Recognition of Bangladeshi Sign Language From 2D Videos Using OpenPose and LSTM Based RNN

by

Tanmoy Dewanjee
16301150
Azibun Nuder
16301045
Md. Imtiaz Malek
16101068
Refah Nanjiba
16301153
Atia Anjum Rahman
16301002

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
January 2021

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Students' Full Name & Signature:

Tanmoy Dewanjee
16301150

Atia Anjum Rahman
16301002

Refah Nanjiba
16301153

Md. Imtiaz Malek
16101068

Azibun Nuder
16301045

Approval

The thesis titled "Recognition of Bangladeshi Sign Language From 2D Videos Using OpenPose and LSTM Based RNN" submitted by

1. Azibun Nuder (16301045)
2. Atia Anjum Rahman (16301002)
3. Md.Imtiaz Malek (16101068)
4. Refah Nanjiba (16301153)
5. Tanmoy Dewanjee (16301150)

Of Fall, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 8, 2021.

Examining Committee:

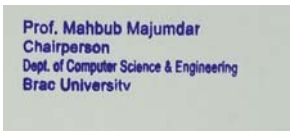
Supervisor:
(Member)

Md. Golam Rabiul Alam, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Md. Saiful Islam
Lecturer
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)



Dr. Mahbub Alam Majumdar
Professor and Dean
Department of Computer Science and Engineering
Brac University

Abstract

Sign-language recognition is an essential part of computer vision to solve a communication obstacle between the deaf-mute and the common. Bangladeshi Sign Language (BdSL) is the medium of communication of the deaf and dumb community of Bangladesh. Where 2.4 million people cannot communicate without a sign language, developing countries like Bangladesh do not have sufficient facilities for these people [34]. Our research represents a sign-language recognizer in Bangladesh which is an approach to understanding Bangladeshi sign language so that it can become a bridge between the deaf-mute community and the normal world. Though many works have been done in this field for foreign languages, there are only a few remarkable works on the Bangladeshi Sign Language, among which they used techniques that are not accessible to all, and their accuracy was also not satisfactory. Moreover, there is a shortage of publicly available datasets of Bangladeshi Sign Language. Our objective is to deliver a compact and highly accurate system that will recognize Bangladeshi Sign Language. We propose an method based on estimation of the human keypoints. First of all, we develop a BdSL dataset containing 1151 videos with ten different words. Our algorithm uses OpenPose to extract human pose from 2D videos and feed the extracted features keeping their temporal nature to an LSTM based RNN classifier that accurately classifies the signs. Our proposed sign language model classifies the signs of Bangladeshi Sign Language with 96.54% accuracy.

Keywords: BdSL; Video Processing; Machine Learning; Sign Language; Classification; OpenPose; Deep Learning; LSTM based RNN

Dedication

We want to dedicate this thesis to our loving parents and all the amazing faculties we encountered and learned from in the course of pursuing our Bachelor's degree.

Acknowledgement

First and foremost, praises and thanks to Allah, the Lord of the entire universe, for showering His blessings on us, giving us perseverance throughout our research work so that we could complete it successfully.

We want to express our deep and sincere gratitude to our supervisor Md. Golam Rabiul Alam, Ph.D., for giving us the opportunity to do this study and thesis and for his continuous support and guidance all over the thesis work. Be it his patience, enthusiasm, motivation, vision, or insightful comments; he enormously inspired us to learn more and know more. Then we would like to thank our co-supervisor, Md. Saiful Islam, who never hesitated to give his precious time to us whenever we failed to understand anything. It was a great honor and privilege to work under their direction. Furthermore, we would like to show our appreciation to the Department of Computer Science and Engineering for providing an amazing working atmosphere that excites learning and also for providing resources that aided our research. Last but not least, we would express our love and regards to our parents and friends, who were our pillars of never-ending moral-support. We would not have come this far and be proficient in it without any of them.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AUC Area Under Curve

$BdSl$ Bangladeshi Sign Language

BSL British Sign Language

$COCO$ Common Objects in Context

CSL Chinese Sign Language

FN False Negative

FP False Positive

$GNBC$ Gaussian Naive Bayes Classifier

ISL Indian Sign Language

KNC K-Neighbor Classification

$LSTM$ Long Short Term Memory

$MPII$ Max Planck Institut Infomatik

NB Naive-Bayes

NBC Naive Bayes Classifier

PAF Part Affinity Fields

$ReLU$ Rectified Linear Unit

RNN Recurrent Neural Network

ROC Receiver Operating Characteristic

SVC Support Vector Classifier

SVM Support Vector Machine

TN True Negative

TP True Positive

Chapter 1

Introduction

This chapter presents the subject, discusses the issue, and provides a brief overview through this review of the study, the intent, aim, and method of solving it.

The essential means of interaction for dumb and deaf people is sign language. A pictorial sign for each language, with separate hand movements, positions, and gestures, requires both hands. In general, there are a variety of static and interactive symbols for the letters and the words [44]. In Bangladeshi sign language, the alphabet is represented by a single hand with 38 symbols (27 for consonants and 9 for vowels). Simultaneously, for the Bengali words, there are around 4000 double-hand symbols. The disability of being dumb or deaf is a little different from most disabilities. However, being affected by this disability in some ways will make it impossible for an everyday activity to be done by a deaf or mute person. While communicating with others, things are complicated for the deaf and the mute. The situation is better within a mute and deaf group since everyone understands the sign language, but they have somewhat worse experiences engaging with other individuals afflicted with antipathy. A human translator may play an essential role in either scenario. Unfortunately, it is expensive to keep anyone at hand and is not always necessary. The people who use Bangladeshi sign language (BdSL) are also faced with the problem because of the shortage of interpreters of BdSL. Computer vision systems to automatically interpret sign languages have been released recently to overcome this challenge. However, to date, there is not sufficiently credible work to identify BdSL. A human translator may play an essential role in this. Unfortunately, it is expensive to keep anyone at hand and not always necessary—the need for an easy-to-use and cost-effective interpreter. The proposal is about recognizing different signs of BdSL with the help of video processing and deep learning networks.

1.1 Motivation

Communication is called the transfer or sharing of information through speaking, writing, or other mediums. Signs and expressions are used much of the time for communicating. Ever since the creation, man's ideas and feelings have been attempted to connect and to express. They then continue to communicate with them in their native languages and increasingly invent signs or alphabets for their purposes. As time went on, the number of the world's inhabitants grew, people continued to split with varying personalities, lived in different locations, and so on. There are also

individuals with physical disabilities like deaf, blind, visual handicaps, Etc. They need to interact by using movements with their hands in many ways, for example. Normal humans use natural language to talk or connect, while dump and silent people use visual sign language to interact. The only way men can communicate is by sign language. Today, because of the immense competition in every area, it is challenging for people with disabilities to participate in the real world. An interpreter (one who understands both the sign language and the usual) is required to ensure hassle-free contact between the ordinary person and the deaf and dumb person.

We can divide sign language into two sections – sign language visual and sign language tactile. Individuals with impaired hearing and voice use Visual Sign Language. Tactile Sign Language is used for individuals disabled by hearing and sight.

We work on dumb and mute visual sign language. In comparison, there is no international sign language standard. Various nations use gestures of their kind. In each country, the sign language varies. It depends on the society, as the sign language in India is ISL, America uses ASL, China uses CSL (Chinese Sign Language). The standard in Bangladesh is the Bangladesh Sign Language Anthology (BSLA). Bangladeshi Sign Language (BdSL) has a form that does not complement other countries' sign languages. As English is the first universal language, British sign language has many works. People are working with the sign language of Bangla in our region. In 1974, a book was written by the National Centre for Special Education Ministry of Social Welfare [44], called the "Bengali Sign Language Dictionary. "In 2015, an alternate book entitled 'Ishara Bhasay Jogajog' was reproduced in Bangladesh to educate deaf children. We have agreed to work with Bangladeshi sign language recognition for this region.

Sign language is a contact medium for the deaf and mute, with various gestures composed of forms of hands and body position. Each conduct has its importance. The alphabets of the sign language are comprised of many fingertips, and the term consists of hands. Even visual sign language is used in facial movements. A visual sign is vital for the deaf and silent as a contact medium. The deaf and silent are difficult to embed into the mainstream, notably because culture lacks sign language knowledge. Scientists have also developed methods to develop automated systems to understand sign language to overcome this communication void. This field of study is still well behind many fields and is still in trouble.

In comparison, studies on recognizing the English language of signs did not prevail in specific other languages of signs. So, to understand Bangladeshi Sign Language, we aim to conduct research. Our study aims to contribute to the Sign Language Recognizer process (SLR). The objective is to create a BdSL interaction interpreter that makes it easier for deaf and silent people to communicate with regular people. When communicating, this system will identify enacted words from a video of dumb and deaf individuals. Although there are two kinds of motions - (one and both), we'll focus mainly on both hands' movements. Our analysis only supports people with speech, and hearing difficulties can quickly and effectively interact.

1.2 Problem Statement

The primary mode of contact for people who are deaf and mute is sign language. It is challenging to include them in the mainstream, and much of the population does not know sign language. Sign languages are natural languages that are based on a region's environment and culture, like other natural languages. The most study was carried out in this area on recognizing American Sign Language (ASL) and Indian Sign Language (ISL). Research based on Bangladeshi sign language recognition has not prevailed for ASL and other countries' sign language. With hundreds of millions of speakers, Bengali is the five most commonly spoken language in the world. It is Bangladesh's official language and the second most spoken language in India. Given its distance, an AI built to understand and translate Bangladeshi Sign Language to Bangla would enhance and ease the lives of millions of dumb and deaf people and the rest. Some studies were carried out recently to recognize static signs referring to alphabets in the English language. However, alphabets are also used in the spoken language for pronouncing proper nouns and unfamiliar words on their fingertips instead of sign languages that are entirely different from the spoken language. To overcome the interaction barrier for people with hearing and speech disability and the rest of Bangladeshi people, we propose a computer vision and neural network-based approach which, through video recognition that classifies dynamic signs which match words and generates the desired words. The developed framework can be used later on in hardware to facilitate daily communications between ordinary and deaf and mute people.

1.3 Research Objectives

Our research goal was to have an effective and robust method. Dumb and deaf people belong to our culture. They will become a significant asset for society if they gain enough support from others. They must overcome certain obstacles in order to succeed. We decided to simplify their lives. Previously, a little work is being done on the recognition of the Bangladeshi sign language. Even they used old techniques. That is why we want to deliver a system with high accuracy by using the new method. Moreover, Our objective is -

1. Minimizing the communication barrier between the mute, deaf, and the regular.

2. Build a video-processing and neural network-based solutions system to understand the Bangladeshi sign language.

3. To form Bengali sentences with the output of the recognizer which will help our Bengali community of mute and deaf to be connected through a device to the world.

4. To create an environment where Bengali words will be equally easy to use for all sorts of people in our Bengali community to exchange thoughts with each other.

1.4 Thesis Outline

We addressed the previous work and assessments based on that process and similar attempts and the disadvantages we approached. The first section of chapter 2 is in the sequence below. The remaining chapters are ordered consecutively. The background details we have collected in Chapter 3 are explored in depth. Based on these, Chapter 4 addresses our model and how we comprehensively create the work environment. Chapter 5 addresses the outcomes we have achieved and their consistency and results. Chapter 6 summarizes the article with our final views on the topic.

Chapter 2

Background Study

Sign language is a visual way of expressing thoughts and their placement through hand gestures with respect to the upper body, body postures, facial expressions, and fingerspelling, particularly for contact with people who are deaf-mute. Through their syntax and vocabulary, sign languages are fully-fledged natural languages. Furthermore, Mr. Sacks once explained in his word that true sign languages are complete in themselves with their syntax, grammar, and semantics. At the same time, they are distinct from any written language or spoken language.[3]

Sign language is primarily used for the deaf and individuals who can hear but cannot speak. However, some hearing people still use it, most notably deaf and interpreter's relatives and friends who allow the deaf and broader group to engage with one another. There are numerous stereotypes and theories concerning deaf sign language. Many people assume that signing is merely a manual interpretation of the language spoken, which is wrong. At present, nothing in common between the spoken language and the language of the deaf. Sign language is as complex as the spoken language, but they are distinct.

The deaf community and sign language have a co-existing relationship. In several countries, with an increase in the sign usage base, sign languages have been standardized.

2.1 The Origin of Sign Language

As illustrated by Socrates' comments in Plato's Cratylus, we gathered the knowledge that at least from the 4th century BC, the deaf used sign language. He explained in his dramatic words that if there had no speech group, then communication is not possible. In that case, people must use signs just like dumb and deaf. In order to express speech in signs, different hand movements can mean softness, heaviness. Furthermore, he explained how we could communicate just like dumb and deaf with signs where we can use different parts of the body and hands when we do not have a tongue or voice. The proof of visual contact is found in the Roman coin named tessera. In tessera, numbers are represented by fingers, which is similar to today's fingerspelling in sign language.[11] In the following fifth to sixth centuries, Saint Bede explained the hand alphabet representing a visual communication mechanism. A group of religious people used that system as they took a vow of silence.[17]

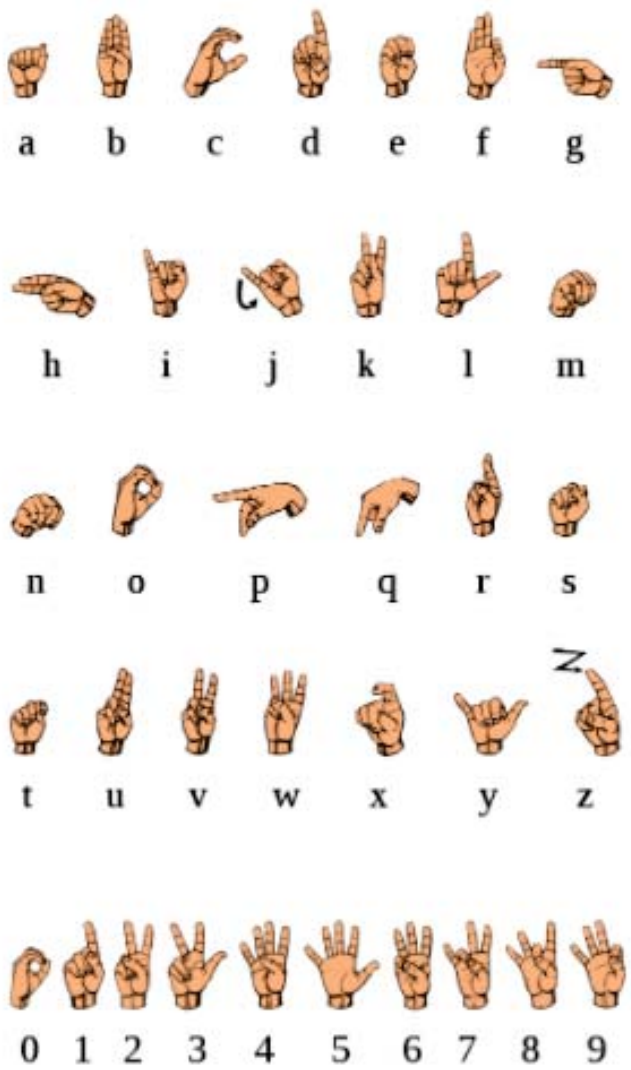


Figure 2.1: Dataset of ASL

Melchor deYedra's 1593 Refugium Infirmorum mentioned that the finger alphabet was developed in the 13th century by Saint Bonaventure.[11] To educate mete-deaf people in communication through gestures in 1620, "Reduction of letters and art for teaching mute people to speak" was published by Juan Pablo Bonnet, a priest. Though Signs had existed for many years in deaf communities, in the 17th century, preliminary studies dedicated to signing languages were found in the Western world.[40] In the 18th century, Abbé Charles-Michel de l-Épée used the morphology of signs developed by Bonet to construct a fingerspelling alphabet.[11] Charles Michael De L" Eppe established the first school for the deaf in Paris, which was free and was open to the public. He was considered the "Father of Sign Language and Deaf Education." [38] In 1817, sign language started migrating and evolving. Thomas Gallaudet and Clerc founded the American Asylum for the Deaf in Hartford. This school's popularity eventually led to others' building and the recruiting of young educators to teach the previously uneducated deaf.

Sacks mentioned how Clerc and Gallaudet had an instantaneous influence on the sign language. Because till then, the American teachers did not think of an ar-

ticulate or competent deaf-mute communication system. People only used manual alphabets until the 19th century to sign a language. It was just the transition from oral to sign language of words. It became apparent in the 1950s that there was a need for higher education. Edward Gallaudet, the uncle of Thomas Gallaudet, was appointed director of the Columbia Institution for the Deaf and Dumb and Blind Instruction in 1957. In 1964, the school received federal support and ultimately became known as Gallaudet College, the country's first college for the deaf.[11][2]

The advancement in developing a signed language for the deaf is really significant because the speechless person expresses what he feels to others and what he believes. Speech is a part of the thought. For the first time, the development of signs became a mechanism of allowing access to intellect.

2.2 Sign Language Classification

Sign Language can be classified into three groups-

1. Deaf sign languages include village sign languages, shared by the hearing community. It also includes deaf-community sign languages, which are the preferred languages of deaf people worldwide.

2. Auxiliary sign languages are not natural languages but sign systems with various variations, used with spoken phrases. Basic gestures are not used, since they do not constitute expression.

3. Links between signed and spoken languages include signed forms of spoken languages, often referred to as manually coded languages.

2.2.1 Sign Languages in Deaf Culture

Many people do not see the deaf as having a community, but they do, and they have their laws, protocols, and recognition rules. Deaf people have a culture. The culture of hearing people and deaf people are similar, but there are significant variations. Deaf people have various perspectives than speaking to individuals. It makes their society distinctive. Discussing deaf culture, O'Banion explained how deaf culture creates an opportunity for deaf people to grow themselves. A person is not only confined in physical abilities but also mental abilities. In terms of that platform, deaf culture plays an important role. To communicate, they need Sign language. Deaf individuals can create a cultural and social identity for themselves through the use of sign language. They will spontaneously interact with each other. A shared sign language helps them to keep their deaf community together.

2.2.2 Deaf Sign Language

There are quite prevalent misinterpretations that sign language is universal, but they are wildly inaccurate. Wherever substantial groups of deaf people are together, sign languages have evolved. There are several distinctive sign languages. The growing deaf community may construct a sign language of its own. In each region,

there is usually a unique sign language. Till present, more than 137 exceptional sign languages linguists have identified. Some of the well-known sign languages are American Sign Language (ASL), British Sign Language (BSL), Auslan Sign Language, Indian Sign Language, Japanese Sign Language (NS or JSL), Mexican Sign Language (LSM), New Zealand Sign Language (NZSL), Quebec Sign Language (LSQ), Turkish Sign Language (TSL), Chinese Sign Language, Bangladeshi Sign Language(BdSL). Since the time of the first deaf-mute, signed language has most likely been around as a simple form of communication, but it was not until far later in the 1800s that sign techniques were established that aligned with the area's grammar and language.

2.2.3 BSL (British Sign Language)

British Sign Language or BSL, is the type of sign language that is most common in Britain. When deaf people came together to form communities throughout the world, BSL began to evolve since then. The language has a grammar structure of its own, but it is not constructed in the same way as English. This mostly alters the word order of phrases - you begin with the BSL subject and then say something about the subject after that. In English, for instance, we might say,' What is your name? 'But in BSL, we would indicate name, what? '. BSL has vocabulary and grammar that are distinct since it evolved separately from English. A single sign can be applied, for instance, to indicate "I have not seen you in ages." What is probably much less well known is that the bulk of the UK's deaf community's sign language, British Sign Language (BSL), also differs from one region of the world to another. Research has revealed that deaf people from different parts of the UK use separate regional signs for the same meanings.

2.2.4 CSL (Chinese Sign Language)

Chinese Sign Language is termed as CSL. While Chinese sign language standardization is reasonably recent, the country is diligent in making education in sign language more available. Chinese Sign Language (CSL) has been evolving mainly since the late 1950s, and its symbols are like written Chinese characters. CSL has various dialects, with the most common being in Shanghai. One is a verbally similar negative handshape to the fingerspelled letter I in American Sign Language (ASL). A horizontal handwave and a side-to-side headshake both tend to have equal negative power, but they should not be used at the same time. The systems of negative terms and phrases indicate that CSL has a unique grammatical system that causes us to reconsider some of our sign language contradiction assumptions.

