learning data throughout the learning process.

7.2.2 Forget Gate

The forget gate decides which information needs to be paid attention to and which can be ignored.The hidden state ht1 and the current input xt are both subjected to the sigmoid function.Sigmoid generates values in the range of 0 and 1.It concludes that the portion of the former production is necessary (by giving the output closer to 1).The cell will eventually multiply points one by one using this value of feet.

$$\mathrm{f}_t = \sigma \left(W_f \times [h_t - 1, x_t] + b_f \right) \tag{7.1}$$

where, t = time step

f_t = forget gate at t

x_t= current input

$h_t - 1$ = previous hidden state

w_f = weight matrix of between forget gate and input gate

b_c = connection bias at t

7.2.3 Input Gate

The input gate modifies the following values to update the cell status. The starting inputs to the second sigmoid function are the current state, xt, and the previously hidden state, ht1.From 0 for important to 1 for unimportant, the values are modified.The same data from the hidden state and current state will then be passed to the tanh function.In order to regulate the network, the tanh operator will supply a vector ct with each possible value between -1 and 1.The output values generated by the activation functions are ready for point-by-point multiplication.

$$\mathrm{i}_t = \sigma\left(W_i \times [h_t - 1, x_t] + b_i\right) \mathrm{c}_t = \tanh(Wc[ht1, xt] + bc) \tag{7.2}$$

where, t = time step

i_t = input gate at t

W_i = weight matrix of sigmoid operator between input gate and output gate

b_i = bias vector at t

c_t = value generated by tanh function

W_f = weight matrix of tanh operator between cell state information and network output

b_c = bias vector at t w.r.t W_f

7.2.4 Cell State

The network has received enough data from the input gate and forget gate.Next, a decision is made, and the data from the new state is stored in the cell state.The previous cell state ct is multiplied by the forgets vector ft by 1.If the outcome is 0, values will be eliminated from the cell state.The output value of the input vector it is then added point-by-point by the network, updating the cell state and generating a new cell state c_t.

$$\mathbf{c}_t = f_t \times c_t - 1 + i_t \times c_t \tag{7.3}$$

where, t = time step

c_t= cell state information

f_t = forget gate at t

$c_t - 1$ = previous cell state at t-1

7.2.5 Output Gate

The output gate determines the value of the next hidden state. This state contains details about earlier inputs. With the values of the current state and the prior hidden state, the third sigmoid function is initially called.The newly formed cell state that was derived from the initial cell state is then subjected to the tanh function. One by one, these two results are multiplied. The network decides which information the concealed state should convey based on the final value.Next, the new cell state and hidden state are carried over to the subsequent time step eq. The forget gate chooses which pertinent data from the prior processes is necessary to finish.The input gate decides what crucial information may be obtained from the current stage, while the output gates complete the subsequent hidden state.

$$\text{ot } = \sigma(\mathrm{Wo} \times [\mathrm{ht} - 1, \mathrm{xt}] + \mathrm{bo}) \tag{7.4}$$

$$ht = ot \times \tanh(\mathrm{ct}) \tag{7.5}$$

where o_t = output gate at t

W_o= Weight matrix of output gate

b_o= bias vector w.r.t W_o

h_t = LSTM output

7.2.6 Training In LSTM Network

The unique architecture of LSTM allows it to forget unimportant data. The sigmoid layer decides which parts of the old output from the x_t and inputs should be eliminated (by outputting a 0). Forget Gate ft is the name of this gate. The output of this gate is $f_t * \ c_t - 1$.The following step is to make a choice and save the information from the new input with the cell state. A Sigmoid layer determines whether of the new bits of information should be updated or ignored. A tanh layer creates a vector with all possible values using the new input. These two are multiplied in order to update the new cell state. The current memory ct is then produced by combining the previous memory c_t.

