

# Hand Gesture Recognition Using Ensemble Method

by

Sahib Kowsar

19301096

Mahzabin Chowdhury

19301084

MD Safin Mahmud

19301231

Shahbaj Shafin Haque

19101566

Asaka Akther Shifa

19301069

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 and Engineering (C.S.E)



Inspiring Excellence

Department of Computer Science and Engineering

School of Data and Sciences

Brac University

May 2023

© 2023. Brac University  
All rights reserved.

## Declaration

It is hereby declared that

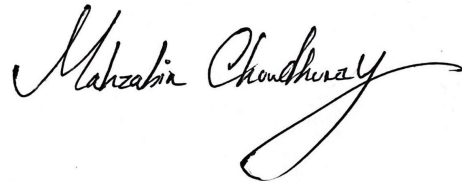
1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material 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.

**Student's Full Name & Signature:**



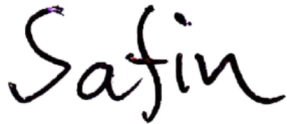
---

Sahib Kowsar  
19301096



---

Mahzabin Chowdhury  
19301084



---

MD Safin Mahmud  
19301231



---

Shahbaj Shafin Haque  
19101566



---

Asaka Akther Shifa  
19301069

# Approval

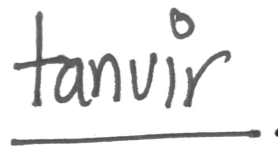
The thesis/project titled ”**Hand gesture to real time character interpretation system through computer vision based approach**” submitted by

1. Sahib Kowsar(19301096)
2. Mahzabin Chowdhury(19301084)
3. MD Safin Mahmud(19301231)
4. Shahbaj Shafin Haque(19101566)
5. Asaka Akther Shifa(19301069)

Of Spring, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on May, 2023.

## Examining Committee:


Primary Supervisor:  
(Member)



---

Mr. Tanvir Ahmed  
Lecturer  
Department of Computer Science and Engineering  
Brac University

Co-Supervisor:  
(Member)



---

Mr. Nabuat Zaman Nahim  
Lecturer  
Department of Computer Science and Engineering  
Brac University

Program Coordinator:  
(Member)

---

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

Head of Department:  
(Chair)

---

Sadia Hamid Kazi, PhD  
Chairperson and Associate Professor  
Department of Computer Science and Engineering  
Brac University

# Abstract

Even though things have improved much more over the last century in terms of communication, there still is a glaring amount of communication gap between the hearing majority and the deaf community due to the lack of resources in the field. Real time hand gesture recognition development tries to tear down this communication barrier and open a new common ground for everyone and hand gesture recognition plays a vital role in human-computer interaction as well. There are several ideas on how to build a model to properly recognize sign languages. The models differ based on the computation time it takes, the algorithms used and if it can be used in real time or not. In this work we take a thorough analysis of real-time hand gesture recognition models and proposes a pipeline-based approach to select the best-performing model as the final output. We chose to work with four datasets that are being used here for comparison, SLR500, AUTSL-226, WLASL2000 and WLASL100. The goal here is to find a way to overcome the limitations of data scarcity in the field along with the imbalance in classification problems. We work with video inputs to run them through different modalities simultaneously through a set of pipelines to produce outputs which would then be used in getting the final classification result by using the core idea of generating the final output of the ensemble technique. Various data pre-processing techniques are used such as regularization, histogram equalization etc. to minimize the varying skin tone bias to make it a more inclusive model for better classification and improved accuracy score. The existing models have no way to deal with biases encountered in sign language detection and we take various different approaches to overcome such limitations. In general pristine cases for around 500 classes the model performs 96.32 percent in terms of top-1 accuracy.

## Keywords:

Pattern Matching, Feature Extraction, SSTCN, SL-GCN, Pipeline, Transfer Learning, Histogram Matching.

## **Acknowledgement**

To our supervisor Mr. Tanvir Ahmed sir for his kind support and advice in our work. He helped us whenever we needed help.