2.2.5 ISL (Indian Sign Language)

Over the last 100 years, sign language has developed in India, and just now, the government agreed to institutionalize it in the form of a dictionary. Dictionaries of British and American sign languages are a significant reference point for the dictionary. Still, because of the linguistic, geographical, and historical context, the signs used by deaf people in India are quite different. Indian sign language is quite scientific and has a grammar of its own. Still, a lack of awareness has ensured that

many deaf people are not even conscious of institutions to study it and prepare themselves for interaction. India has five million deaf and hearing-impaired persons, according to the new report. Yet the nation has only about 700 schools teaching sign language. And it is not written, unlike English or Hindi. All around the world, basic signs used to signify familiar words such as "marriage" or "tea" are not the same. Although the "wearing a ring" activity indicates "marriage" in the US, it is expressed by the "holding of hands" in India.

2.3 Grammar and Tense in Sign Language

Like all other spoken languages, sign languages have grammar, syntax, vocabulary, structure, and terminology. Sign languages are not just something that anyone acts. Without specific rules, they are not movements chained together. Sign languages are the natural language that must be learned, not merely a series of basic activities to communicate that somebody would dream up. The primary distinction between spoken languages and sign languages is that, while spoken languages use sounds to form sentences; sign languages use body and hand gestures to construct signs. Each sign in sign language represents a meaningful word in spoken language.

Grammar is used in both spoken and sign language to form sentences from words and gestures, respectively. That is what makes them languages other than just mimes or simple gestures. Signs are made up of different hand gestures, distance from hand to face simultaneously hand to the body, facial expressions, indications to any object, and body gestures. In various combinations, these components are used to construct signs in the language. In some sign languages, one hand is used to show some sign; both hands are used in others. Sometimes movements of the body corresponding with the mouth make a sign valid. Body gestures are significant in grammar. For instance, raised eyebrows to signify a question in several sign languages with a "yes" or "no" answer. Even in the same language, American Sign Language is different from the spoken English language. Sign languages share some similar terms with spoken language occasionally, but they are typically very distinct. The grammar rules are identical to each other for various sign languages, but they are not the same. For sign languages, there is not standardized grammar; each sign language has a sentence structure. For instance, sign languages use different hand forms to make signs; each sign language seems to have its own collection of hand shapes.

Tense is a vital part of grammar Tense expresses time reference. There are certain specific words or some form of verbs that indicate the tense in every spoken language. In Sign Language, tense acts differently than spoken language. If a sentence means that it happened in the past, when saying the word "fish," a simplified version of "finish," the sign stops at chest height, either the beginning or end of the term. In this way, it signals that everything has already happened. ASL offers knowledge about time with a particular collection of signs produced concerning time. What is regarded as a timeline as seen in Figure 2.2. There is a relative position in the time signs and agrees with their sense on the timeline. The height determines, sign location co-ordinate scheme, in addition to the timeline. Height separates tense markers in some situations, Adverbials from time.

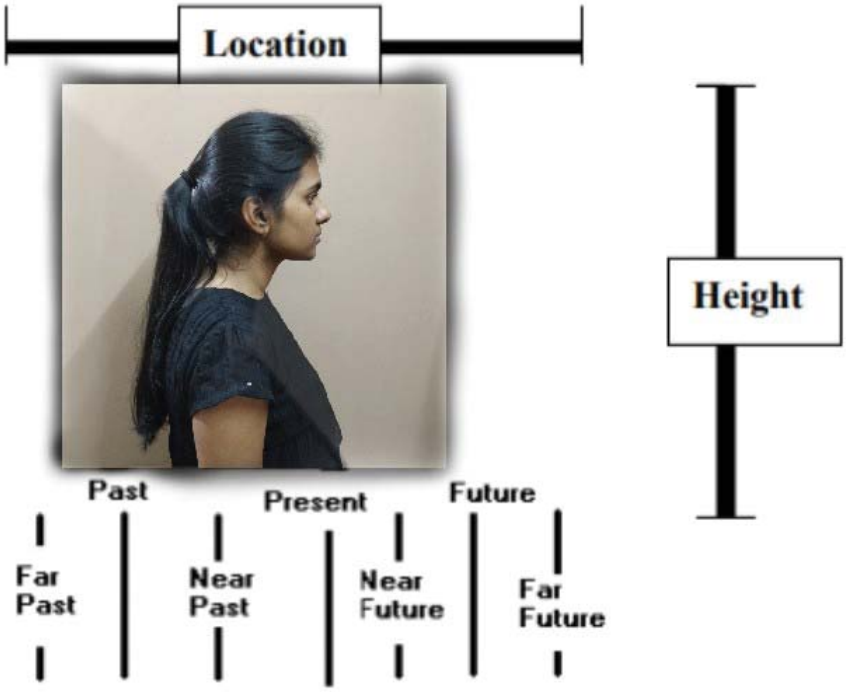


Figure 2.2: Timeline

In the above figure, the timeline is shown as starting with the vertical lines from left to right from far past, past, near past, present, near future, and far future respectively.

2.4 Sign Language components

The first linguist to understand that signs are not completely unanalyzed was Stokoe (1960).Phonems are introduced by Stroke.Firstly he categorized signs and named them"cheremes" which eventually was accepted as phonems.In addition,he figured out the difference between spoken and sign language.Earlier phonems were sequential whereas, they seem to be simultanious in latter. His phonemes were divided into three categories: active hand forms (what moves), location (on the face, body, or other hand), and motion. Later, orientation was introduced as a fourth phoneme category (the way hands point or face or connect with each other) (Battison, 1978).[15] However, signals are not limited to manual activities, only hands and weapons, for instance.Non-manual features are also essential criteria used to express a sign's sense, such as head orientation, head inclination, body posture, eye movements, and mouth shapes.[23]Facial features are, in truth, essential to express emotions. To distinguish between questions, negations and affirmations, some of these features are also crucial.

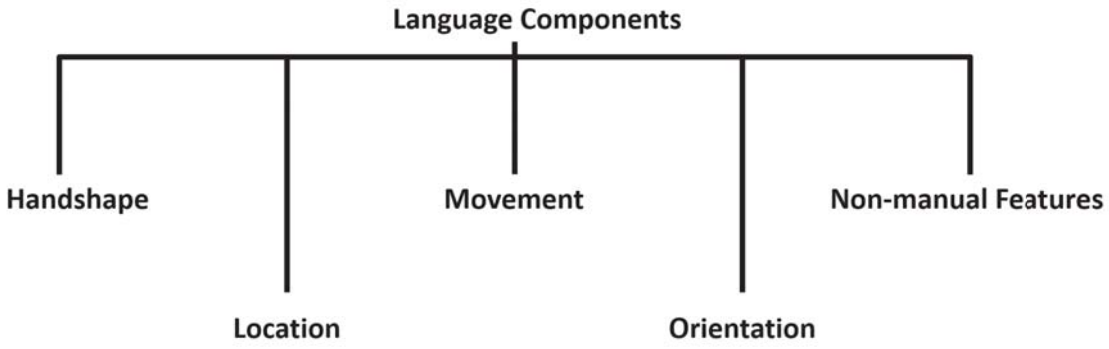


Figure 2.3: 5 types of language components.

2.4.1 Finger-spelling

Every Alphabet in each sign language requires finger-spelling. Finger-spelling is a tool. The most common use of finger-spelling is for names and numbers. Sometimes signer may use finger-spelling for any word to avoid confusion. Finger-spelling is not universal at all. In various sign languages, it's distinct. Different sign languages can have an indistinguishable finger-spelling system. Or, there might be very different finger-spelling schemes for them. Some languages uses one hand for finger-spelling others uses both hands to perform different alphabet or numbers.

2.4.2 Gestures

Gestures are an essential factor in Sign language. A movement of the hands, arms, or head to express an idea or feeling is called gestures. Gestures are used to describe things, greet and farewell, draw attention, show agreement or disagreement, give directions, and express thoughts and ideas. It expresses emotions. In every sign language, hand gestures, facial expressions, body gestures are part of the language. Hand movement is needed to form any word or sentence in sign language. Even to express emotions like sad, happy, joy, excited facial expression with mouth gesture is required. Some words can be expressed through one hand in one sign language, but the same word can require both hands in another language. In terms of forming sentences gestures play a vital role. Sentence structures are different in spoken language and sign language. There are no verb forms or ing, es which can indicates the tense in a sentence in sign language. To express past, future or present different type of body gestures are used in different sign languages. Shape of hand, location, movement, primes, faces all these are needed in sign language. The shape can vary in terms of the fingers used, whether the fingers are stretched or curved, and the hand shapes in general (s). In BSL "When signing THANK YOU, the position of the hand is "palm up" rather than "palm down". The hand may be directed in a variety of other forms in other gestures, such as the "flat hand, palm towards signer" shape used to signify MINE. In every word hand gesture is essential in sign language.

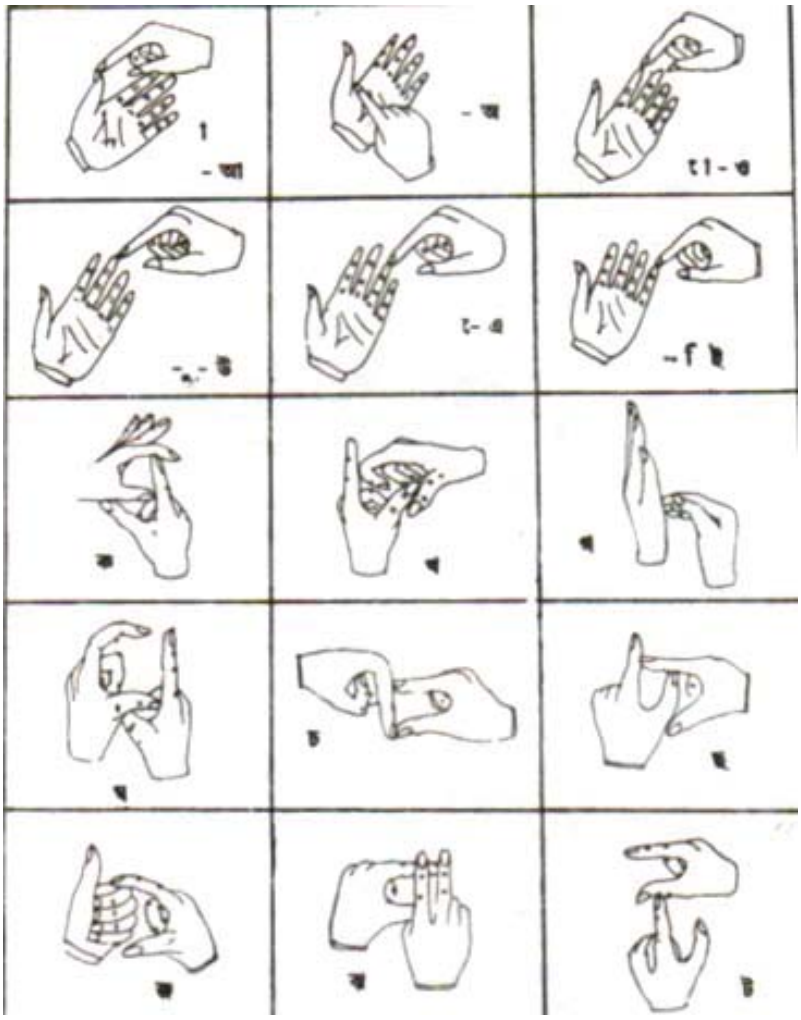


Figure 2.4: Picture of alphabets in BdSL.

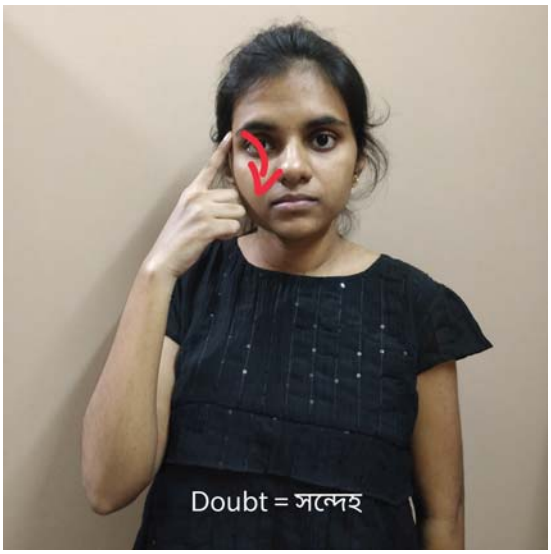
2.5 Bangladeshi Sign Language (BdSL)

Bangladeshi Sign Language (BdSL) is a widely used mode of communication in Bangladesh for people with hearing impairments. About 2.4 million people of Bangladesh are reportedly Deaf and Hearing Impaired.[34] Bangladeshi Sign Language (BdSL) has begun the official journey not so long ago. The Centre of Disability in Growth (CDD) has been working on this ever since the 2000s. In order to enhance their ability to communicate and boost their self-confidence, CDD specializes in communicating with people with disabilities. CDD started to disseminate awareness and began incorporating multiple coping techniques to explain the disability. They eliminated the barriers of speech to enable persons with disabilities to communicate themselves. They have developed Bangladeshi Sign Language. Only a simplified version of ASL, BSL, Auslan, and some Aboriginal singing languages were available before 2001. The CDD now has a complete sign language collection. CDD has written several books rich in grammatical and lexical laws, and in their training center, they also provide sign language training.

The largest group of the language-based minority groups in Bangladesh is the community of Bangladeshi Sign language users. The Bengali language is the fifth most

(a) "Bhanga" (break).

(b) "Chhera" (tear).



(c) "Sondeho" (doubt).

(d) "Boka" (fool).

Figure 2.5: Similar gestures for different signs in Bangladeshi Sign Language.

commonly spoken language in terms of population in the writing system. There are eleven vowels named "sôrôbôrnô" and 36 consonants, called bænjônbôrnô. There are different signs for all these alphabets and for meaningful words, too.

Signs can be represented in various ways. Some signs require one hand, while other signs require both hands. Usually, the right hand is used to convey all the signs as it is the dominant hand. If the signer is left-handed, then the left hand is used for the signs. Another essential factor to consider is the fact that signs can be static and dynamic. While delivering dynamic signs, one hand can be in motion, and the other can be static. In such cases, the generally dominant hand is in motion. Moreover, if there is any sign that requires both hands simultaneously, we need to keep in mind that both hand shapes and movement need to be the same simultaneously. Otherwise the sign is not correct. Furthermore, if the sign requires both hands, but only one hand is in motion, the other can differ.

Figure 2.6: Signing Space

Another vital factor sign language to remember is that it is possible to use the same hand forms or gestures to convey different signals. For example: In Bangladeshi sign language, "break" 2.5a and "tear" Figure 2.5b have similar hand position, but the hand form for each sign is different. Comparably, using the same hand shape and palm position, "doubt" 2.5c, and "foolish" 2.5d are signed but in those hand movement is different.

Lastly Sign location which means signing space is significant feature to consider. Signing space means the region usually used to express all the signs which includes head area, the middle part of body and both arms.Signing space is represented in figure 2.6
All the modules must be appropriately used to grasp a sign fully. Moreover, to correctly communicate a sign, there are specific laws that need to be followed. Any of these components and regulations can be universal, while others are entirely local to a particular civilization, community, or region.

There is only one high school for deaf children in Bangladesh named Dhaka Bodhir High School other than CDD. People are still ignorant of this in Bangladesh. Modes of communication will still not lead an uncomplicated life here and hence the deaf children. Steps are being taken to make the people of Bangladeshi Sign Language conscious, though, and we should hope that life will finally get even more straightforward for the hearing and speech impaired in the future.[24]

2.6 General Supervised Algorithms

Supervised algorithm is a type of algorithm which requires both input and output in order to deliver a result or comparison. The main task is to match both input and its desired output. The algorithm usually gets trained by inputs so that later it can label a new and unknown input [1]. Later it gets divided into regression and

classification.

2.6.1 K-Neighbors Classification

One of several simplest machine learning algorithms based on the Supervised Learning method is K-Nearest Neighbour. The algorithm assumes that similar things exist in close proximity. It is a non-parametric as well as a lazy learning algorithm. This algorithm does not make any assumption on the underlying data. Unlike most of the machine learning models, it does not learn from the training set immediately. While training, it stores the dataset. Finally, at the time of classification, it performs an action on the dataset. The number of neighbors is the most important parameter that decides the performance of the model. The number of nearest neighbors is generally denoted by K. Odd numbers as the value for K tends to give us better results. To find the closest similar points, we find the distance between the points using distance measures such as Euclidean distance, Manhattan distance, Minkowski distance and Hamming distance. K-Neighbors Classifier basically follows the below steps:

- calculating distance
- finding closest neighbors
- voting for labels

2.6.2 Naive Bayes Classifier

Naive Bayes classifier is a type of probabilistic machine learning model based on the Bayes Theorem, used in classification tasks. For each sample of data it generates conditional posterior probabilities. Due to its simplicity and linear run-time, the Naive Bayes classifier is identified as a common learning algorithm for data mining. applications. It not only handles a large number of variables and large data sets but also handles both discrete and continuous variables of attributes. The naive Bayes classifier refers to learning tasks where a combination of attribute values defines each instance x and where the target function f(x) will assume any value from the same finite set V. It produces two probabilities in case of binomial classification.

2.6.3 Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning algorithm. It can be used for both classification and regression problems. For our research, we have also used SVM for classification. To carry out the SVM algorithm, every data piece is plotted as a point in an n-dimensional space where n is the number of features. In addition, the value of a particular coordinate takes the value of a feature in our data [47]. The goal of the SVM algorithm is finding a hyperplane that can distinctly partition the classes.

The hyperplane is known as the decision boundary. The hyperplane helps us in differentiating between the data points, the two sides of the hyperplane represent different classes and data points that belong to either side can be attributed to those classes. The number of features determine the dimension of the hyperplane.

For example, if the number of features is 2 and 3, the hyperplane is a straight line and two-dimensional plane respectively. There are several possible hyperplanes that can be plotted for separation between the two classes of data points but our goal is to find such a plane that has the maximum margin. Maximum margin means the distance between the data points of both classes must be the maximum. This is of utmost importance because by maximizing the margin distance, we are creating reinforcement for when newer data points are to be classified so that they can be done with more reliance.Data points that are closer to the hyperplane or decision boundary are called Support Vectors. Both the position and orientation of the hyperplane are affected by these Support Vectors. These points are what allows us to build our SVM model. We can maximize the margin of the classifier using such Support Vectors. Even deleting a point will have an effect on the hyperplane. Grid search technique is used to optimize the accuracy of a classification model. But it provides a large computational problem. Upon GridseacrCV, we found that an optimum value for the parameter "c" was 1000 and not the default value to 100. Inputting the value of c to be 1000 increased the accuracy of the classification model but provided a computation backlog.

2.6.4 LSTM based RNN

LSTM based RNN is introduced to overcome the problem of long term dependencies. Typically all Recurrent Neural Network is designed to remember previous steps to determine the next steps. It is supposed to do that. But in real life, RNN's usage shows otherwise. Most of the time, Basic Recurrent Neural Network performs poorly. It becomes incompetent to remember steps that have a distant relationship with the current state. For example: in language translation, if RNN process a sentence "Color of the ocean is, "then RNN is supposed to put "blue" at the end. In order to do that, RNN needs to remember the word "ocean" and "color." But in real life, RNN struggles to do that. Hochreiter and Bengio et al. identified the problem. They identified it as a Vanishing gradient problem. In a Recurrent Neural Network, output depends on one of the inputs. In the initial state, when backpropagation takes place, then the weight values become very small. Even sometimes the weights don't update properly. It alters output. This Vanishing Gradient causes the problem of Long term dependencies.