7.3 Block Diagram

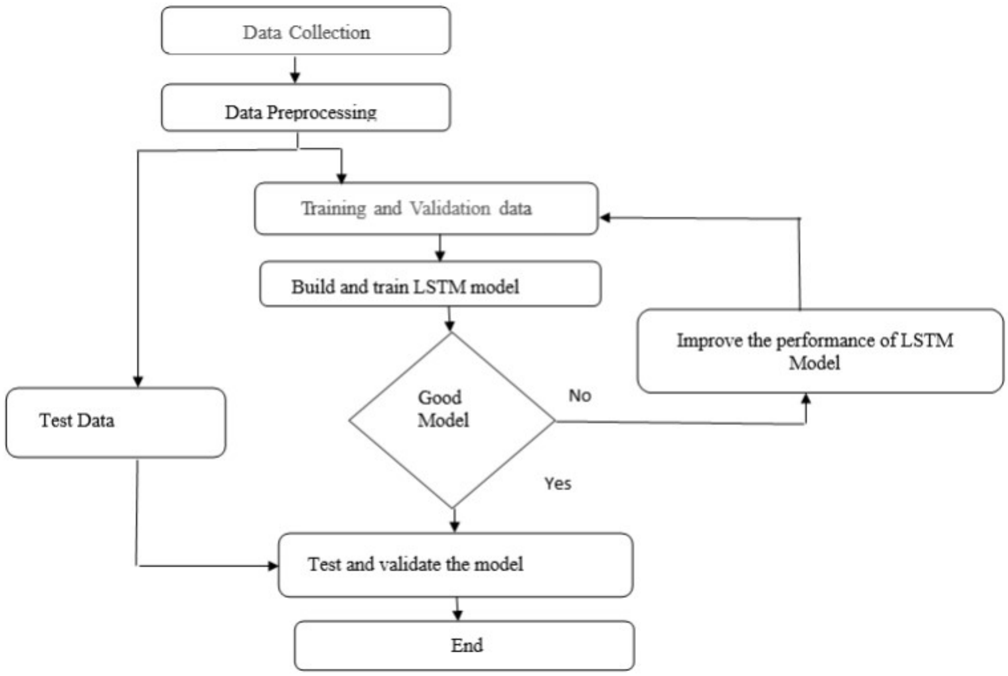

Figure 7.2: Flowchart of proposed model

The flowchart summarizes the main steps. The proposed approach includes four key steps. Collect the data from the armband. Then pre-process and clean data by removing outliers and imputing missing values, normalize the orginal data. Then train, validate and test the LSTM model.

7.4 Concluding Remarks

This chapter discussed about proposed model that is LSTM. Next chapter shows the result and analysis

Chapter 8

Result and Analysis

8.1 Overview

This chapter discusses the comparison of two deep learning techniques for hand gesture recognition is explained.

8.2 Result Of CNN

The result includes the model accuracy and loss curves of training and validation and the confusion matrix. The CNN method produce an accuracy of 92.96%

8.2.1 Accuracy and loss curve

The curve with accuracy in both training and validation is more significant. Overfitting is evidently present when there is a variance in accuracy between training and validation. From figure 8.1 it is clear that both the training and validation curve are similar, so there is no chance of overfitting or underfitting. The loss curve calculates model error and indicates how poorly our model is performing. Therefore, for the time being, the lower our loss, the better our model's performance will be. The output graph obtained is shown in Fig.8.1 and Fig.8.2