And finally to our parents without their throughout support it may not be possible. With their kind support and prayer we are now on the verge of our graduation.

# Table of Contents

<b>Declaration</b>	<b>i</b>
<b>Approval</b>	<b>ii</b>
<b>Abstract</b>	<b>iv</b>
<b>Acknowledgment</b>	<b>v</b>
<b>Table of Contents</b>	<b>vi</b>
<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>ix</b>
<b>Nomenclature</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Real Time Sign Language Detection . . . . .	1
1.2 History of Real Time Hand Gesture Recognition System . . . . .	1
1.3 Thoughts Behind Our Model . . . . .	2
1.4 Research Objectives and Motives . . . . .	2
1.5 Problem Statement . . . . .	3
1.6 Our Contributions . . . . .	5
<b>2 Literature Review</b>	<b>6</b>
2.1 Wearable sensor based approach . . . . .	6
2.2 Computer vision based approach . . . . .	6
2.3 Depth based approach . . . . .	8
2.4 Deep learning based approach . . . . .	9
<b>3 Dataset</b>	<b>10</b>
3.1 Dataset Description . . . . .	10
<b>4 Comparative Analysis</b>	<b>11</b>
4.1 Visual representation of papers . . . . .	12
<b>5 Architecture of Proposed System and Methodological Analysis</b>	<b>14</b>
5.1 Pipeline: . . . . .	14
5.2 Graph Reduction and Convolution: . . . . .	15
5.3 Multistream SL-GCN: . . . . .	16

5.4	SL-GCN Block: . . . . .	16
5.5	SSTCN: . . . . .	16
5.6	2D Convolution with Recurrent Neural Network: . . . . .	17
5.7	3D Convolution Network: . . . . .	18
5.8	Pose-based Recurrent Neural Network: . . . . .	18
5.9	Pose-based Temporal Graph Neural Network: . . . . .	19
5.10	Ensemble: . . . . .	19
<b>6</b>	<b>Result Evaluation</b>	<b>20</b>
6.1	Results: . . . . .	20
6.2	Checking for Skin-tone Biases in the Models: . . . . .	23
6.3	Improvement: . . . . .	23
<b>7</b>	<b>Epilogue</b>	<b>25</b>
7.1	Insufficient Resources: . . . . .	25
7.2	Conclusion: . . . . .	25
	<b>Bibliography</b>	<b>28</b>



# List of Figures

4.1	Architecture from study [17]	12
4.2	Architecture from study [12]	12
4.3	Architecture from study [5]	13
5.1	RGB Ensemble [14]	14
5.2	Multi-streamed modality [14]	16
5.3	Connected Networks [14]	17
5.4	Multilayered Pose TGCN [15]	18
6.1	Model Results Based on AUTSL226 Dataset	20
6.2	Model Results Based on SLR500 Dataset	21
6.3	Model Results Based on WLASL100 Dataset	21
6.4	Model Results Based on WLASL2000 Dataset	22
6.5	Model Results Based on WLASL100 Dataset Without Bias	24
6.6	Model Results Based on WLASL Dataset Without Bias	24