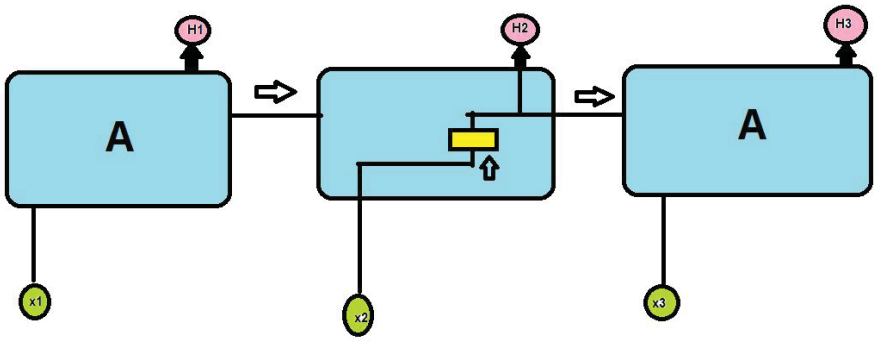


Figure 2.7: Recurrent Neural Network

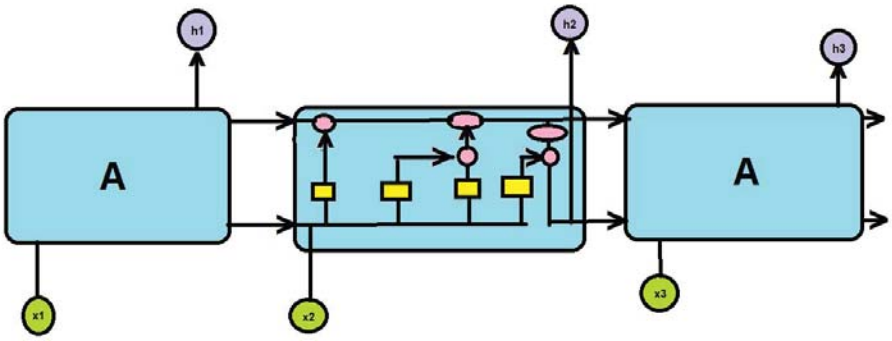


Figure 2.8: LSTM based RNN

Now a day almost everybody is getting success by using RNN. LSTM based RNN is more preferable, but basic LSTM based RNN is hardly used in any process. All LSTM based RNN based process uses a slightly different approach to achieve the goal. As a result, the algorithm changes all the time.

Basic Recurrent Neural Network has a simple chain of repeating modules [33] ; Figure 2.7. But in LSTM based RNN , here are four interacting modules; Figure 2.8 .

LSTM based RNN has some key elements inside of its architecture.

- Cell State
- Gates
- Sigmoid function.

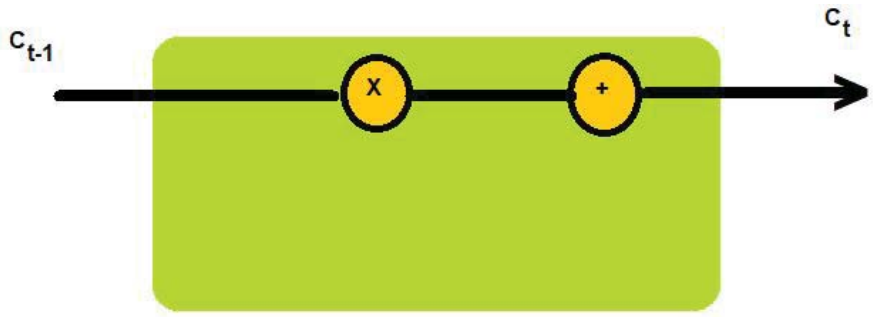


Figure 2.9: cell state

LSTM based RNN has a cell state, [33], figure 2.9. This state is like a conveyor belt. It can be easily manipulated by gates.LSTM based RNN is carefully controlled by frameworks called gates to delete or add cell state data. LSTM based RNN has in total of three gates. Gates are an optional means of transmitting the information. They consist of a neural sigmoid net layer and an operation of pointwise multiplication. It delivers a value between one and zero. The value one means “let the data through” and zero means “let not the data through.”

Working procedure

The first step in figure 2.10 is to identify which details the cell state will through away. The process is done by a sigmoid layer known as the "forgotten gate layer." It looks like. It delivers a number between 0 and 1. It means "keep the data," and 0 means "through away the data."[33]

$$f_t = \sigma\left(W_f \cdot (h_{t-1}, x_t) + b_f\right) \tag{2.1}$$

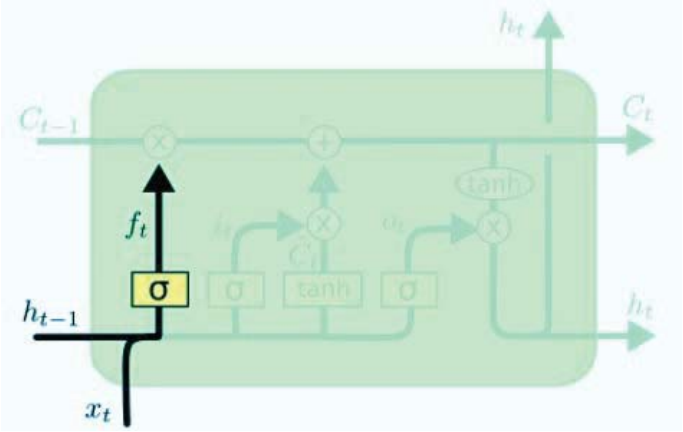


Figure 2.10: Forgotten gate layer [33]

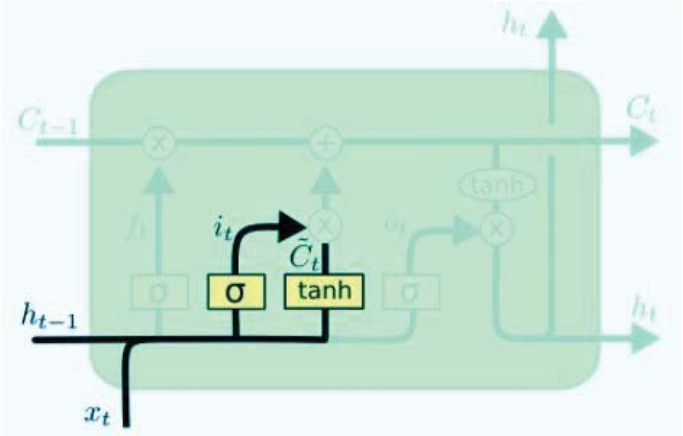


Figure 2.11: Storing data [33]

$$o_t = \sigma\left(W_0\left[h_{t-1}, x_t\right] + b_o\right) \tag{2.2}$$

$$_t = o_t * \tanh\left(C_t\right) \tag{2.3}$$

After that, another gate activates and does its part in Figure2.11. It decides which information the RNN is going to store in the cell state. This process has two-part.

A sigmoid layer comes to action [33]; it is called the "input gate layer." It identifies which values RNN is going to update. Then in the next process tanh layer comes to action. It makes a vector of a candidate value, later o these vector values will

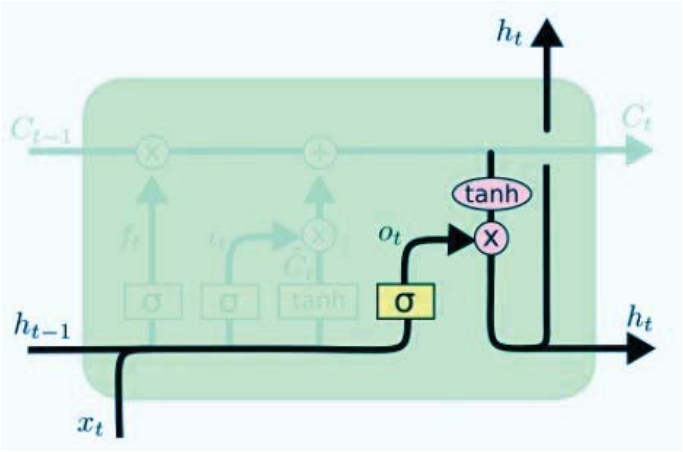


Figure 2.12: Result generating layer [33]

be stored in the cell state. In the next state, RNN will combine these two layers to update the cell state.

Now old cell state gets updated into the new cell state; Here, the forget gate layer is multiplied by the old state. Then the result will be added with the multiplication of the input gate layer and candidate values layer. As shown in Figure 2.13, this is the new candidate value, which is scaled according to how much we can update each cell state value.[33]

$$C_t = f_t * C_{t-1} + i_t * \bar{C}_t \tag{2.4}$$

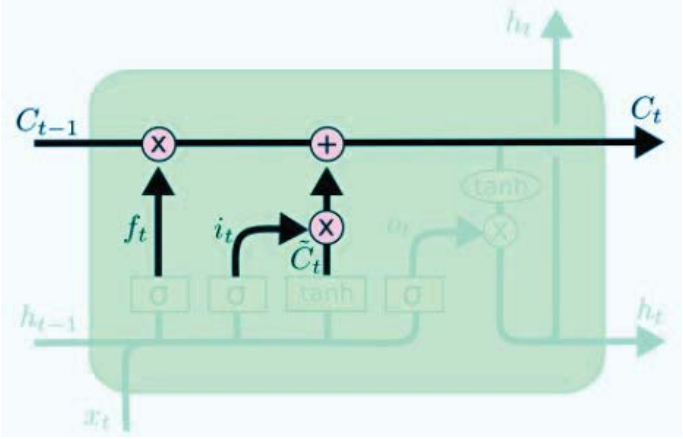


Figure 2.13: Candidate value-generating layer [33]

In the end, in Figure 2.12 we will get an output. The output will be according to the cell state, but it will be highly filtered. As we have overviewed before, the first RNN will decide which date it is going to keep running the sigmoid function. Then the cell state goes through tanh layer [33], and then it gets multiplied by the output of the sigmoid state. As a result, we will get the work that we desired to get.[33]

2.7 OpenPose

2.7.1 Joint Extraction

OpenPose takes RGB images as input and produces 2-dimensional anatomical key points [42] for each image's human body. In Figure 2.14, the 1st stage in 2-benched CNN predicts confidence maps, and the 2nd stage predicts the affinity field. A 2D vector calculates and encodes the position and placement of all the body parts of each person in the image.

Again both the affinity field and the confidence map are being parsed my greedy inference in order to present 2D key points of all human body in the image. OpenPose usually identifies 25 key body joints in the human body. Later these body joints can be used for gesture recognition and activity recognition.

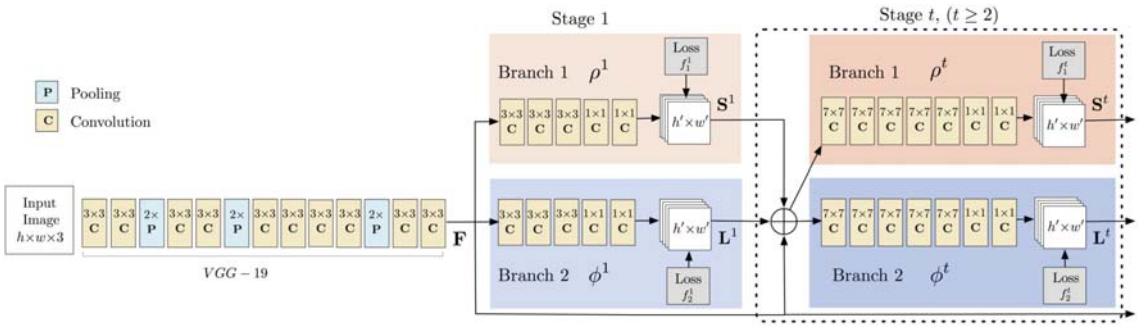


Figure 2.14: OpenPose Architecture [42]

The architecture takes images as input and produces array-like matrices confidence maps and part affinity heat maps.

2.7.2 The Stages

1. Stage 0: In the 0 stages from Figure 2.14, VGGNet's first ten layers come together to reduce feature maps for the input pictures.

2. Stage 1: The architecture uses 2-branch multi-stage CNN - A confidence map predicts the highest possible position where it more likely the possibility of presence of a body part. Confidence map(S) is a grayscale image where the values are high where there is a possibility of a body limb. The second branch predicts a number of Part Affinities (PAF) 2D vector fields (L), which encode the extent of the connection between parts (keypoints). PAF matrices are the 20th to 57th patterns. [42]

3. Stage 2: The affinity maps and confidence maps are then loaded by greedy inference in order to create two-dimensional key points for all the human subject present in the image.

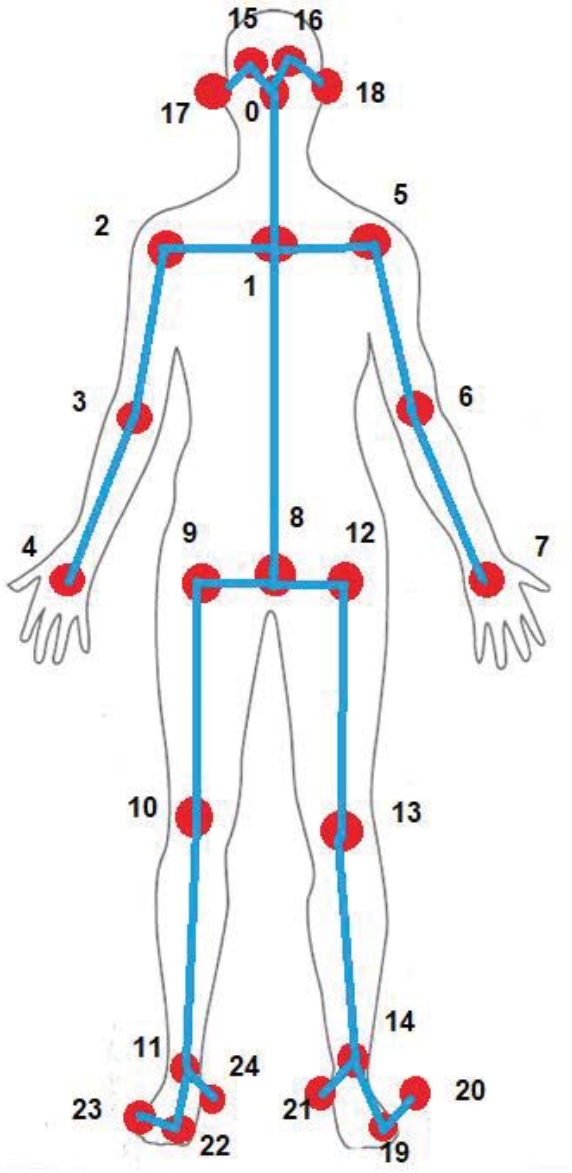


Figure 2.15: Body_25 keypoints.

2.7.3 Human Body Pose Estimation

Currently, open pose architecture is available in three different models. 1st one is trained based on a dataset that is Multi-Person Dataset(MPII). It produces 15 key points on a human image [42]. 2nd model was trained based on the COCO dataset. It produces 18 key points on a single human image [42]. The 2016 COCO key points challenge was won by OpenPose architecture. And finally, there in body 25 architecture. We used this architecture in our work in Figure 2.15

Then there is face key point estimation. It produces 70 key points from a person's head. There is a model for hand key point estimation. It produces 21 key points from each hand [42], in total, 42 key points from two hands.

Using all these key points, OpenPose creates a skeleton, which is only the visualization of a frame of all key points and their connections. In order to use this key point as inputs of a model, inputs open pose also provide a JASON file as output. It can be used for both training and testing datasets for an estimation or recognition model.

Chapter 3

Related Work

This chapter elaborates on other papers and studies that are directly or indirectly related to our research topic. It also includes the overall limitations of the previous studies, which led us to choose our topic.

Sign language has been around for a very long time. Before the invention of verbal language, humans used sign language. Nowadays, mostly physically challenged people like dumb and deaf use sign language for communication. We are in an era of modern computation. We came so far in order to make "Sign Language Recognition" more user-friendly and advanced.

Research based on sign language recognition started a long time ago. In the late 20th century, works on sign language detection became popular among research society. In a paper [4], Uras et al. (1994) implemented the Size function in order to recognize Alphabets. Size functions are integer-valued functions that reflect the visual shape's both qualitative and quantitative properties. All individual signs were delivered as a means of a feature vector. In the end, they used the k-nearest-neighbor rule so that it could classify feature vectors for unknown signs. Later on, in the next year [6], Yamaguchi et al. (1995) generated a process where they recognized Japanese sign language by generating associative inference from the information of a video. Their system required two types of associative memory features- associative robustness Associative memory combination. They implemented 16 words to test the system.

In the same year, [9], Starner et al. (1995) proposed a method using Hidden Markov Model that can recognize American sign language. They followed a simple system structure and got an accuracy of 99.2% without modifying anything. Their model could recognize sign language till sentence level. Again Huang et al. (1995) [5] proposed three systems for recognizing sign language: tracking hand, extraction of features, and recognize gestures using a 3-D Hopfield neural network. Hopfield has an overall identification precision of 87% after many matches. However, they have been misidentified because the difference between the two datasets' shape and gesture is minimal, and the trajectories are identical. Later in 1996, Kim et al. [7] implemented a system where they made a method that can recognize a sign and modify it into the Korean language. In order to complete the process, they used a pair of data-gloves to detect motion of hand and finger. They used a fuzzy min-max neural network for further betterment, which was used for online pattern recogni-

tion. It was an effective method, but its portability and usability were in question. Using an additional device to detect hand gestures is not practical. The system was unique at that time, but its implementation was limited.

In 1996, Yoshino et al. (1996) [8] introduced the world with a unique system. They applied color combinations on image sequences and recognized sign language. Japanese by far was way ahead in developing a sign language recognition system. Yoshino and his team were not different from that. According to their system, a person needs to wear colorful hand gloves. Then the system would estimate a word with a motion by matching color patches. In their system, they used three parameters to estimate the Japanese finger alphabet visible patch. They were a color combination, dispersion, and hand direction. The authors used all those methods to identify all finger alphabets and words. Just like the previous work, another system used data-gloves. Liang et al. (1998) [10] created another method to recognize gestures. However, this time, they made it real-time and continuous. Their method requires wearing gloves, which would collect data and would feed the data to the Hidden Markov Model. They worked with four parameters- posture, position, orientation, and motion. They got decent accuracy, approximately 80.4% in their system. It was immediately after that year, Shin et al. (1999) [12] worked on a fascinating topic. They implemented a Multilayer Perceptron classifier to recognize Korean sign language. Till now, Most of the research is performed on sign language recognition took place was two-dimensional. Nevertheless, from Shin's work, a new opportunity opens for the tech world. This MLP classifier based model could work for higher dimensions. This MLP classifier uses Parameter-free species Genetic Algorithm is a combination of both Species Genetic Algorithm and Parameter-free Genetic Algorithm.

By this time, recognition of sign language shifted towards the neural networks. Wang et al. (2000) [13] created a method to feed ANN and k-mean with the segment of data steam. The system would collect the segment from phonemes. Phonemes are more efficient than using the basic unit of a sign. Next year (Jiangqin Wu et al . 2001) [14] deals with the detection of CSL. The features of CSL sign language are based on the proposed greedy time axis(GLATA) clustering algorithm for the segment sign word, GLATA based training algorithm is proposed to train a template for each word, and a GLATA based recognizing algorithm to recognize 227 words chosen randomly from CSL at 96% accuracy [14]. Hamada et al. in 2004 [16] introduced the hand shape estimation method under difficult backgrounds. By using a shape transition network, they minimized matching candidate models. They also inserted the models without contours and tracked the models properly in the transition network to prevent the subsequent errors induced by the blurred hand picture during hand movement. To choose the best-suited model, they proposed the match criterion defined as the portion's length, which covers the accurate hand contour, considering the probability of the edge in the background. Later, In [18], an OP/Viterbi algorithm in real-time to Continuous recognition of sign language is designed to be Implemented by Yao et al. in 2006. In the first classifying stage of OP searching, they use the center matching method, and transition factors [18]. The experimental results show that the continuous sign language may have a different recognition increase in speed with a broad vocabulary with 4942 signs and 543

words, which will satisfy the real-time recognition system [18]. Two years later, in [20], Wang et al. in 2008 presented the Sign Language Recognition multilayer architecture to recognize signer-independent CSL, where Under an initiative framework, classical DTW and HMM are implemented. They define the confusion sets in the two-stage hierarchy and apply the DTW/ISODATA algorithm as the solution in the vocabulary space to build confusion sets. The experiments show that the Sign Language Recognition multilayer design improves total detection time by 94.2% with precision, which is 4.66% higher than the HMM-based recognition process [18].

Though sign language recognition systems were present in the tech world from the late '90s, works on Bangla sign language recognition came to light, not before 2008. Science then many works have taken place to attempt to provide a recognition system. We came across an early work by Begum et al. (2009) [21] through our research. In their works, they wanted to make Bangla sign language recognition more simple. They used PCA (Principal Component Analysis) based method that works by matching patterns. They used a CCD camera in order to capture video to process. They got approximately 82.5% accuracy for 6 Bangla vowels. In this paper [22], in 2010, S.J Wang et al. are using a tensor sub-space analysis for a multi-view hand to classify 26 alphabetical manual letters. Every hand is taken from five different points of view in our experiment. On gray images and binary images, respectively, two experiments are performed. The findings indicate a strong multi-view performance of the proposed system. In 2012 [24], the Kinect Depth Camera was used by Choudhury et al. The built-in features of the SimpleOpenNI library, and part of the OpenNI framework was used with the Kinect Depth Camera. Extraction of seven features is included. Ten samples were for each sign and two inexperienced signatories who learned these signs for this particular research. Finally, Ten 70 x 1 matrix were used to generate one 70 x 10 matrix. If separate signs are presented as input, the machine is 88% accurate. However, hand-shaped characteristics must be extracted, or it would be impossible to distinguish signs of identical movement and positioning of the hands using the current features. Using skin color segmentation in [25], [29], the human body's skin parts were defined from the frames. Here, (Kaushik et al. 2012) [25] For refining the color model Red-Green-Blue (RGB), heuristically choose the threshold value for candidate area recognition (Hand and wrist band sign regions). Apply color segmentation uses two distinct color band areas and filtering. A variation of the pixel position and the priority-based information is used to remove incorrect pixels detected. A statistical model matching technique is used to classify the hand sign areas. They were 97.5% effective in the identification and 96% in recognition.