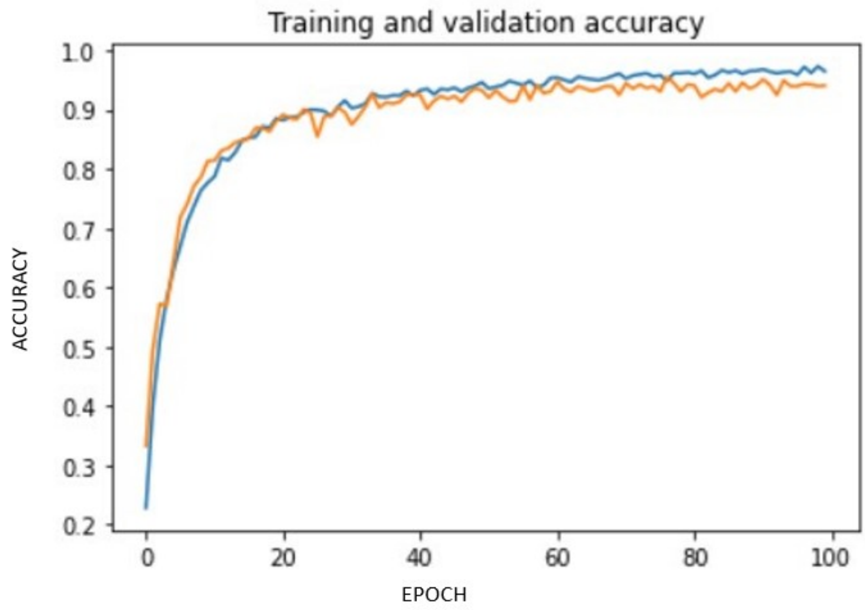

Figure 8.1: Model accuracy of CNN

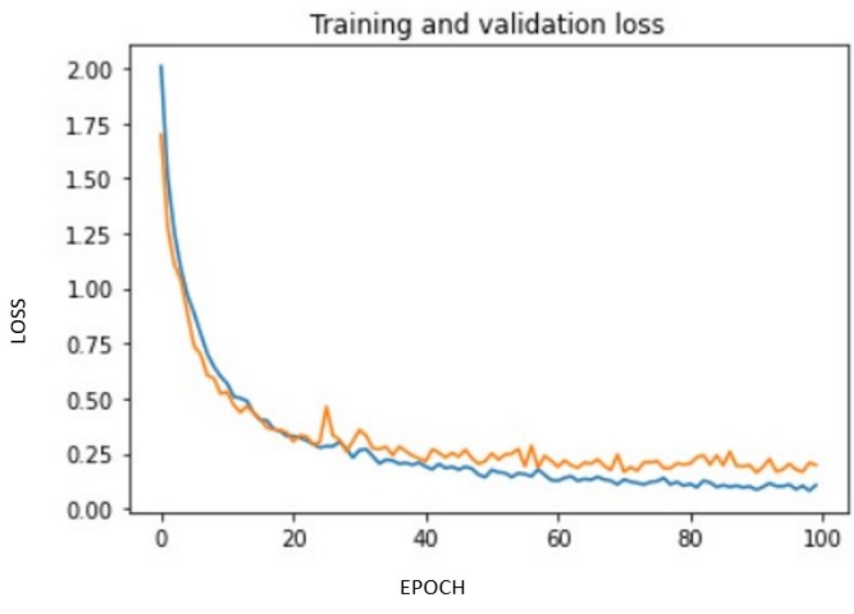

Figure 8.2: Model loss of CNN

8.3 Result of LSTM

The result includes the model accuracy and loss curves of training and validation and the confusion matrix.The resulting graphs are overlapping one another. The LSTM method produce an accuracy of 94.78%

8.3.1 Accuracy and loss Curves

The Fig.8.3 shows the model performance on the training and testing datasets and also diagnose an underfit, overfit and goodfit model.

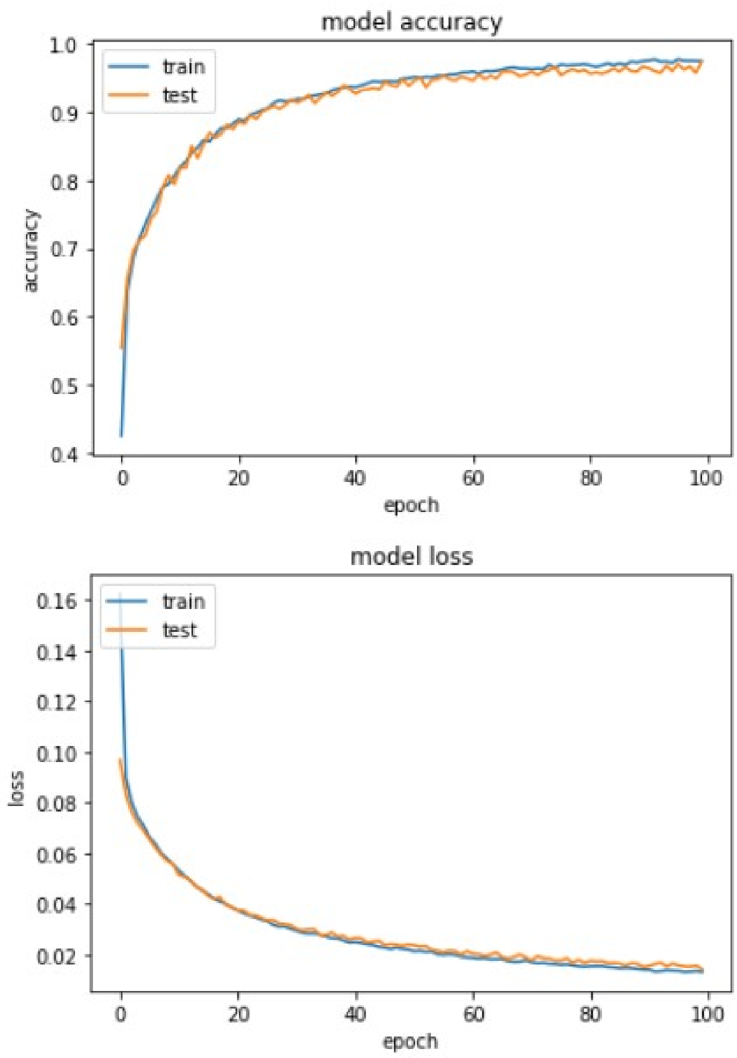

Figure 8.3: Model accuracy and loss of LSTM

8.3.2 Confusion Matrix Evaluation

A N x N matrix known as a confusion matrix is used to assess the effectiveness of a prediction model, where N is the total number of classes. The predicted goal values in the deep learning model are contrasted with the actual goal values in the matrix. This provides us with a full understanding of the mistakes that our categorization model is making as well as how well it is functioning. Information retrieval depends on both recall and precision, thus they are both essential. It also gives the positive classes priority when contrasted to the negative classes. Only the true positive (TP), false positive (FP), and false negative (FN) are used in terms of precision and sensitivity; true negative is ignored (TN).

$$Precision = \frac{TP}{TP + FP} \tag{8.1}$$

Out of all the positive predictions, precision calculates the proportion of actual positive results.

$$Precision = \frac{TP}{TP + FN} \tag{8.2}$$

In contrast, recall provides the expected positive proportion of all positive results.

$$F1score = 2 * \frac{P * R}{P + R} = \frac{2TP}{2TP + FP + FN} \tag{8.3}$$

The arithmetic mean of sensitivity and precision is known as the F1-score.

The figures below displays the confusion matrixes that was obtained for the various deep learning models.

True positive(TP) means that predicted class and actual class both are positive, True negative(TN) define predicted class and actual class both are negative, when predicted class is positive but actual class is negative then it is False positive(FP) and finally False negative(FN) means that predicted class is negative but actual class is positive.The performance of the proposed direction prediction model has been validated using criteria including accuracy, precision, recall, and f1-score. The mathematical representation of these parameters are as,

A common tool for assessing classification algorithm performance is the confusion matrix. It is a matrix having n number of rows and coloumns, where n stands for the number of classes. Given that there are 8 classes, we receive a $8/8$ matrix here. it is plotted against the predicted value and actual value.The matrix's diagonal members display the true positive values.Because