Two years later, to detect the hand in each frame (Rahaman et al. 2014) [29] uses the HAAR-Cascade classifier and removes the hand symbol based on Hue and Saturation values, which correlate with human skin color. The binary images are then categorized compared to pre-trained binary images using the K-Nearest Neighbors (KNN) classifier. This system will classify 6 Vowels of Bengal and 30 Consonants of Bengal and 98.17% of Vowels and 94.75% of Consonants are correct [29]. This year Sarwant et al. have identified hand sign characteristics from the Indian Sign Language from images and used the Principal Component Analysis (PCA) algorithm to interpret gestures and translate them into texts [30]. For the separation of hands

from the bottom, segmentation and morphological filters were used. 260 pictures of 26 hand signals were used, and the outputs are shown as text. In one of the papers of 2014 [27], Hussain et al. Proposed detailed estimates of the palm and fingertip posture by hand contour. The forearm may also be the outline, and the device has strong manual titration and rotational tolerances. They took PCA to differentiate similar gestures and an optical flow algorithm to minimize the effect of movement epenthesis; they tested the system against certain ASL hand gestures. Also, In [28], (Mohandes et al. 2014) used the Arabic sign language recognition with Leap Motion controller. Ten samples are collected with a leap step sensor of every 28 characters. They chose the 12 most appropriate features for each data frame and further process 23 features returned by the Leap Motion Controller. The Naive Bayes and the Multilayer Perceptron were used to classify 28 letters in the Arabian sign language. Using Multilayer Perceptron, a correct identification rate of 99.1% and using Naive Bayes it is 98.3%.

Next year in 2015, Hasan et al. [31] implemented LDA and PCA to extract features and reduce dimensions. For both the Bangla and American sign languages, the same database was used. The method has defined sixteen daily works and 10 Bangla numbers. Their primary approach works using skin detection and extraction of features. In this BdSL translator, eigenvectors, fisher-vectors, SVM are all used. Subsequently, this year Jarman et al. [32], based on the Fingertip Finder Algorithm, The vowels, consonants, and numerals are derived from Bangladeshi sign language recognition system capable of recognizing 46 Bangladesh sign languages types. The image input was changed in size before being translated into a binary format using the Otsu thresholding process. Then morphological operations such as a 9x9 Median Filtering and cleaning was performed on the converted binary image. To distinguish the sign digits, 11 kinds of features were extracted. For the purpose of the system, a feed-forward Neural Network was used, resulting in 88.69 percent effectiveness in recognizing the BdSL sign digits. Also, this year [34], LSVM is proposed by Yasir et al. for recognizing Bangla signs and alphabets. Scale-Invariant feature transformation (SIFT) was implemented for the extraction of features, and all the descriptors were then performed using k-means clustering. This hybrid approach was then implemented in the Bag of Words model. Lastly, a binary linear vector support vector classifier was trained with a corresponding training data set.A resulting recognition rate for the device was achieved. Next year Uddin et al.in 2016 [37], a hand sign recognition research was published which claimed that they took RGB images of signs and translated them to HSV and then applied the Gabor filter to obtain the correct characteristics. In standard terms, kernel PCA has perfectly established and defined hand signals (100%), with marginally high precision at different angles (98.6 %), varying lighting conditions (99.5%), and an average accuracy of 99.5% for dimensional reduction.

Naglot et al. used a leap motion controller this year in [35], which is a 3D contactless motion sensor which maps hands and fingers and can detect them.The multi-layered perceptron (MLP) with a neural network, combined with a Back Propagation algorithm, generates the classification model by taking the feature sets as inputs. There were 520 samples in the dataset, which is 20 samples of each of the 26 ASL alphabets, and their method was 96.15% effective in identifying hand signals. Nev-

ertheless, due to the similarities between a few signs and the hand gesture's direction in front of the LMC, certain gestures are not clearly interpreted.The model defined by (Soni et al. 2016) [36] is implemented using MATLAB. The signs are captured and further analyzed using a webcam, and the features are derived using PCA from the captured photos.The comparison of features is rendered using Euclidean Distance of the training sets. The minimum distance of Euclidean permits recognition of the character. This framework would help non-language speakers to understand and communicate with people with hearing impairments more effectively. They achieved almost 98% precision under strong lighting and a clear light-colored backdrop.Next year, in (Santa et al.2017) [39], For skin color segmentation, first RGB images with hand signs are transfigured into YCbCr color space. Subsequently, the region of interest (ROI) and portions were obtained by deleting other portions of the skin [39], and Local Binary Pattern was added to it for extracting features and performing Support Vector Machine (SVM) features for classifying candidate features. They reached 94.26% comprehension precision for words and 94.49% for sentences. However, their system has experienced issues detecting all the hands effectively if they are separated, and the result of this scheme is significantly suffered due to low contrast video.

Also in 2017, implementing the skin detection algorithm and removing all noise from MATLAB, the hand gesture in [40] was correctly identified by Uddin et al. in 2017 and the image categorized according to a signing gesture. Their model uses YCbCr algorithms to detect all types of skin colors and uses Bag of feature for feature extraction and a support vector machine (SVM) for training and evaluation. They used both the men's and women's hand movements to test their suggested model with their own Bangladeshi sign languages dataset. The overall accuracy of the measurement range is 86%. A total of 830 pictures, 15 groups, and some 54 to 56 types were used, but they did not have sufficient data set resources.Also, In [38], (Ahmed et al. 2017) implemented a method in which a message is presented as a sign. Kinect takes the user's depth image. The Kinect SDK system was used to track the Skeleton. The k-curvature algorithm is used for finger detection. Later Sarkar et al. (2018) [41] implemented a paper where they used various sensors like a gyroscope, accelerometer, etc. these sensors measure the movements and bending values of limbs. Then this data goes into the PC by Bluetooth. Then it gets compared by other datasets, and recognition takes place. They got an accuracy of 99.5%. This work is very primitive as any change in the input value can cause miss calculation, and it would not recognize the sign. Other tasks also provide more systems to recognize sign language. Subsequently, Urmee et al. (2019), in their paper[45], delivered a system where they take still images and deliver them into convolutional neural networks. For this particular system, they used Xception model architecture to achieve 98.93% accuracy. Another work was done by Hossain et al. (2020) [46]. They made the system only capable of identifying static still pictures. The images were fed to CNN and got an accuracy of 98.75%.

Chapter 4

Proposed Methodology

In this chapter, we discuss about data description and workflow of our proposed model. We implemented a step-by-step method of performing the experiment to legitimize our analysis in a flowchart in Figure 4.1.

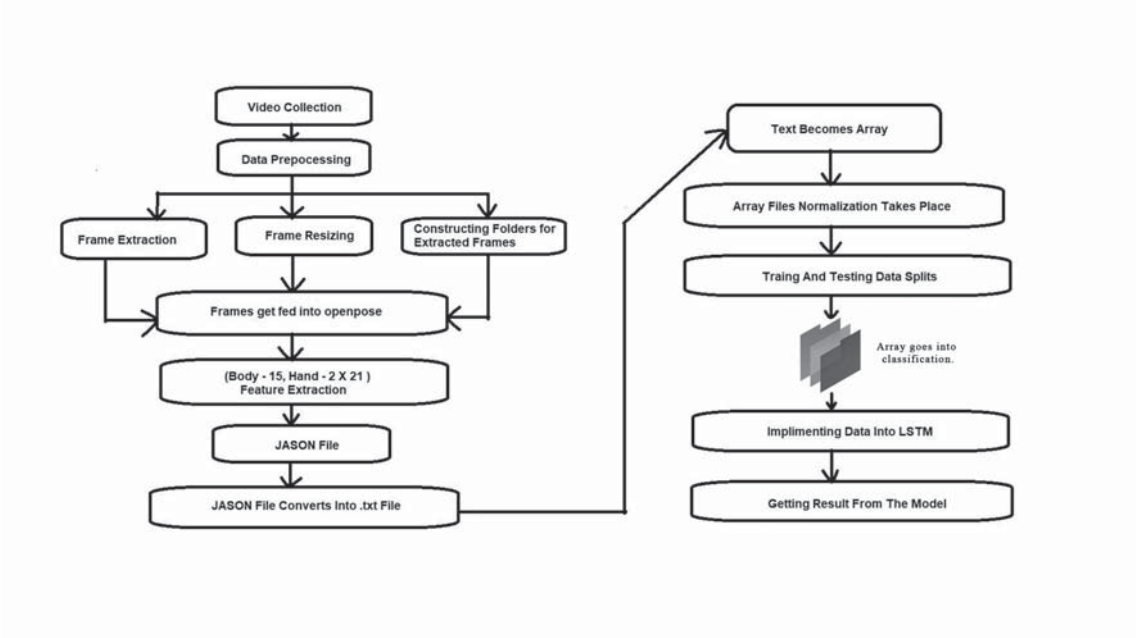


Figure 4.1: Flowchart of our Proposed Model

4.1 Data Collection and Dataset Description

Our initial plan was to collect data from the Dhaka Badhir High School. Unfortunately, due to the CoViD-19 situation, Dhaka Badhir School is closed from March 2020. Therefore, our initial plan could not be executed, and we could not collect the data from Dhaka Badhir High School students and teachers in person. Hence, we searched the internet for a dataset relevant for our research purpose. We found out from our search that there is very little work on Bangladeshi sign language words. Moreover, we needed video data of sign words because signs of words are not always static. There are movement, position changes in one sign word, which make the signs dynamic—those we could not find anywhere on the internet for Bangladeshi

sign language. As a result, we created our dataset with some relative members, friends, and ourselves.

For the dataset, we selected 10 Bengali words. We followed various sign language sources to confirm the signs. Primarily, only the thesis members learned how to perform the signs from the YouTube video and "Bangla Ishara Vashar Ovidhan." After that, we taught others some of our friends and relatives and ensured they performed the signs correctly. Then finally, in various steps, we collected the data. 10 words from our dataset are:

- Bangla
- Bhasha
- Amar
- Apnar
- Dhonnobad
- Naam
- Bhai
- Daktar
- Jama
- Shundor

In this research, as stated above, the dataset is consists of 10 words. Each sign is performed by 10-11 signers, among which 6 are females, and 4-5 are males. In our dataset, there are 1151 videos in total.

The process of collecting data is camera setup - background setup - position analysis - precision in video length.

No.	Words	Amount of Videos	No.	Words	Amount of Videos
1.	Bangla	135	2.	Bhasha	132
3.	Amar	125	4.	Apnar	113
5.	Dhonnobad	114	6.	Naam	108
7.	Bhai	112	8.	Daktar	110
9.	Jama	96	10.	Sundor	109

Table 4.1: The amount of data for each word

For our work, firstly, We decided to use a series of videos taken by five RGB cameras. We used two DSLR cameras and three Smartphone cameras for the videos. In those DSLR cameras, the video frame rate is 50 fps, and for the Smartphone cameras, the frame rate is 60fps. Another important factor which we strictly maintained is different backgrounds. Videos were taken in 5 different backgrounds so that data

variation can happen. For each background, we tried to take two videos for each word for one signer. Other than the background in our dataset, we tried to keep variation in lighting too. Some data are taken in daylight; some are in an open place; a few are in low light. Thirdly, while collecting data, position analysis played an essential role because signing space is crucial. We took several videos, primarily practice for the signers, as they are beginners in sign language. After some practice, each sign's position was nearly perfect for some signers; others were all right. Lastly, in our dataset, all the videos are within 4-6 seconds. All the signers had to perform the signs properly so that the frames do not get hazy when frames are taken from the videos.

For each sign, there are continuous gestures that make a sign meaningful. Sign representation for each word from our data set is given in figure 4.2 - 4.11.

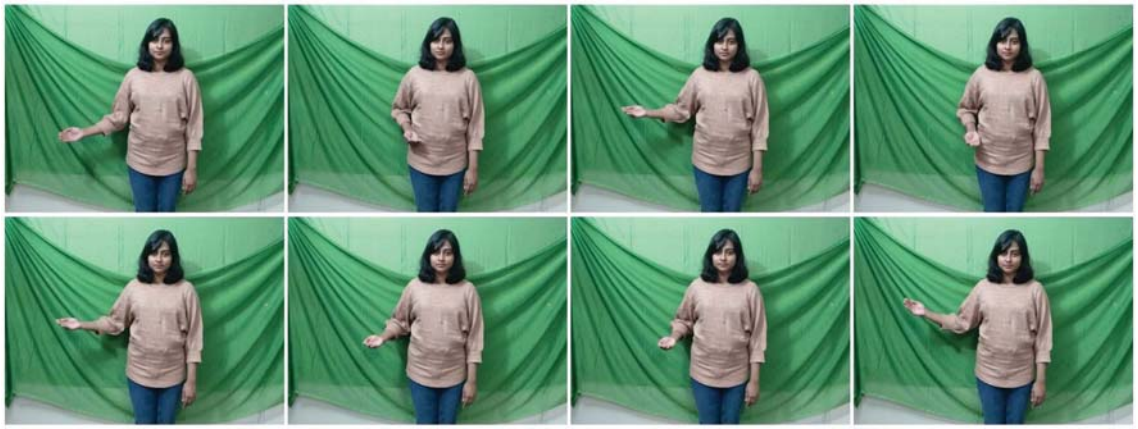


Figure 4.2: Complete sequential gesture for Bangla.

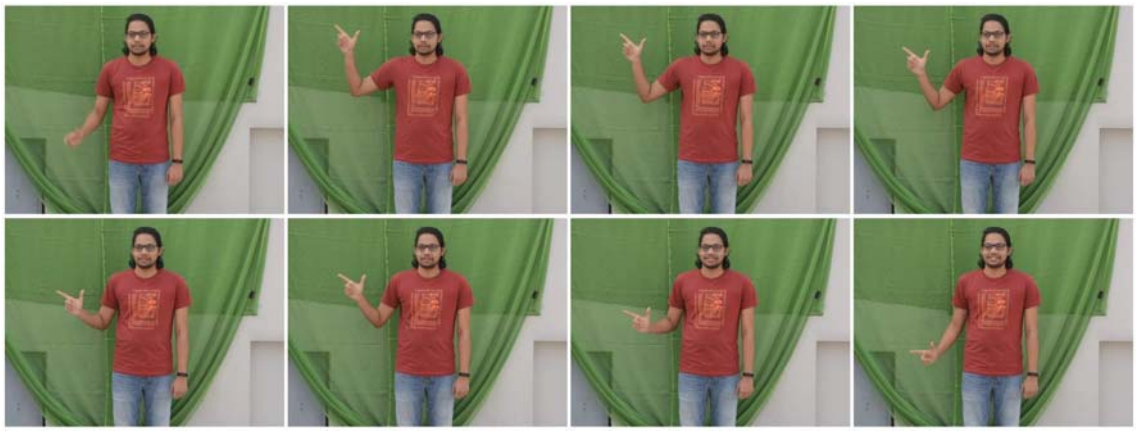


Figure 4.3: Complete sequential gesture for "Bhasha".

Sign language reorganization using a 2-dimensional RGB camera is problematic. For identification purposes, we had to extract significant and noiseless features from the videos that would be captured. 3-dimensional cameras that can collect detail depth can do this job quickly. However, we wanted to use a 2-dimensional camera as it is cheaper than 3-dimensional cameras, and the device is available to all. Firstly,

Figure 4.4: Complete sequential gesture for "Amar".

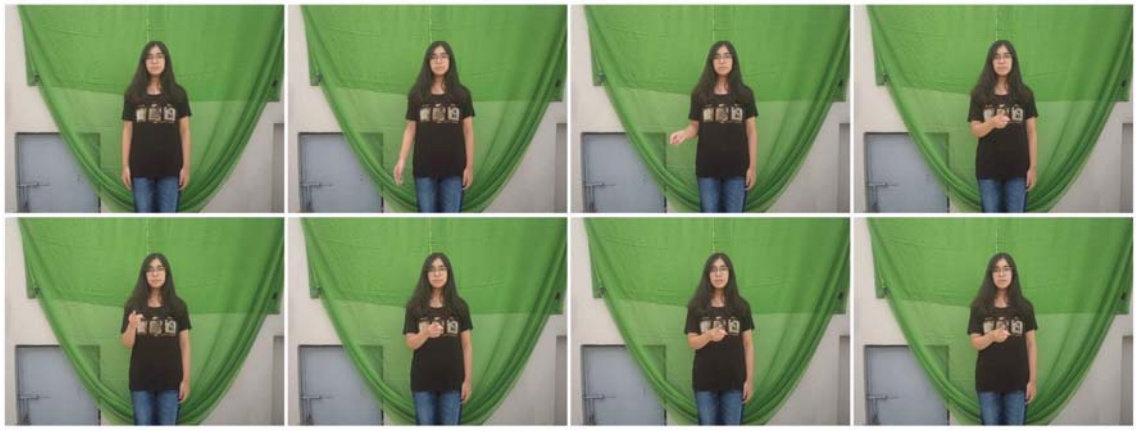


Figure 4.5: Complete sequential gesture for "Apnar".

Figure 4.6: Complete sequential gesture for "Dhonnobad"

frames are extracted from the video. Resized frames are kept in video-wise directories; then, these directories are given as input to the OpenPose, which extracts the frames' features for the classification purpose. The extracted features, along with temporal data, are goes through four classification algorithms: "Gesture Classification LSTM module.", K Neighbours Classification, Gaussian Naive Bias, Support Vector Machine (SVM).

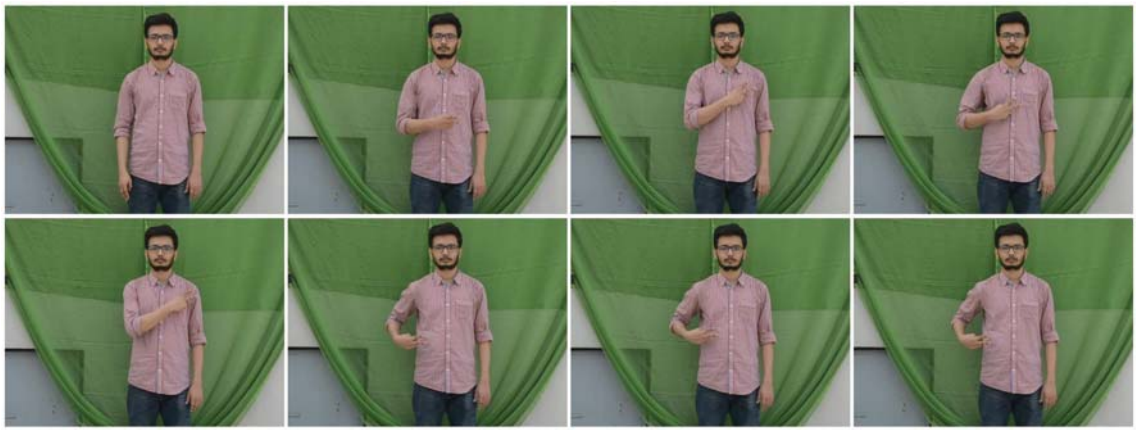


Figure 4.7: Complete sequential gesture for "Naam".