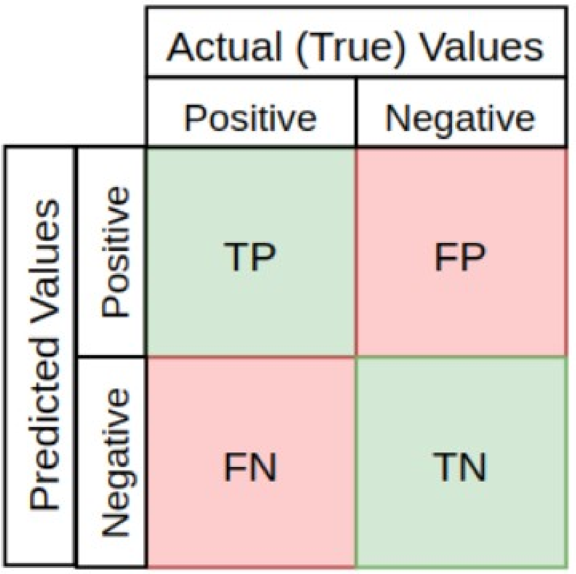

Figure 8.4: Confusion matrix depicting classes

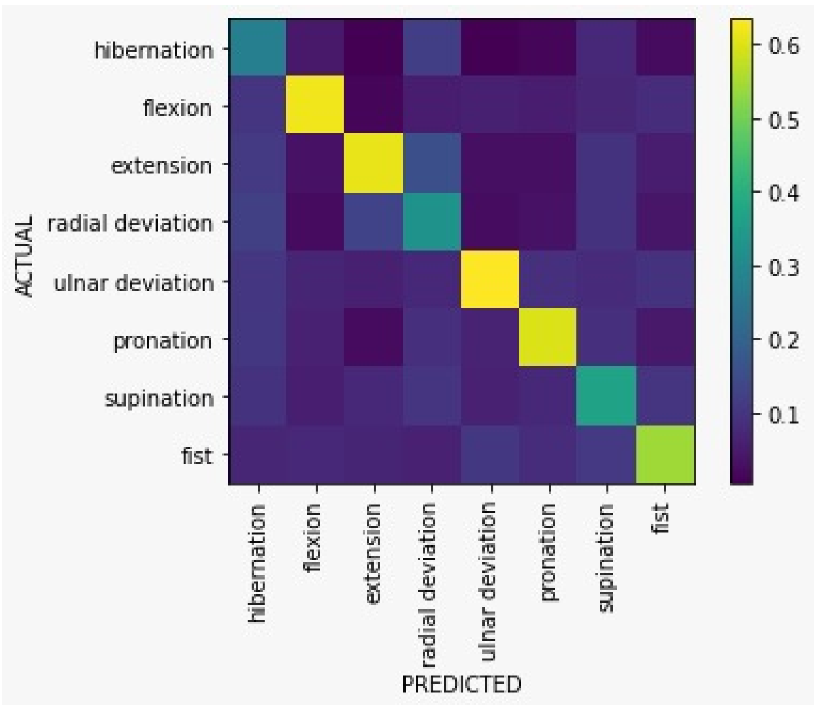

Figure 8.5: Confusion matrix of CNN

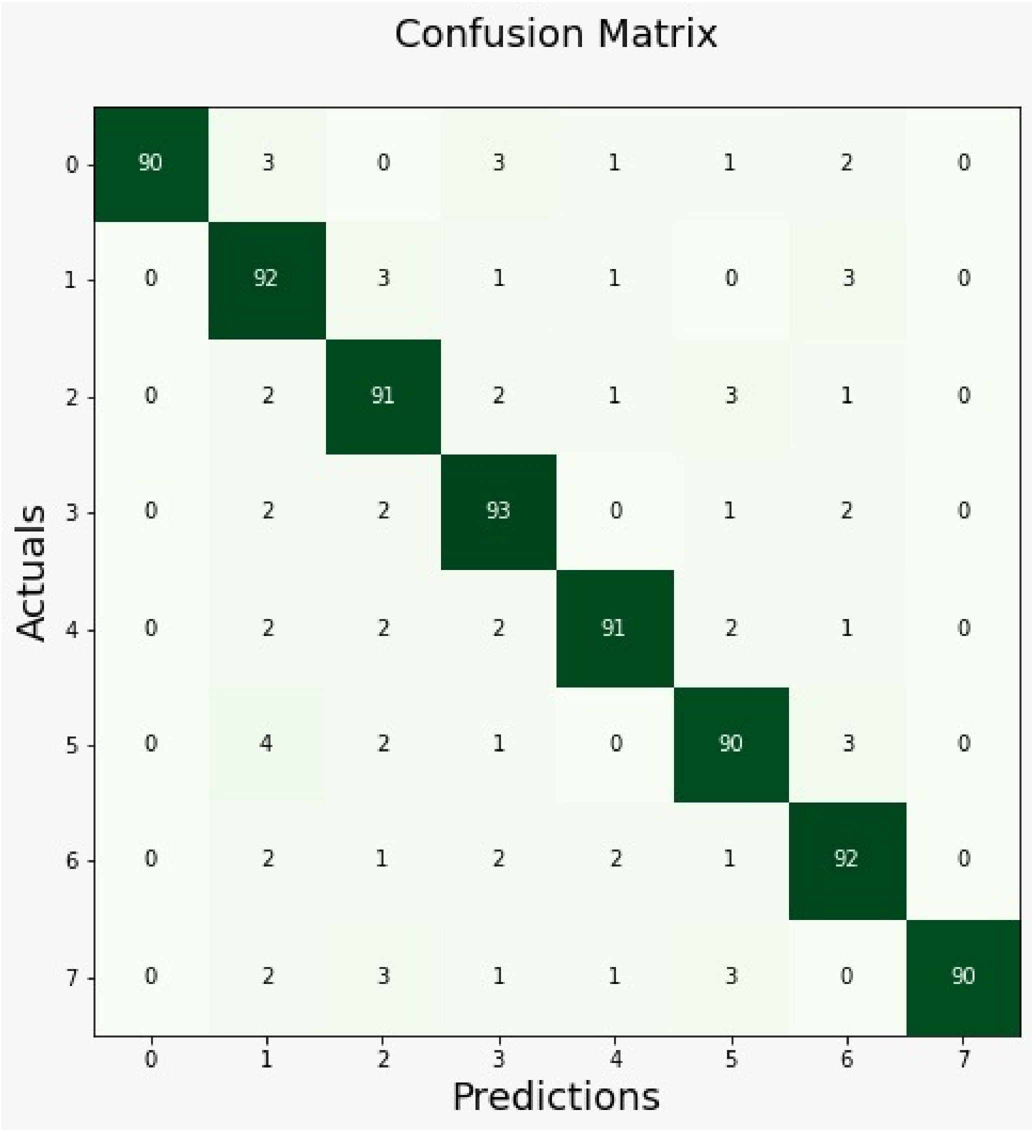

Figure 8.6: Confusion matrix of LSTM

of Multi-Class Classification, the false positive values are determined by the sum of the corresponding columns except the true postive value. For example, in Fig 5.1, the false postive value of PT class is 1. Similarly, the false negative value is calculated by the sum of values of the corresponding row except the true postive value. The true negetive value is calculated by the sum of values of the corresponding row and column, where the true positive case is expected.

8.4 Comparison of Two Models

Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) are taken into account for comparison purposes, and the network models for each technique are generated using the identical procedures. Each model's output is assessed, and the relevant conclusions are drawn. The accuracy of the CNN model was 92.96%, while the output of the LSTM technique was 94.78%. There was a significant difference between the two methods when testing with various sensor values. Additionally, these models and user input did not accurately manage wheelchair navigation.

Table 8.1: Comparison of various deep learning techniques

DL model	Metrics	0	1	2	3	4	5	6	7
LSTM	Precision	1.00	0.77	0.79	0.83	0.65	0.48	.43	1.00
	Recall	0.77	0.78	1.00	0.95	0.81	0.56	0.40	0.46
	f1-score	0.87	0.78	0.88	0.89	0.72	0.52	0.41	0.63
	Accuracy	94.78%							
CNN	Precision	0.28	0.48	0.58	0.36	0.71	0.64	0.36	0.55
	Recall	0.68	0.55	0.30	0.38	0.29	0.36	0.39	0.44
	f1-score	0.40	0.51	0.40	0.37	0.41	0.46	0.37	0.49
	Accuracy	92.96%							

Chapter 9

CONCLUSION AND FUTURE SCOPE

The research described in this study helps impaired people communicate certain fundamental daily exchanges. Here, we used two deep learning methods like CNN and LSTM to classify ten various types of motions. The two models were constructed using the EMGnet Architecture. On the Myo dataset, the accuracy and precision of the proposed algorithms' performances were evaluated. Our study demonstrated considerable improvements with the LSTM approach, demonstrating its robustness and dependability with an accuracy of 94.78 percent. The accuracy of the LSTM approach is clearly higher than that of the CNN approaches, coming up at 92.96 percent.The future scope of the work include real time prediction for deaf people.

REFERENCES

[1] K. S. Krishnan, A. Saha, S. Ramachandran, and S. Kumar, “Recognition of human arm gestures using myo armband for the game of hand cricket,” in *2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*, pp. 389–394, IEEE, 2017.