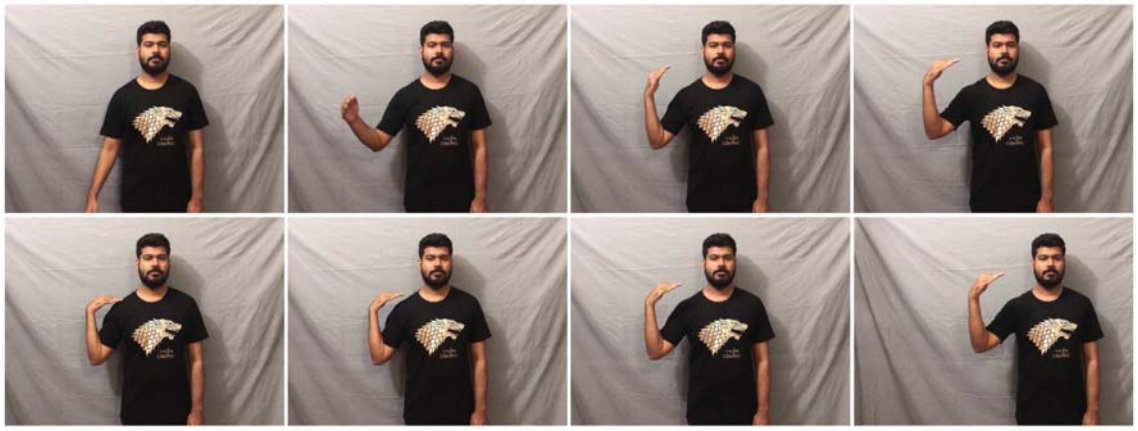


Figure 4.8: Complete sequential gesture for "Bhai".

Figure 4.9: Complete sequential gesture for "Daktar"

4.2 Data Pre-processing

After taking the videos, we kept them in a folder named videos. Under the videos folder, there are ten separate folders for each word. Each word folder has video files in them. The exact amount of videos for each word is given previously.

Figure 4.10: Complete sequential gesture for "Jama".

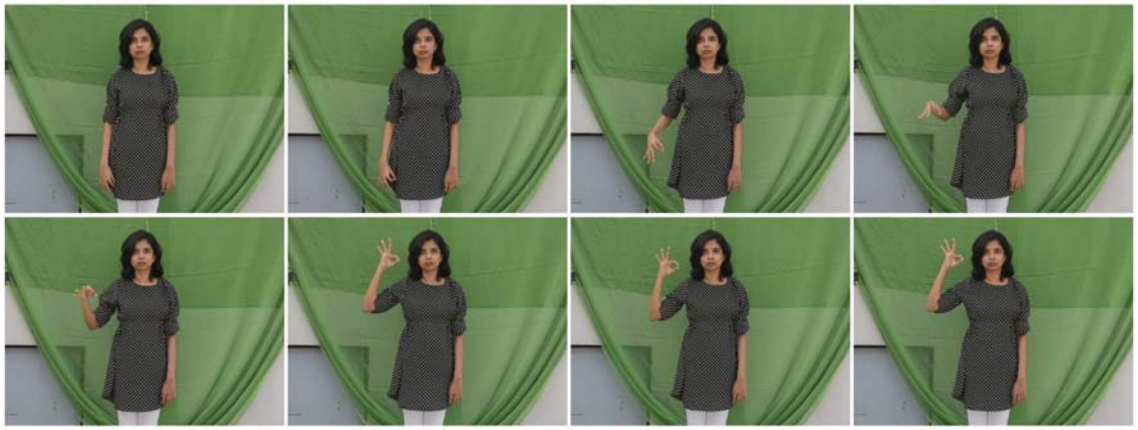


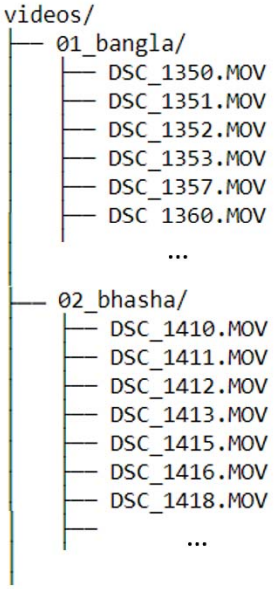
Figure 4.11: Complete sequential gesture for "Sundor".

4.2.1 Frame Extraction

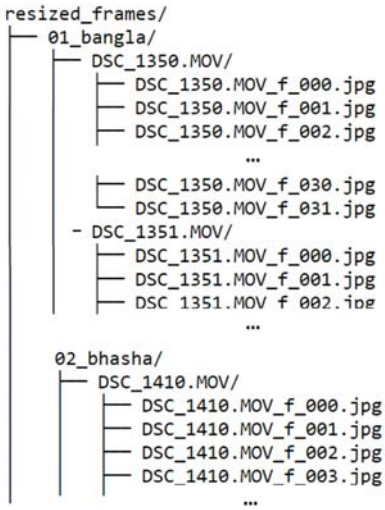
We captured the videos at 50-60 frames per second. Moreover, for each video, the length is 4-6 seconds. As a result, we got 200-360 frames for each video. All the frames are not essential for our task. We extracted the frames from the video and selected only the necessary frames to determine the signs. Hence, we took evenly spaced 32 frames from each video. Evenly spaced means if one video has 320 frames, we took one frame after every ten frames. In this way, we got 32 frames for that video.

4.2.2 Frame Resize

Primarily there were two types of video resolution. The total number of pixels contained in each frame is the resolution of the video. The videos were taken in 720 x1280p and 1080 x1920p.For our work, we resized all the frames in 720 x 1280p. After frame extraction and resize, the folder structure changed. Now each video file is an individual folder, and under that folder, there are 32 frames for each video. For example: under the "Naam" folder, there are 108 videos. After taking evenly-spaced 32 frames for each video, we got 3456 frames in total.



(a) Folder structure.



(b) Folder structure of resized image.

Figure 4.12: Folder structure in pre-processing stage.

4.3 Feature Extraction with OpenPose

OpenPose is the first multi-person visual interface to collectively recognize key-points for the human body, hand, face, and foot (135 keypoints in total) on single images. [42] Resized frames are kept in video-wise directories, and then these directories are given as input to OpenPose.For individual bodies detected in the image, which produces two-dimensional structural key-points. OpenPose can take videos and images both as inputs. In our process, we fed images to OpenPose as inputs. In two-branched CNN, the first stage predicts confidence maps, and the second stage predicts part-affinity fields - a 2D vector that encodes the direction and orientation of each limb. [42] [43] To visualize the 2D keypoints of all people in the picture, both the confidence maps and affinity fields are parsed by greedy inference. [42]

4.3.1 Confidence Maps

A Confidence Map is a 2D illustration of the assumption that any given pixel can be located in a specific body component.The following equation defines confidence maps: [42]:

$$S = (S_1, S_2, ., S_J)\, where S_j \epsilon R^{w*h}, j \epsilon 1 \ldots J \tag{4.1}$$

Where J represents the number of positions for body components.

4.3.2 Part Affinity Fields

Part Affinity is a series of 2D vector fields encoding various individuals' joints' position and direction in the image. In the form of pair-wise connections between

body parts, it encodes the data.[42]

$$L = (L_1, L_2 \ldots, L_C)\, where L_c \epsilon R^{w*h*c}, c\epsilon 1 \ldots C \tag{4.2}$$

Three key phases in the multi-CNN structure explain how they function in the context.

4.3.3 Loss functions

To measure the loss between the expected confidence maps and the fields of Part Affinity to the regression coefficients maps and fields, an L2-loss function is used.

$$\begin{aligned} f_{\mathbf{L}}^{t_i} &= \sum_{c=1}^{C} \sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \left\| \mathbf{L}_c^{t_i}(\mathbf{p}) - \mathbf{L}_c^*(\mathbf{p}) \right\|_2^2 \\ f_{\mathbf{S}}^{t_k} &= \sum_{j=1}^{J} \sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \left\| \mathbf{S}_j^{t_k}(\mathbf{p}) - \mathbf{S}_j^*(\mathbf{p}) \right\|_2^2 \end{aligned} \tag{4.3}$$

Where $\mathbf{L}_c^*$ represents the ground truth part affinity fields, $\mathbf{S}_j^*$ represents the ground truth part confidence map, and W is denoted as a binary mask with $\mathbf{W}(\mathbf{p}) = 0$ when the annotation is missing at the pixel p. That is to avoid the extra losses that these masks will generate.
At each point, intermediate monitoring is used to resolve the gradient problem by regularly regenerating the gradient.

$$f = \sum_{t=1}^{T_P} f_{\mathrm{L}}^t + \sum_{t=T_P+1}^{T_P+T_C} f_{\mathbf{S}}^t \tag{4.4}$$

The Confidence maps for each person k and each body part j is defined by:

$$\mathbf{S}_{j,k}^*(\mathbf{p}) = \exp\left(-\frac{\|\mathbf{p} - \mathbf{x}_{j,k}\|_2^2}{\sigma^2} \right) \tag{4.5}$$

It is a Gaussian curve with incremental shifts where the distribution of the peak is regulated by sigma. An average of the individual confidence maps by a maximum operator is the estimated peak of the network.
Particularly in multiple person pose identification, the part affinity area is needed that we are expected to map the appropriate body parts to their body. There could be two people in the future, as in our dataset, so we used it. There are multiple heads, hands, shoulders, and so on with multiple individuals. Sometimes, as they are closely clustered together, it becomes impossible to differentiate. PAF gives the relation between various parts of the body that belong to the same entity. A more substantial PAF relation between body parts reflects the high probability that certain body parts correspond to the same person.
If the p indicates on the limb, then L* indicates the unit vector otherwise it is 0.

$$\mathbf{L}_{c,k}^*(\mathbf{p}) = \begin{cases} \mathbf{v} & \text{if } \mathbf{p} \text{ on limb } c,k \\ \mathbf{0} & \text{otherwise} \end{cases} \quad \mathbf{v} = (\mathbf{x}_{j_2,k} - \mathbf{x}_{j_1,k}) \,/\, \|\mathbf{x}_{j_2,k} - \mathbf{x}_{j_1,k}\|_2 \tag{4.6}$$

The expected part affinity area, Lc along the line section, is to test confidence for dj1 and dj22 candidate part positions.:

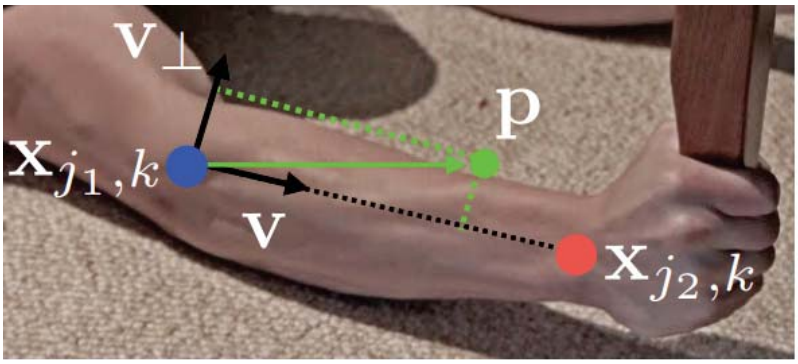


Figure 4.13: Equation for connection between different part of the body [42].

$$E = \int_{u=0}^{u=1} \mathbf{L}_c(\mathbf{p}(u)) \cdot \frac{\mathbf{d}_{j_2} - \mathbf{d}_{j_1}}{\|\mathbf{d}_{j_2} - \mathbf{d}_{j_1}\|_2} du, \tag{4.7}$$

It's necessary to optimize Total E for multiple person

$$\max_{\mathcal{Z}_c} E_c = \max_{\mathcal{Z}_c} \sum_{m \in \mathcal{D}_{j_1}} \sum_{n \in \mathcal{D}_{j_2}} E_{mn} \cdot z_{j_1 j_2}^{mn} \tag{4.8}$$

4.3.4 Model Selection

Currently, there are three different models available in OpenPose architecture. 1st one is trained based on a dataset that is Multi-Person Dataset (MPII). It produces 15 key points on a human image. 2nd model was trained based on the COCO dataset. It produces 18 key points on a single human image. The 2016 COCO key points challenge was won by OpenPose architecture.

Moreover, there is a body 25 architecture. There is a model for hand keypoint estimation. We used this architecture in our work, along with the hand parameter. It produces 21 keypoints from each palm and fingers, in total 42 keypoints from two hands. OpenPose gives output in X, Y, C format. X denotes the X-axis pixel value, Y denotes the y-axis pixel value, and C denotes confidence score. The confidence score declares the precision of X and Y values. The confidence score is always between 0-1.1 indicates it is 100% sure about the X and Y value, and 0 indicates it is not sure about the values at all. In our model, OpenPose gives X, Y, and C values for each 25 body points and 42 hand points.
To understand the signs, we only need 15 key points from 25 points. So from the Body_25 model, we are using only 15 points selectively. The selected points are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 12, 15, 16, 17, 18.

In total, our model is using 57 key points, which includes body 15 and hand 42 points. These keypoints are acting as "features." OpenPose is giving X, Y, C three values for each point. We disabled the C factor for our model. As a result of OpenPose, we are getting $57 * 2 = 114$ features as output.

In the X array, there are 36832 rows, which include 114 features. In the y array, there are 1151 rows—1 value on each row denoting the label of every 32 rows in X.

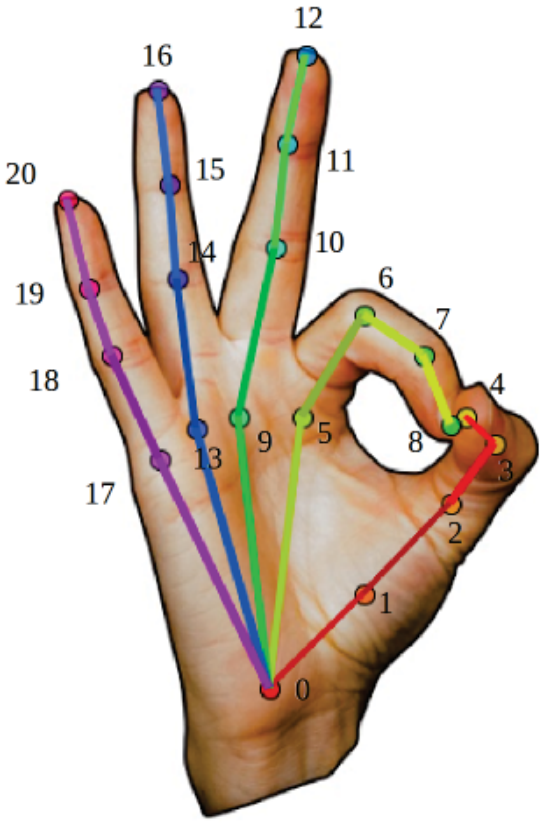


Figure 4.14: 21 hands key points of one hand.

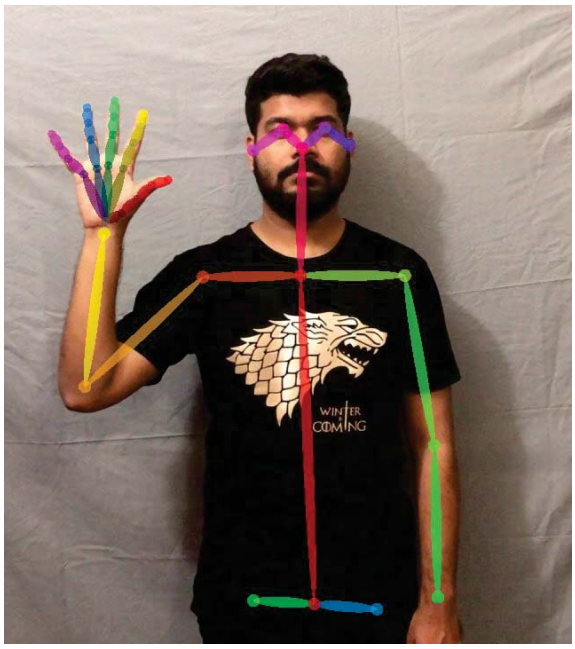


Figure 4.15: 57 keypoints of each frame.

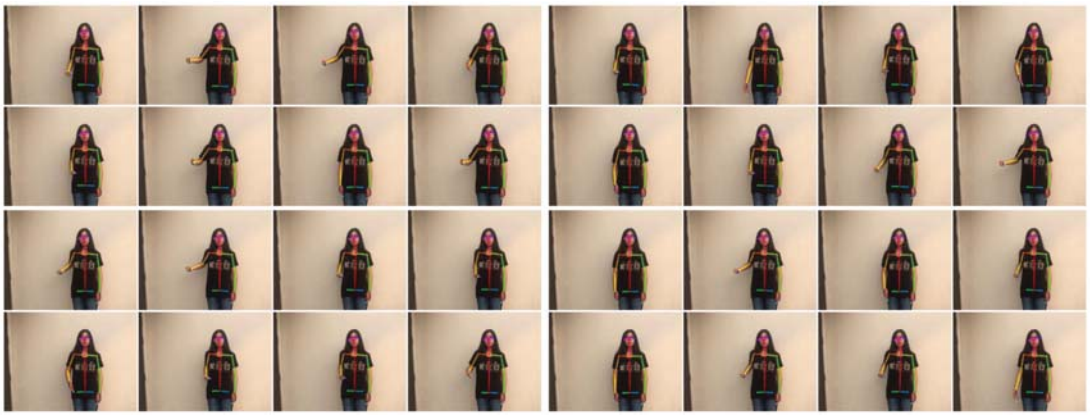


Figure 4.16: 57 key points detection for 32 frames for one video

4.3.5 OpenPose Output Format

There are several options to get the OpenPose output. We are getting JSON files where the people pose data are saved. For each image, we are getting one JSON file. JSON files are converted into .txt files. From there, the array is created.

This approach selected for "Feature Extraction" decreases the reliance on the subject and the backgrounds and only helps concentrate on the subject's gesture.

4.4 Classification Model

A classification model has the functionality to assess an outcome, i.e., input variables, from a series of observed values. The findings are to be labeled and can be applied to the dataset of concern. We normalized the data by dividing each X value with 1280 and each Y value with 720. We used normalization to converge the model faster.

There are mainly two approaches to implementing classification algorithms. They are - supervised learning and unsupervised learning. In a supervised model, to construct and train the model, the classification algorithm is trained using a training dataset. After training, we try to predict the classes of the testing data with that model and determine the model's accuracy based on the performance. In our research work, we have implemented the supervised learning method. The training and test data set was derived from the OpenPose output, which is now in arrays. The dataset was divided into 80%:20% ratio. 80% of the data belonged to the training dataset, while 20% of the data belonged to the testing dataset. It was done using the function train_test_split() imported from the module "sci-kit-learn." $X\_train.shape, Y\_train.shape, X\_test.shapeY\_test.shape$ in this four-part. This kind of splitting of the dataset was done for all four classification algorithms, namely LSTM based RNN, K Neighbors Classification, Support Vector Machine (SVM), and Naive Bayesian (NB).

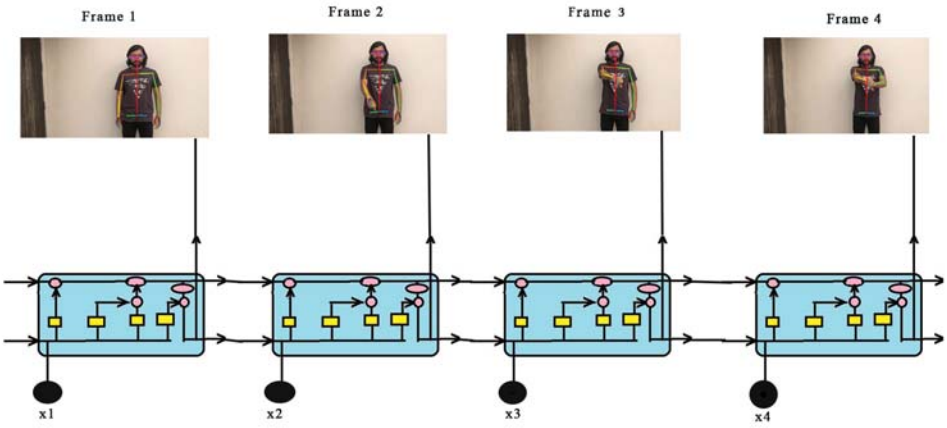


Figure 4.17: Process Under LSTM

4.4.1 LSTM Based RNN implementation

An RNN with LSTM cells was introduced to tackle the temporal dependencies in our results. Previously, RNNs and LSTMs have been proved to be efficient in modeling temporal sequences such as those used in speech [26] and handwriting recognition. [19] This is due to their ability to maintain 'memory' over a series of time steps by enabling the states to impact the present state of the RNNs from subsequent time steps. Although this makes the architecture a good option for modeling time series results, there are some limitations when working with long time series. If the duration of the sequence becomes too long, during back-propagation over time, the RNN may suffer from vanishing or exploding gradients (BPTT). By incorporating multiple learnable parameters or gates that influence weight updates during BPTT, LSTMs minimize this concern, providing greater control over what is stored in the LSTM cell's internal state and what it 'forgets' each point of time. The .text file obtained from the feature extraction module is fed as the input of this module. The .text file contains the information of X, Y coordinates of 57 key points, and the frame number of the video. We kept evenly spaced 32 frames for each video for the proper working of "LSTM." The inputs are then fed to the LSTM. The module learns the pattern of the coordinate changes frame by frame and provides the class that the particular video belongs to.