[2] J. J. A. Mendes Junior, M. L. Freitas, H. V. Siqueira, A. E. Lazzaretti, S. F. Pichorim, and S. L. Stevan, “Feature selection and dimensionality reduction: An extensive comparison in hand gesture classification by semg in eight channels armband approach,” *Biomedical Signal Processing and Control*, vol. 59, p. 101920, 2020.

[3] H. Kaur and J. Rani, “A review: Study of various techniques of hand gesture recognition,” in *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, pp. 1–5, IEEE, 2016.

[4] M. L. B. Freitas, J. J. A. Mendes, D. P. Campos, and S. L. Stevan, “Hand gestures classification using multichannel semg armband,” in *XXVI Brazilian Congress on Biomedical Engineering*, pp. 239–246, Springer, 2019.

[5] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[6] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, and J. Li, “Gesture recognition by instantaneous surface emg images,” *Scientific reports*, vol. 6, no. 1, pp. 1–8, 2016.

[7] M. Rojas-Martínez, M. A. Mañanas, and J. F. Alonso, “High-density surface emg maps from upper-arm and forearm muscles,” *Journal of neuroengineering and rehabilitation*, vol. 9, no. 1, pp. 1–17, 2012.

[8] M. U. Rehman, F. Ahmed, M. A. Khan, U. Tariq, F. A. Alfouzan, N. M. Alzahrani, and J. Ahmad, "Dynamic hand gesture recognition using 3d-cnn and lstm networks," *Comput. Mater. Contin*, vol. 70, pp. 4675–4690, 2022.

[9] N. Alnaim, M. Abbod, and A. Albar, "Hand gesture recognition using convolutional neural network for people who have experienced a stroke," in *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pp. 1–6, IEEE, 2019.

[10] G. Strezoski, D. Stojanovski, I. Dimitrovski, and G. Madjarov, "Hand gesture recognition using deep convolutional neural networks," in *International conference on ICT innovations*, pp. 49–58, Springer, 2016.

[11] U. Côté-Allard, C. L. Fall, A. Campeau-Lecours, C. Gosselin, F. Laviolette, and B. Gosselin, "Transfer learning for semg hand gestures recognition using convolutional neural networks," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1663–1668, IEEE, 2017.

[12] U. C. Allard, F. Nougarou, C. L. Fall, P. Giguère, C. Gosselin, F. Laviolette, and B. Gosselin, "A convolutional neural network for robotic arm guidance using semg based frequency-features," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2464–2470, IEEE, 2016.

[13] F. Orabona, C. Castellini, B. Caputo, A. E. Fiorilla, and G. Sandini, "Model adaptation with least-squares svm for adaptive hand prosthetics," in *2009 IEEE International Conference on Robotics and Automation*, pp. 2897–2903, IEEE, 2009.

[14] U. Côté-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 27, no. 4, pp. 760–771, 2019.

[15] Y. Wu, B. Zheng, and Y. Zhao, "Dynamic gesture recognition based on lstm-cnn," in *2018 Chinese Automation Congress (CAC)*, pp. 2446–2450, IEEE, 2018.

[16] T. Tommasi, F. Orabona, C. Castellini, and B. Caputo, "Improving control of dexterous hand prostheses using adaptive learning," *IEEE Transactions on Robotics*, vol. 29, no. 1, pp. 207–219, 2012.

[17] L. Chen, J. Fu, Y. Wu, H. Li, and B. Zheng, "Hand gesture recognition using compact cnn via surface electromyography signals," *Sensors*, vol. 20, no. 3, p. 672, 2020.

[18] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 4580–4584, IEEE, 2015.

List of Publications

[1] Baby Rukhiya, Sumayya Jaleel, "Hand Gesture Recognition Using Deep Learning Techniques," Communicated to *International Conference On Multimedia, Signal Processing and Communication Technologies(IMPACT)*, 2022.-

[2] Baby Rukhiya, Sumayya Jaleel , "Hand Gesture Recognition Using LSTM Model," Communicated to *IEEE 19th India Council International Conference (INDICON)*, 2022.