The training data are in X_train.shape and Y_train.shape. 920 is the number of videos, 32 is frame number for each video and 114 is the feature for each frame in X_train. Y_train has 920 rows and 1 value on each row denoting the label of every 32 rows in X_train.

Testing data are in X_test.shape and Y_test.shape. 231 is the number of videos, 32 is frame number for each video and 114 is the feature for each frame in X_test. Y_test has 920 rows and one value on each row denoting the label of every 32 rows in X_test.

The model used in this work consists of two layers containing 128 LSTM cells with Rectified Linear Unit (ReLU) activation function. A dropout layer is inserted after the first LSTM layer with a dropout rate of 0.2. A dense layer with 32 cells and a ReLU activation function is placed before the output layer. The fully connected output layer with the SoftMax activation and ten nodes represents the different signs. We used sparse categorical cross-entropy as the loss function during batch gradient descent, and Adam optimizer is used as the optimizer, where we set the initial learning rate to 0.00001. The model is trained for 200 epochs with a batch size of 16. These hyperparameters were experimentally selected according to which values produced the task's best outcomes. Table **??** summarises the values for different hyperparameters.

Hyperparameter	Value
Learning rate	1e-5
Decay rate	1e-4
Batch size	16
Epoch	200

Table 4.2: Values of different hyperparameters for LSTM based RNN.

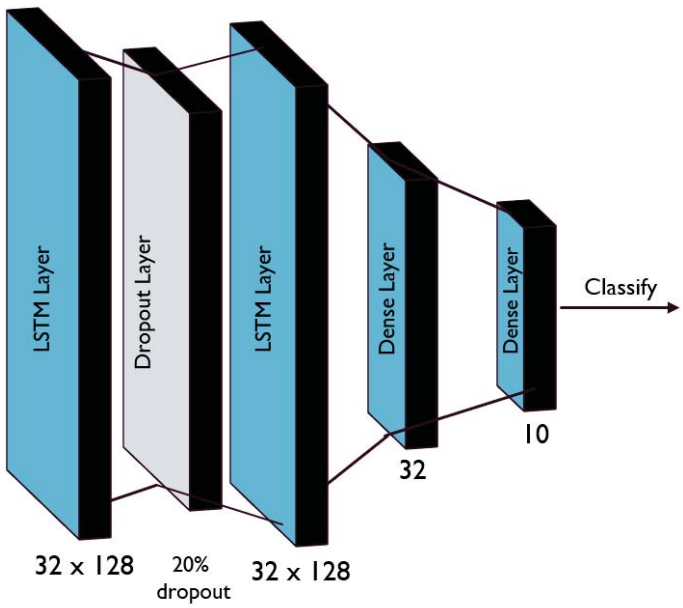


Figure 4.18: Structure of chosen LSTM-RNN model.

4.4.2 K-Neighbors Classifier

K-Neighbors Classifier is a non-parametric and lazy learning algorithm. It does not assume anything about the data, making it very useful in practical uses because most real-life data do not follow any mathematical and theoretical formula. We have used different unit measures to calculate the distance between neighbors and found

no significant differences in the accuracy scores. So, we used Euclidean distance as the unit of measuring distances.

4.4.3 Naive Bayes Classifier

We also implemented the Gaussian Naive Bayes algorithm in our research work. Naive Bayes classifier works based on Bayes theorem, which uses the concepts of conditional probability. Gaussian Naive Bayes is a variant of Naive Bayes that follows Gaussian normal distribution and supports continuous data. Naive Bayes based classifiers take less time for training because to do the classification, and these only need to calculate the likelihood of all the features of the data point.

4.4.4 Support Vector Machine (SVM)

We have ten classes- Bangla, Bhasha, Amar, Apnar, Dhonnobad, Naam, Bhai, Daktar, Jama, and Shundor. Therefore, the hyperplane segregating the ten classes is polynomial. That is why, for this study, we have used the SVM classifier with a polynomial kernel. Using a polynomial kernel SVM improves the classification performance by remarkably reducing the complexity of the model.SVM takes the 2D matrix as input. In our research, we have discussed earlier in the LSTM based RNN that in LSTM, we are feeding X_train.shape (920, 32,114) is the array values. X_test.shape = (231,32,114) Which means it is a 3D matrix. In order to give input the array to SVM, we converted the 3D array to a 2D array.X_train.shape = (920, 3684), X_test.shape=(231, 3684). Similarly, Y_train.shape (920, 1) is converted into Y_train.shape = (920) and Y_test.shape(231, 1) into Y_test.shape=(231) which means 2D to 1D array. We have divided this sample into a train-test split with the train : test ratio being 80:20 of the entire sample. The SVM model uses a polynomial model, as per the discussion above. The SVM model was then learned by fitting the model into the training data using the feature fit(). We used it after the model was learned to forecast effects on the test data. This was done by using the predict() function on the test results.

4.4.5 Stratified 5 Fold Cross Validation

Cross-validation is a resampling procedure used to evaluate machine learning models while training on limited data samples. In k-fold cross-validation, k refers to the number of groups that the given data sample is split into. We have used stratified 5-fold cross-validation for all the classifiers. Stratified cross-validation keeps the same data ratio per class in the testing split as it is in the total dataset.

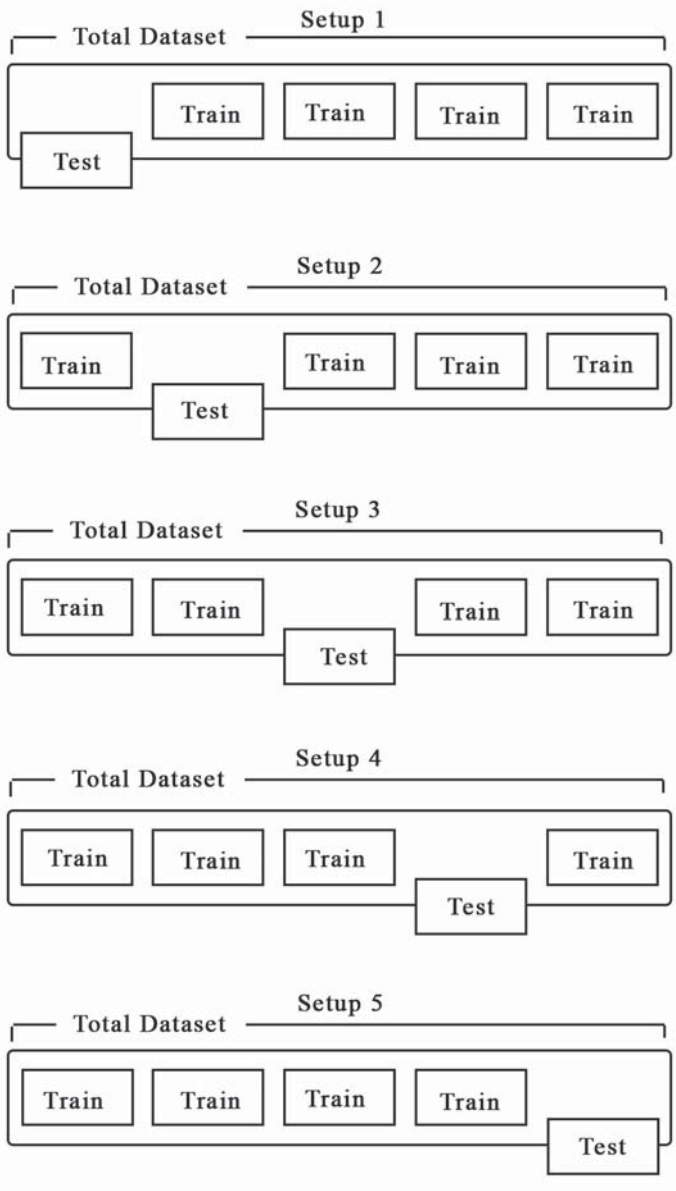


Figure 4.19: The process of cross-validation.

Chapter 5

Experiment and Results

The main purpose of this chapter is to describe results and their interpretation based on the features derived from videos and different configurations of LSTM-RNN models.

5.1 Performance Metrics

5.1.1 Confusion Matrix

Confusion matrices help us understand the output of a classification model on a collection of test data for which the true values are known. For a binary classification problem, this is a table that includes four different combinations of predicted and real values as shown in Figure 5.1 It is handy for measuring Recall, Precision, Specificity, Accuracy, and AUC-ROC Curve.
The accuracy of a classification model is defined as the percentage of correctly classified cases and non-cases among all the example points. Accuracy score can give us an idea of how accurate the classification model is when the dataset is balanced.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{5.1}$$

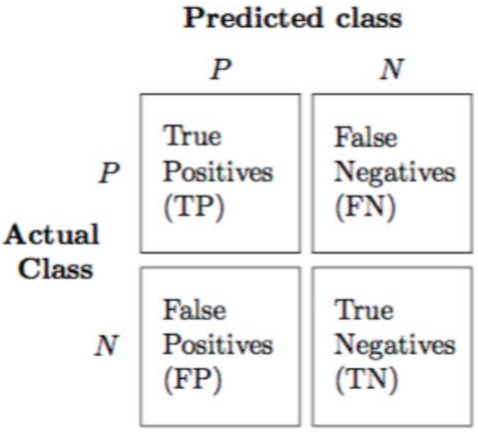


Figure 5.1: Confusion Matrix

The recall or sensitivity of a classification model is defined as the percentage of true cases (TP) that are correctly classified. The measurement of recall is essential for a dataset that is unbalanced.

$$\text{Sensitivity } = \frac{TP}{TP + FN} \tag{5.2}$$

A classification model's specificity is defined as the percentage of true non-cases (TN) correctly classified as non-cases. Specificity is also a required metric for understanding the credibility of a model that is trained on an unbalanced dataset.

$$\text{Specificity } = \frac{TP}{FP + TN} \tag{5.3}$$

The precision of a classifier is defined as the fraction of predicted positives events that are positive. This metric is necessary To understand how accurately a model trained on an unbalanced dataset predicts.

$$Precision = \frac{TP}{TP + FP} \tag{5.4}$$

The f1 score is the harmonic mean of recall and precision, with a higher score as a better model.

$$F - score = \frac{2TP}{2TP + FP + FN} \tag{5.5}$$

The equations show sensitivity, specificity, accuracy, precision, and f-score, respectively. TP denotes true positives, FP denotes false positives, TN denotes true negatives, and FN denotes false negatives.

5.2 Analysis of models with different configurations

5.2.1 K-Neighbors Classifier

The most critical parameter that determines the success of a K-Neighbors Classifier is K. The most appropriate value for K was obtained by finding the accuracy of the model for K's values from 1 through 50. Figure 5.2 shows the accuracy of the model for different values of K. From the experiments; it is seen that the accuracy drops the value of K is greater than 10.

Metrics	Score
Accuracy	84.28%
Precision	86.12%
Recall	85.65%
F1_score	85.56%

Table 5.1: The summary of the results from the model.

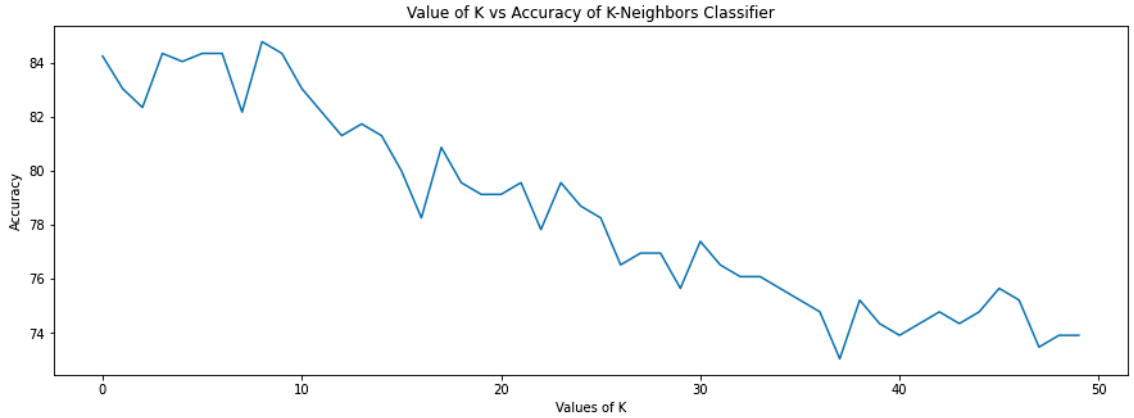


Figure 5.2: Accuracy of K-Neighbors Classifier for different values of K.

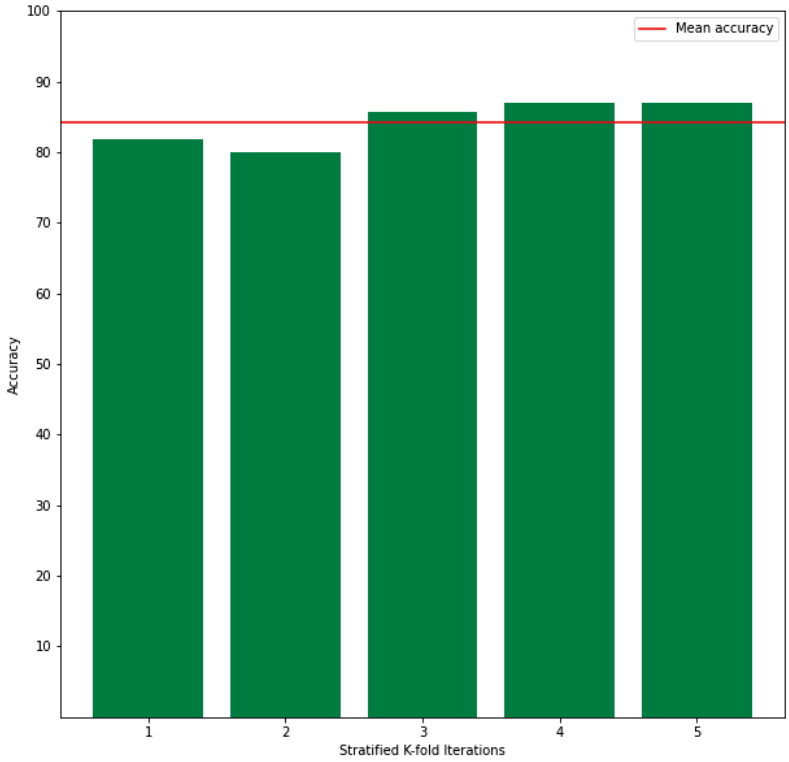


Figure 5.3: Accuracies of K-Neighbors Classifier for each iteration of stratified 5 fold cross validation.

Figure 5.4 shows the breakdown of the performance of the model. We can see that the model can predict the words "Bangla," "Bhasha," and "Daktar" correctly. However, the prediction success of other words varies around 70% to 95%. The overall accuracy of the model with test data is 84.28%.

By applying stratified fivefold cross-validation on the model, we get the model's accuracy for five iterations. Figure 5.3 shows the mean accuracy of the model.
Figure 5.5 shows the obtained confusion matrix for the model. As expected from the accuracy, precision, recall, and f1-score graph, the prediction of "Bangla," "Bhasha," and "Daktar" are perfect, which in other words have some prediction errors.

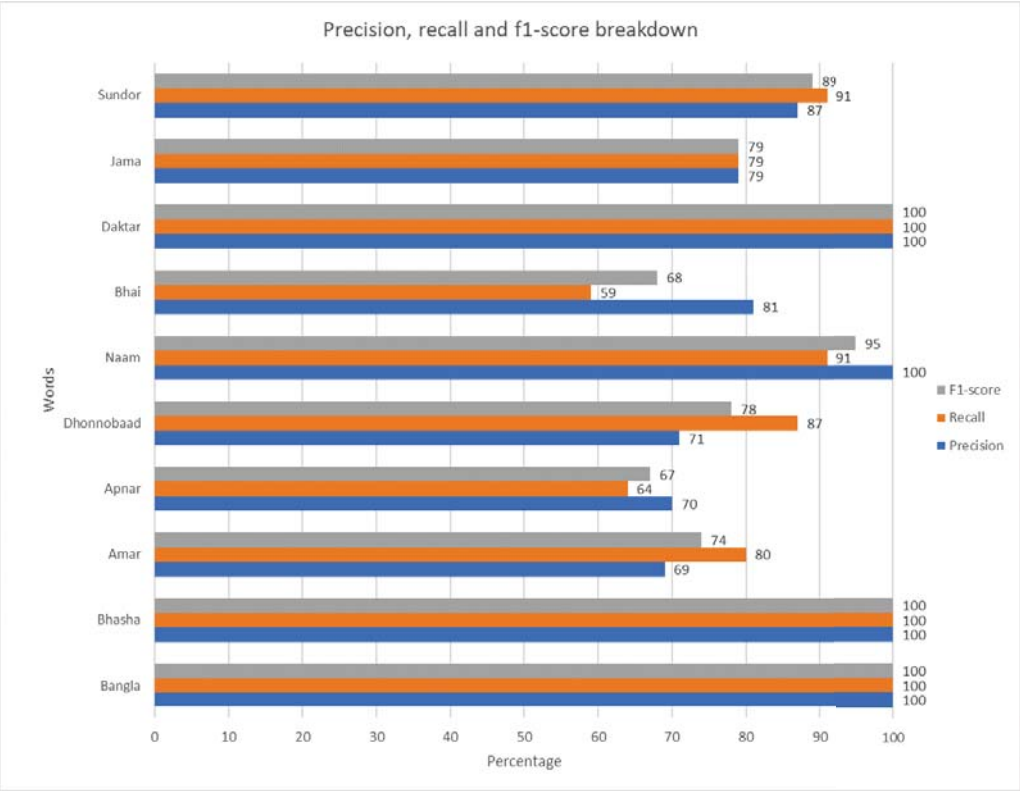


Figure 5.4: Precision, recall and f1-score breakdown of K-Neighbors Classifier model

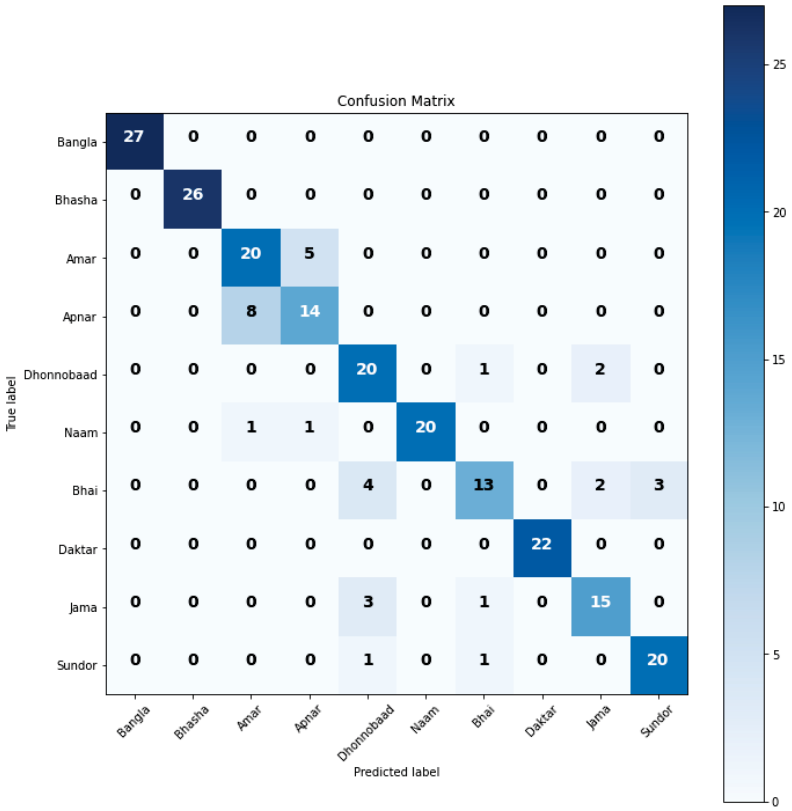


Figure 5.5: Confusion matrix for the K-Neighbors Classifier model

5.2.2 Gaussian Naïve Bayes Classifier

We implemented the Gaussian Naive Bayes Classifier variant in our research work. By applying stratified fivefold cross-validation on the model, we get the model's accuracy for five iterations. Figure 5.6 shows the mean accuracy of the model.

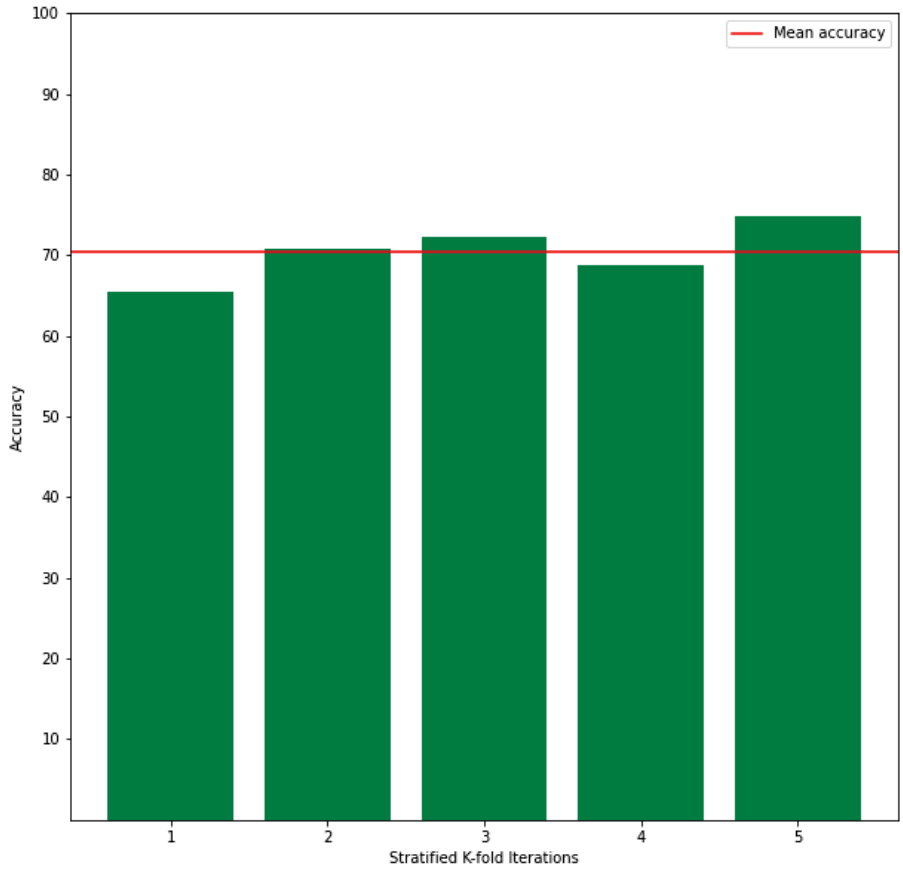


Figure 5.6: Accuracies of Gaussian Naive Bayes Classifier for each iteration of stratified 5 fold cross validation.

Metrics	Score
Accuracy	70.38%
Precision	72.33%
Recall	72.17%
F1_score	71.57%

Table 5.2: The summary of the results from the model.

Figure 5.7 shows the breakdown of the performance of the model. Like the K-Neighbors Classifier, this model can also predict the signs "Bangla" and "Daktar" correctly, but the model performs poorly in detecting other signs.
Figure 5.8 shows the obtained confusion matrix for the model. The confusion matrix shows that the classifier was not good enough to classify most of the signs. Hence, it caused high TN, FP values.

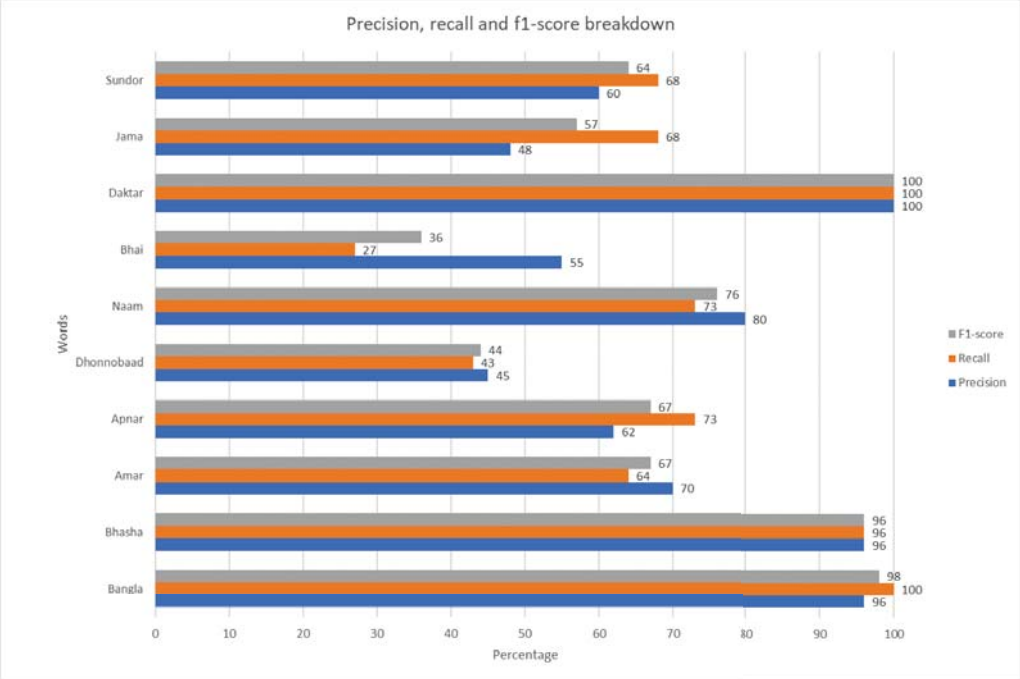


Figure 5.7: Precision, recall and f1-score breakdown of Gaussian Naïve Bias Classifier model

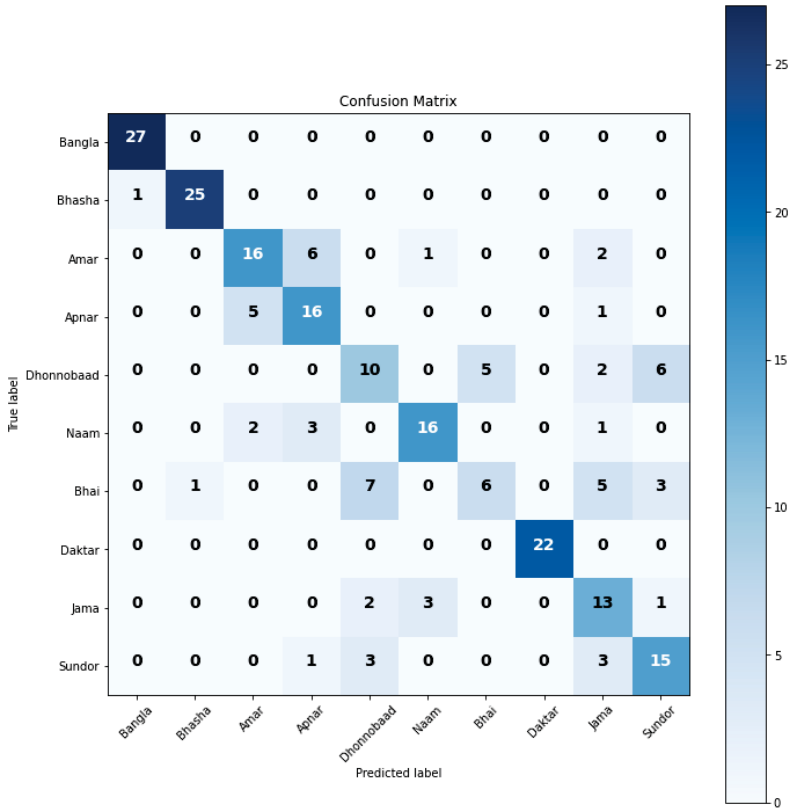


Figure 5.8: Confusion matrix for the Gaussian Naïve Bayes Classifier model

5.2.3 Support Vector Classifier

For the Support Vector Classifier, we apply stratified fivefold cross-validation on the model and get the model's accuracy for five iterations. Figure 5.9 shows the mean accuracy of the model.

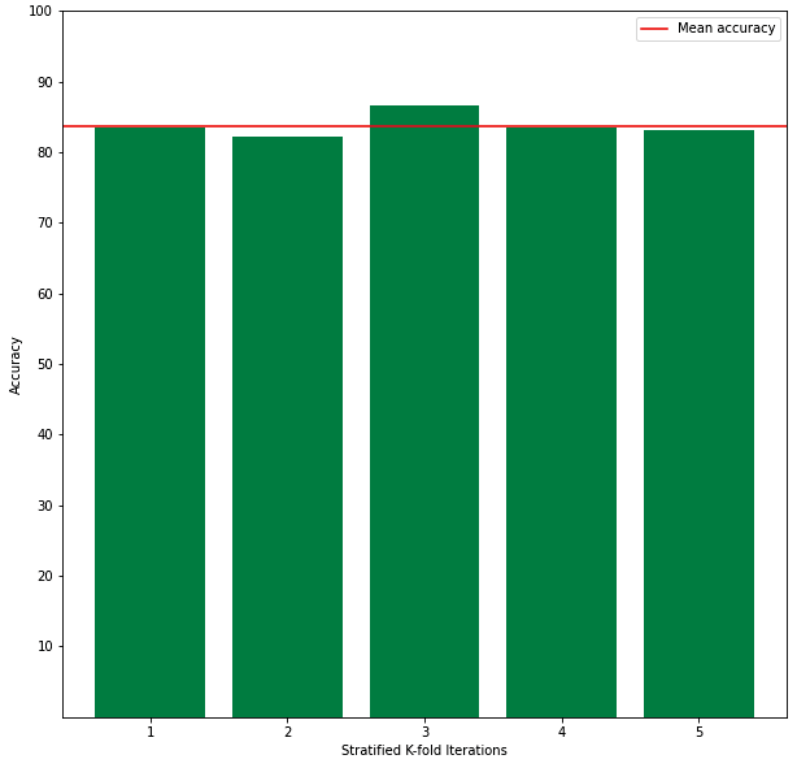


Figure 5.9: Accuracies of Support Vector Classifier for each iteration of stratified 5 fold cross validation.

Metrics	Score
Accuracy	83.75%
Precision	83.83%
Recall	83.55%
F1_score	83.37%

Table 5.3: The summary of the results from the model.

Figure 5.10 shows the breakdown of the performance of the model. The SVC model performed similarly to the K-Neighbor Classifier model. From the three models - SVC, KNC, and G-NBC, we can see that the classifiers performed very well in detecting "Bangla," "Bhasha," and "Daktar." It is because these signs are very distinguishable from the others.

Figure 5.11 shows the obtained confusion matrix for the model. The model performed similar to the KNC model.

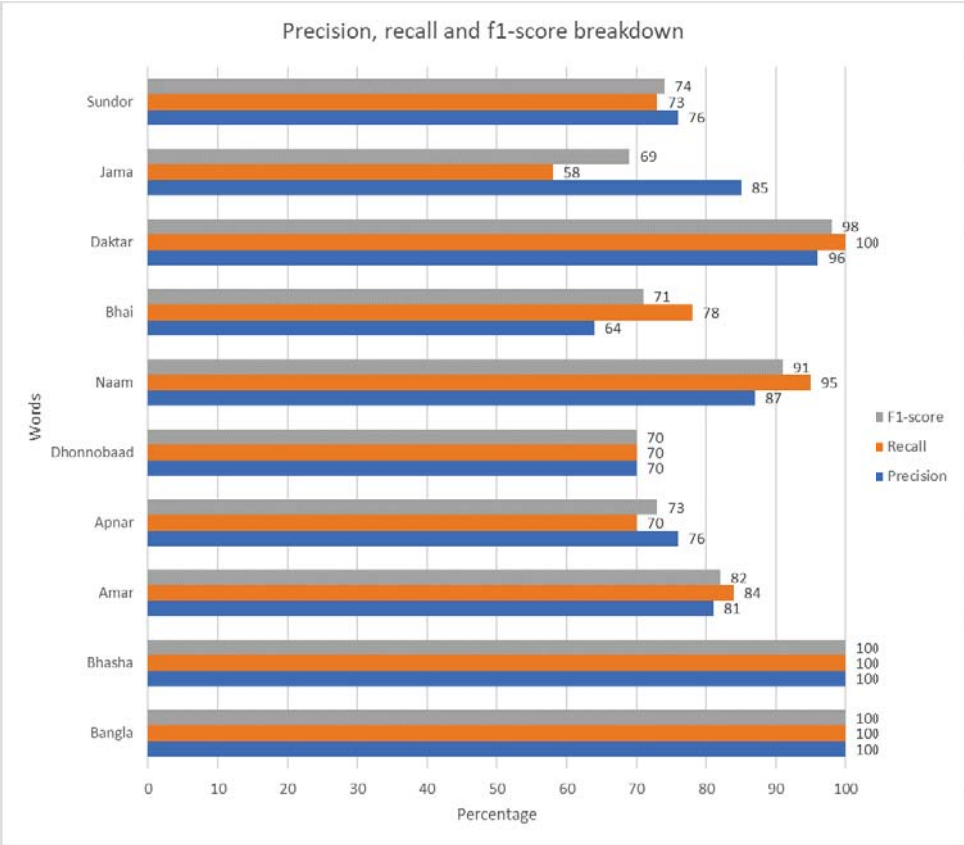


Figure 5.10: Precision, recall and f1-score breakdown of Support Vector Classifier model

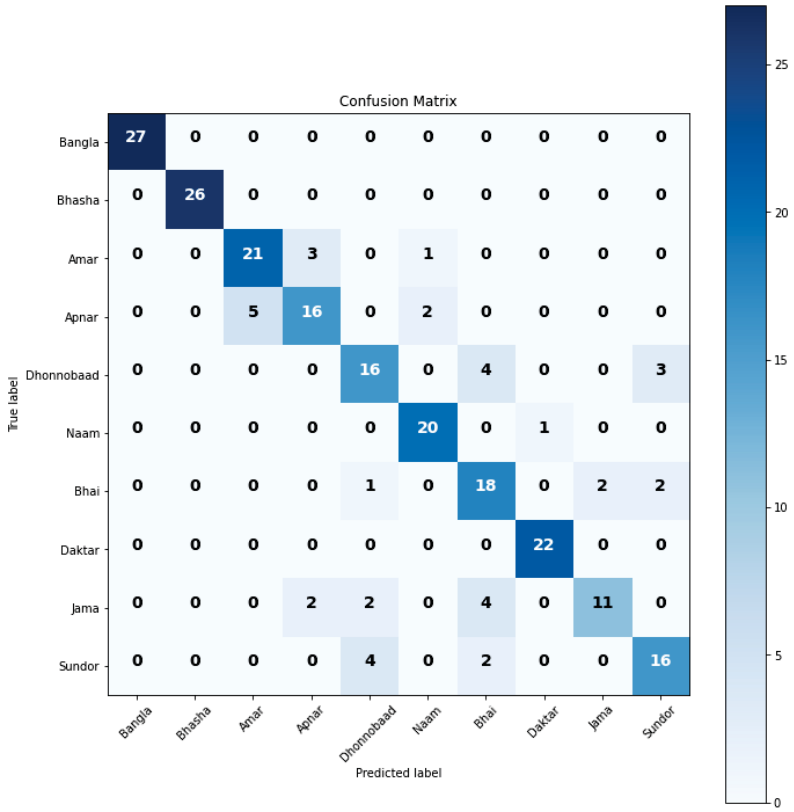


Figure 5.11: Confusion matrix for the Support Vector Classifier model

5.2.4 Long Short-Term Memory based Recurrent Neural Network

To build a deep neural network model that works for a particular problem, it is expected to require some experimentation. Different configurations work better for different problems.

We have tried different configurations of the LSTM-RNN model. Our aim was to build a model that uses low computational power and classifies the signs with a high accuracy rate. We experimented by changing the network breadth, network width, initial learning rate, decay rate, batch size and number of epochs.

We started our experiment with a simple network with 1 LSTM layer with 32 nodes having a ReLU activation function and one fully connected output layer with ten nodes with a SoftMax activation function. The initial learning rate was set to 0.0001, and the decay rate was set to 0.00001. With a batch size of 16, we run this for 100 epochs. Figure 5.12a shows the training and validation accuracy over iterations. Figure 5.13a shows the training and validation loss over iterations. The model was under fitted and gave us an accuracy of 81%.

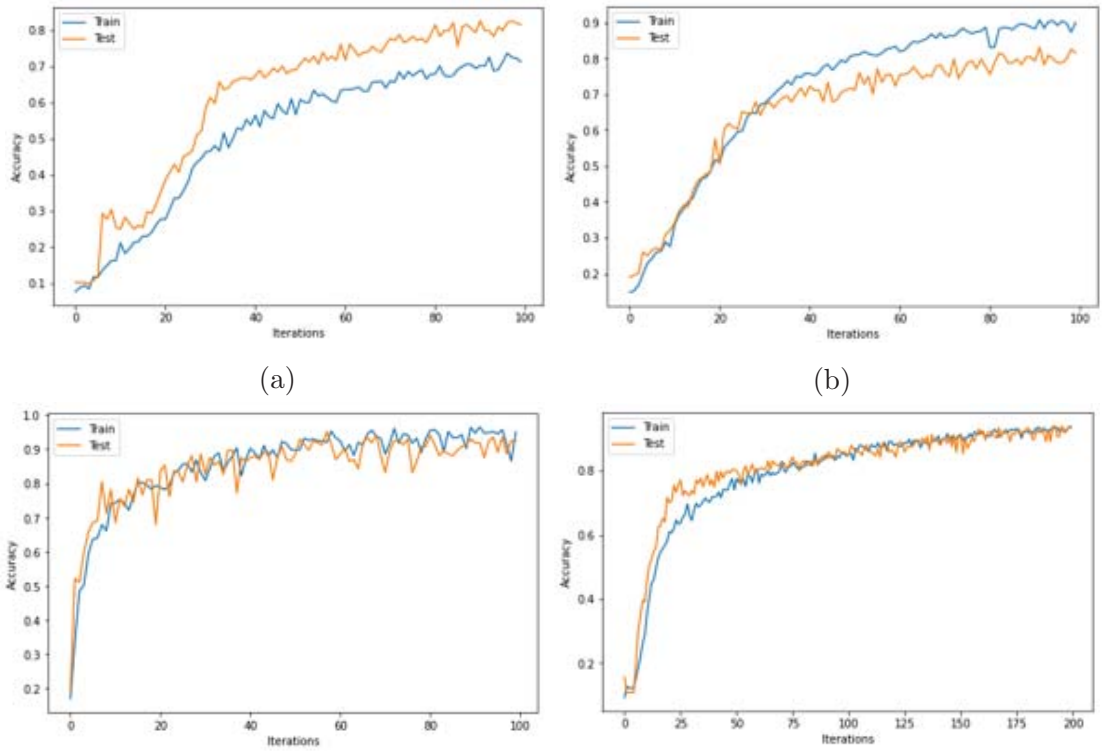
We figured out that the model was too simple for our classification problem. So, we increased the nodes in the LSTM layer to 128. From figure 5.12b, we see that the model accuracy increased to 90%, but it was overfitted. Figure 5.13b shows the training and validation loss over iterations.

When a model gets overfitted, we assume that some of the weights in the model is getting overly weighted. In order to fix that, we can nullify a percentage of the weights in each iteration. So, we add a dropout rate of 20% to tackle the overfitting issue. Figure 5.12c denotes the result we received after making these changes. Figure 5.13c shows the training and validation loss over iterations. The overfitting issue is almost resolved. However, the accuracy over iterations keeps flickering frequently.

Adding another LSTM layer with 128 nodes with ReLU activation function and a dense layer with 32 nodes before the output layer and changing the learning rate to 1e-5 and decay rate to 1e-4 gave the model stability. Running the model for 200 epochs gave an accuracy of 96.54%. Figure 5.12d shows the accuracy over iteration for this network. Fig 5.14 depicts the performance breakdown of the model.

Metrics	Score
Training Accuracy	94.06%
Validation Accuracy	94.02%
Testing Accuracy	96.54%

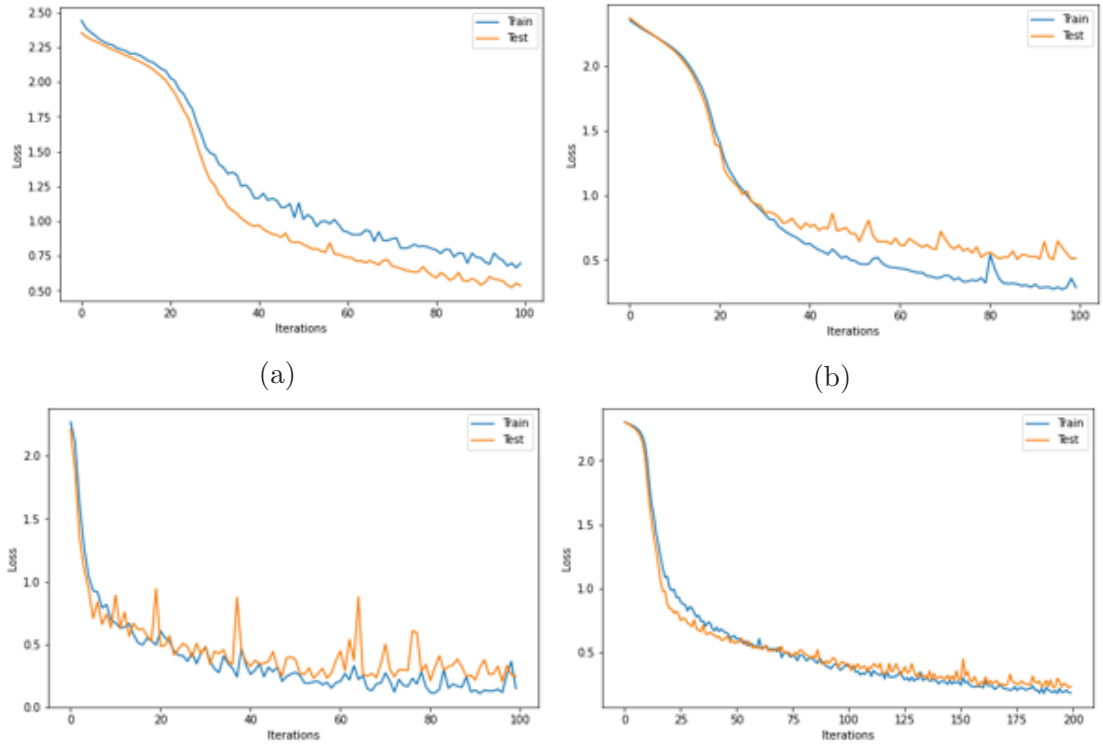
Table 5.4: Evaluation metrics obtained from LSTM based RNN.



(a) (b)

(c) (d)

Figure 5.12: Training and validation accuracy over iterations for various LTSM-RNN configurations.



(a) (b)

(c) (d)

Figure 5.13: Training and validation loss over iterations for various LTSM-RNN configurations.

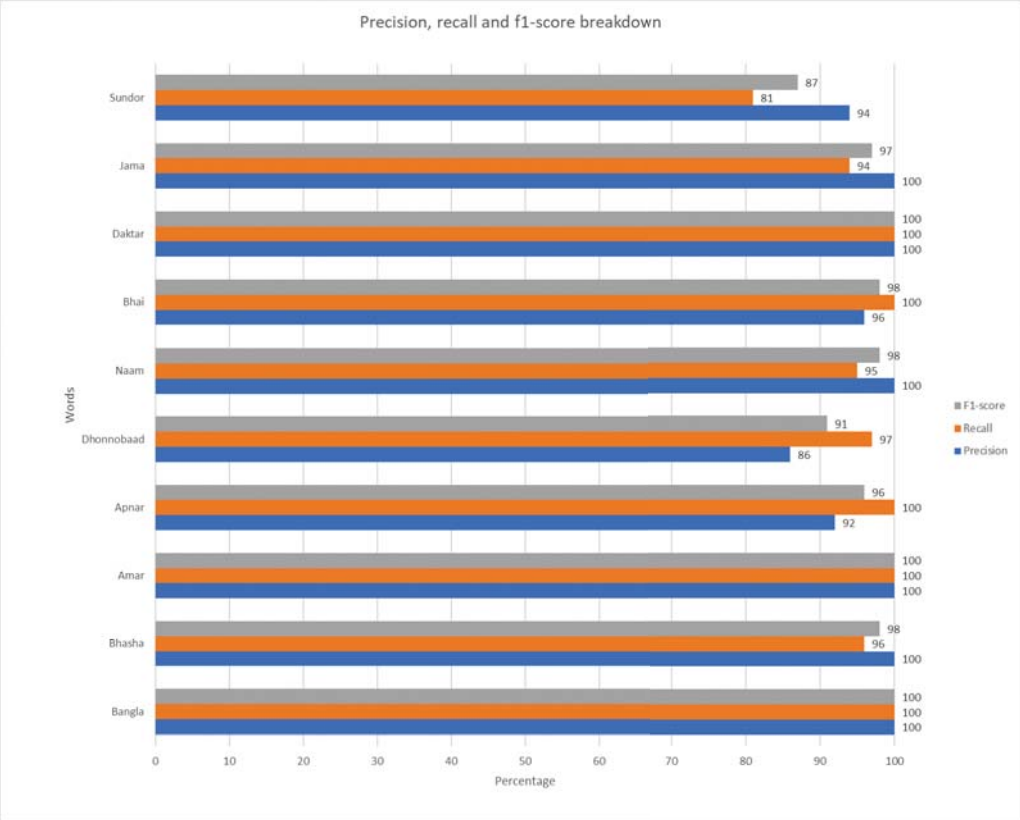


Figure 5.14: Precision, recall and f1-score breakdown of LSTM based RNN.

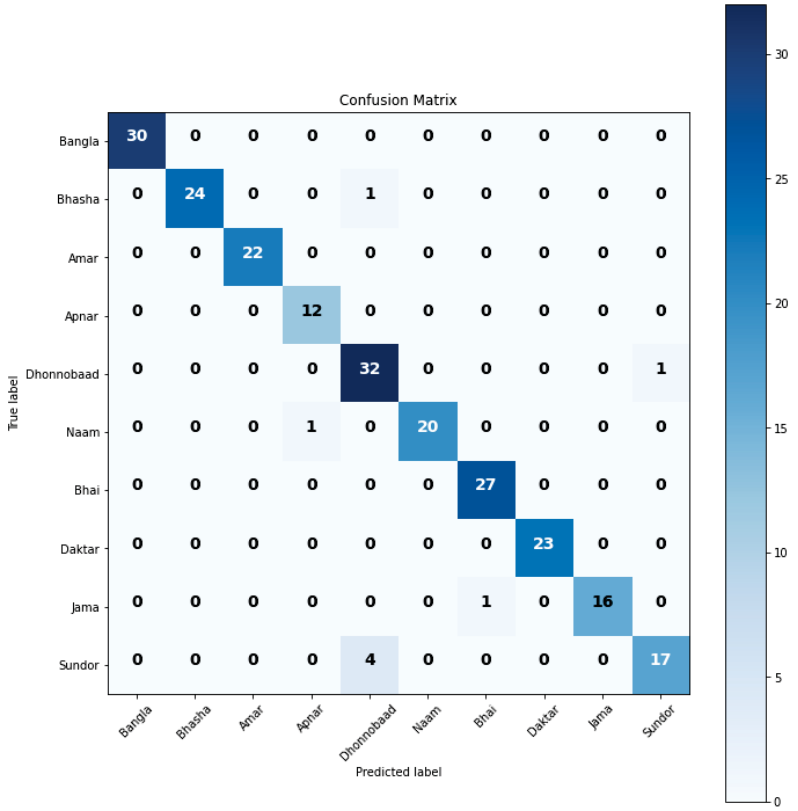


Figure 5.15: Confusion matrix of the predictions from LSTM based RNN. From the matrix, we can see that the model does a great job in classifying all of the signs.

5.2.5 Result Analysis

Figure 5.16 shows the accuracy scores of different classifiers. From the figure, we can see that the LSTM based RNN works best in this case, giving us a 96.54%accuracy. K-Neighbors Classifier and Support Vector Classifier works show almost similar performance. Gaussian Naive Bayes Classifier performed the worst with an accuracy score of 70.38%.

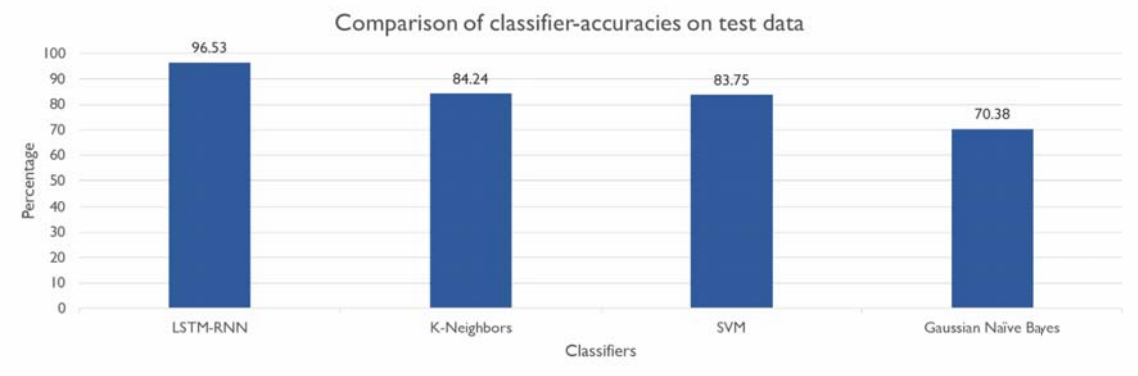


Figure 5.16: Analysis of the accuracy score of different models.

Chapter 6

Conclusion

6.1 Conclusion

This thesis introduced a Bangladeshi Sign Language (BdSL) recognition tool that acts as a text-translator of certain sign language signs. It can detect the significance of a sign-word from basic 2-D clips. There were very few works on this subject previously, particularly with BdSL. Most of the researchers of previous works relied on numerous sensors that are not available to the mass. The sensors used so far are advanced hardware such as specialized gloves or 3-dimensional camera such as Microsoft Kinect, Intel RealSense. Our technique uses an RGB camera, making it more available on most smartphones nowadays, and it works like how an ordinary text translator operates. Here, the thesis also avoided the reliance on the context, lighting conditions, body form, body-color, and clothes using the OpenPose library to concentrate on the extraction of the main points of the human skeleton, which was the most critical step before classification. Finally, the LSTM model was able to provide a satisfactory 96.54% precision. The accuracy shows that 2D cameras, which are the most available devices, will effectively provide good results for this model. So people can use their smartphones to understand movements as well.

6.2 Future Work

According to these objectives below, this recommended framework can be generalized and strengthened in the future-Ten movements, in particular, are in the current dataset. In the future, more words could be added to this dataset. To have a model that can be more reliable and generalized, the volume of data for each word or phrase can be increased. In the data collection, words that include not only hand motions but also facial expressions can be added. The precision can be improved by tuning the parameters and time steps of the network architecture. Afterward, this recognition system will concentrate on the construction of real-time sentences. The framework should be able to mark each frame with a motion that is present or not present for this to function. Depending on which the images with movements can be picked and submitted for classification. Since RNN-LSTM employs the basic RNN calculation as an intermediary candidate for the internal memory unit memory cell (state). In the LSTM RNN model, the Gated Recurrent Unit (GRU)-RNN reduces the gate signals to 2.

Bibliography

[1] M. S. Bartlett, "Periodogram analysis and continuous spectra," *Biometrika*, vol. 37, no. 1/2, pp. 1–16, 1950.

[2] O. Sachs, "Seeing voices: A journey into the world of the deaf," *Berkeley and Los Angeles*, 1989.

[3] M. L. Sternberg, *American sign language concise dictionary*. Perennial Library, 1990.

[4] C. Uras and A. Verri, "On the recognition of the alphabet of the sign language through size functions," in *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3-Conference C: Signal Processing (Cat. No. 94CH3440-5)*, IEEE, vol. 2, 1994, pp. 334–338.

[5] C.-L. Huang, W.-Y. Huang, and C.-C. Lien, "Sign language recognition using 3-d hopfield neural network," in *Proceedings., International Conference on Image Processing*, IEEE, vol. 2, 1995, pp. 611–614.

[6] T. Yamaguchi, M. Yoshihara, M. Akiba, M. Kuga, N. Kanazawa, and K. Kamata, "Japanese sign language recognition system using information infrastructure," in *Proceedings of 1995 IEEE International Conference on Fuzzy Systems.*, IEEE, vol. 5, 1995, pp. 65–66.

[7] J.-S. Kim, W. Jang, and Z. Bien, "A dynamic gesture recognition system for the korean sign language (ksl)," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 2, pp. 354–359, 1996.

[8] L. Yoshino, T. Kawashima, and Y. Aoki, "Recognition of japanese sign language from image sequence using color combination," in *Proceedings of 3rd IEEE International Conference on Image Processing*, IEEE, vol. 3, 1996, pp. 511–514.

[9] T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden markov models," in *Motion-based recognition*, Springer, 1997, pp. 227–243.

[10] R.-H. Liang and M. Ouhyoung, "A real-time continuous gesture recognition system for sign language," in *Proceedings third IEEE international conference on automatic face and gesture recognition*, IEEE, 1998, pp. 558–567.

[11] G. Montgomery and A. F. Dimmock, "Venerable legacy: Saint bede and the anglo-celtic contribution to literary, numerical and manual language," Scottish Workshop Publications Edenborough, 1998.

[12] S.-H. Shin, S.-W. Kim, and Y. Aoki, "A structural learning of mlp classifiers using pfsga and its application to korean sign language recognition," in *Proceedings of IEEE. IEEE Region 10 Conference. TENCON 99.'Multimedia Technology for Asia-Pacific Information Infrastructure'(Cat. No. 99CH37030)*, IEEE, vol. 1, 1999, pp. 190–193.

[13] C. Wang, W. Gao, and J. Ma, "An approach to automatically extracting the basic units in chinese sign language recognition," in *WCC 2000-ICSP 2000. 2000 5th International Conference on Signal Processing Proceedings. 16th World Computer Congress 2000*, IEEE, vol. 2, 2000, pp. 855–858.

[14] J. Wu, W. Gao, J. Liang, and X. Wu, "A greedy clustering algorithm along the time axis for chinese language recognition," in *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat. No. 01CH37236)*, IEEE, vol. 4, 2001, pp. 2439–2444.

[15] S. D. Fischer and H. Van der Hulst, "Sign language structures," *Deaf studies language and education*, pp. 319–331, 2003.

[16] Y. Hamada, N. Shimada, and Y. Shirai, "Hand shape estimation under complex backgrounds for sign language recognition," in *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, IEEE, 2004, pp. 589–594.

[17] R. J. Ruben, "Sign language: Its history and contribution to the understanding of the biological nature of language," *Acta oto-laryngologica*, vol. 125, no. 5, pp. 464–467, 2005.

[18] G. Yao, H. Yao, X. Liu, and F. Jiang, "Real time large vocabulary continuous sign language recognition based on op/viterbi algorithm," in *18th International Conference on Pattern Recognition (ICPR'06)*, IEEE, vol. 3, 2006, pp. 312–315.

[19] A. Graves, M. Liwicki, H. Bunke, J. Schmidhuber, and S. Fernández, "Unconstrained on-line handwriting recognition with recurrent neural networks," in *Advances in neural information processing systems*, 2008, pp. 577–584.

[20] X. Wang, F. Jiang, and H. Yao, "Dtw/isodata algorithm and multilayer architecture in sign language recognition with large vocabulary," in *2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, IEEE, 2008, pp. 1399–1402.

[21] S. Begum and M. Hasanuzzaman, "Computer vision-based bangladeshi sign language recognition system," in *2009 12th International Conference on Computers and Information Technology*, IEEE, 2009, pp. 414–419.

[22] S.-J. Wang, D.-C. Zhang, C.-C. Jia, N. Zhang, C.-G. Zhou, and L.-B. Zhang, "A sign language recognition based on tensor," in *2010 Second International Conference on Multimedia and Information Technology*, IEEE, vol. 2, 2010, pp. 192–195.

[23] S. Lang, M. Block-Berlitz, and R. Rojas, "Sign language recognition with kinect," *Bachelor, Institut für Informatik, Freie Universitä t Berlin*, 2011.

[24] N. N. Choudhury, G. Kayas, *et al.*, "Automatic recognition of bangla sign language," Ph.D. dissertation, BRAC University, 2012.

[25] D. Kaushik Deb, M. I. Khan, H. P. Mony, and S. Chowdhury, "Two-handed sign language recognition for bangla character using normalized cross correlation," *Global Journal of Computer Science and Technology*, 2012.

[26] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International conference on machine learning*, 2014, pp. 1764–1772.

[27] I. Hussain, A. K. Talukdar, and K. K. Sarma, "Hand gesture recognition system with real-time palm tracking," in *2014 Annual IEEE India Conference (INDICON)*, IEEE, 2014, pp. 1–6.

[28] M. Mohandes, S. Aliyu, and M. Deriche, "Arabic sign language recognition using the leap motion controller," in *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, IEEE, 2014, pp. 960–965.

[29] M. A. Rahaman, M. Jasim, M. H. Ali, and M. Hasanuzzaman, "Real-time computer vision-based bengali sign language recognition," in *2014 17th International Conference on Computer and Information Technology (ICCIT)*, IEEE, 2014, pp. 192–197.

[30] S. N. Sawant and M. Kumbhar, "Real time sign language recognition using pca," in *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, IEEE, 2014, pp. 1412–1415.

[31] S. K. Hasan and M. Ahmad, "A new approach of sign language recognition system for bilingual users," in *2015 International Conference on Electrical & Electronic Engineering (ICEEE)*, IEEE, 2015, pp. 33–36.

[32] A. M. Jarman, S. Arshad, N. Alam, and M. J. Islam, "An automated bengali sign language recognition system based on fingertip finder algorithm," *International journal of electronics & informatics*, vol. 4, no. 1, pp. 1–10, 2015.

[33] C. Olah, *Understanding lstm networks*, 2015. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

[34] F. Yasir, P. C. Prasad, A. Alsadoon, and A. Elchouemi, "Sift based approach on bangla sign language recognition," in *2015 IEEE 8th International Workshop on Computational Intelligence and Applications (IWCIA)*, IEEE, 2015, pp. 35–39.

[35] D. Naglot and M. Kulkarni, "Real time sign language recognition using the leap motion controller," in *2016 International Conference on Inventive Computation Technologies (ICICT)*, IEEE, vol. 3, 2016, pp. 1–5.

[36] N. S. Soni, M. Nagmode, and R. Komati, "Online hand gesture recognition & classification for deaf & dumb," in *2016 International Conference on Inventive Computation Technologies (ICICT)*, IEEE, vol. 3, 2016, pp. 1–4.

[37] M. A. Uddin and S. A. Chowdhury, "Hand sign language recognition for bangla alphabet using support vector machine," in *2016 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, IEEE, 2016, pp. 1–4.

[38] K. R. Ahmed, S. M. Mumu, and F. T. Z. Shuvra, "Basic bangla sign language recognition and sentence building using microsoft kinect," Ph.D. dissertation, BRAC University, 2017.

[39] U. Santa, F. Tazreen, and S. A. Chowdhury, "Bangladeshi hand sign language recognition from video," in *2017 20th International Conference of Computer and Information Technology (ICCIT)*, IEEE, 2017, pp. 1–4.

[40] J. Uddin, F. N. Arko, N. Tabassum, T. R. Trisha, and F. Ahmed, "Bangla sign language interpretation using bag of features and support vector machine," in *2017 3rd International Conference on Electrical Information and Communication Technology (EICT)*, IEEE, 2017, pp. 1–4.

[41] S. Sarker and M. M. Hoque, "An intelligent system for conversion of bangla sign language into speech," in *2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, IEEE, 2018, pp. 513–518.

[42] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, pp. 172–186, 2019.

[43] F. M. Noori, B. Wallace, M. Z. Uddin, and J. Torresen, "A robust human activity recognition approach using openpose, motion features, and deep recurrent neural network," in *Scandinavian Conference on Image Analysis*, Springer, 2019, pp. 299–310.

[44] A. M. Rafi, N. Nawal, N. S. N. Bayev, L. Nima, C. Shahnaz, and S. A. Fattah, "Image-based bengali sign language alphabet recognition for deaf and dumb community," in *2019 IEEE Global Humanitarian Technology Conference (GHTC)*, IEEE, 2019, pp. 1–7.

[45] P. P. Urmee, M. A. Al Mashud, J. Akter, A. S. M. M. Jameel, and S. Islam, "Real-time bangla sign language detection using xception model with augmented dataset," in *2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, IEEE, 2019, pp. 1–5.

[46] S. Hossain, D. Sarma, T. Mittra, M. N. Alam, I. Saha, and F. T. Johora, "Bengali hand sign gestures recognition using convolutional neural network," in *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, IEEE, 2020, pp. 636–641.

[47] F. M. Abujalala, H. R. Abulifa, and A. M. Abushaala, "Training a support vector machine classifier,"