# List of Tables

4.1 Comparative Study of Models . . . . .	11
---	----

# Nomenclature

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

*ANN* Artificial Neural Network

*CNN* Convolutional Neural Network

*CSL* Context-sensitive Language

*GRU* Gated Recurrent Unit

*HMM* Hidden Markov Model

*IMUL* Inertial Measurement Unit

*ISLR* Image-based Sign Language Recognition

*KL – HMM* Kullback-Leibler divergence Hidden Markov Model

*LBP* Local Binary patterns

*LDA* Latent Dirichlet allocation

*LSTM* Long Short Term Memory

*NN* Nearest Neighbor

*PCA* (Principal Component Analysis

*PCA* Principal Component Analysis

*RNN* Recurrent Neural Network

*SRL* Sign Language Recognition

*SSTCN* Separable Spatio-Temporal Convolutional Network

*SVM* Support Vector Machine

*TGCN* Temporal Graph Convolution Network

# Chapter 1

## Introduction

### 1.1 Real Time Sign Language Detection

Sign language is the communication medium between the hearing community and the deaf community or within the deaf community. It is a full-edged complex language that is a subset of human gesture communication. There are a lot of people around the world whom are unable to speak or hear, so to stop them from falling behind from the rest of the world sign language was developed. Both hearing people and people whom are unable to speak or hear needs to know sign language otherwise there is always be a communication gap between them. But as it is a very complex and hard to grasp language, many are unable to understand it. Thus it becomes very challenging for one side to communicate with the other and vice-versa. Moreover, similar to spoken language, there is no single sign language that is used worldwide. It's an extraordinary language that differs from culture to culture and place to place. There are a lot of sign languages that are being used around the world currently. For example, American Sign Language (ASL), British Sign Language (BSL), Turkish Sign Language, Chinese Sign Language, Taiwan Sign Language etc. Each of the aforementioned sign language has its own set of grammar and rules. In case one is unable to understand or learn sign language, an interpreter would be very useful in this case. Thus, it is desirable to make a system, within devices we use on a daily basis, to understand sign language so that it can serve as an interpreter but for it to work properly and efficiently it has work on real time with as little amount of delay as possible in terms recognizing a sign correctly.

### 1.2 History of Real Time Hand Gesture Recognition System

During the early approaches, hand gesture recognition models were mainly based on feature extraction and template matching which were definitely limited to simple gesture recognition under certain type of lighting and background. [13] Then as time went by and the concepts of deep based models starting to arise, Microsoft Kinect's built-in hand tracking and recognition system came to be which worked using the combination of depth and skeletal tracking. But the main revolution took place after the big breakthrough and resurgence of deep learning in the 2010s. CNN and RNN were being used for learning spatial and temporal features from hand

gesture data and these deep learning models increased the recognition accuracy to a very significant amount. [9] Afterwards, real time optimization came to focus as computer power and hardware accelerated processing capabilities expanded. [7]

### 1.3 Thoughts Behind Our Model

There are lot of approaches for creating a model that can accurately recognize sign language. The algorithms used, the amount of computation time required, and whether or not real-time use is possible all affect how the models differ. Our goal is to create a model that can operate in real time, which means that it will only require a very short amount of time for the system to recognize a sign gesture meaning that the computation cost will be very low. A video that contains hand gestures will be divided into segments, the segments will be based on the beginning and end of a hand gesture. Furthermore, as this is a classification problem, we will need functions to extract features from the segmented parts and use pattern matching to identify the signs. We will run the segmented parts through five different pipelines containing different models and choose the best one out of all the generated outputs for a single hand gesture [14]. Additionally, we are aiming for a combination of eccentric approaches during the data-preprocessing such as histogram equalization, regularization etc. [5] to reduce the existing skin tone bias so that It can give a more improved result even for a data-set with huge diversity in terms of skin tone. The machine should be trained in such a way that it can interact with the users who are using sign language to communicate. They should be able to recognize and learn signs and its meaning. Moreover, computer vision researchers has long been interested in this field of automatic hand gesture recognition. This automatic gesture and sign language recognition offers to enhance the communication capabilities of people with hearing and speech impairment. We have used a variety of techniques, such as RNN that will help to detect large motions so that it is able to capture the moving thing better and then use CNN on it for the finer changes, artificial neural networks (ANN), transfer learning to gain advantage in solving related task, CSL and Hidden Markov Models (HMMs) to recognize hand gestures. [14] [9]

### 1.4 Research Objectives and Motives

Our research on Sign Language Translation(SLT) and Sign Language Recognition is directed at refining a real time SLT system using a combination of Deep learning based algorithms to create separate models which will then run on pipelines (simultaneously) that are going produce output and based on them the best one will be chosen for the final result. Moreover, some eccentric methods have been approached to reduce the computational time while improving the result meaning increasing the test accuracy score all while eliminating the skin tone bias. Our main focus here is to decrease the computational cost so that we can get a much better, more specifically faster recognition and translation of the sign language from a video source all while keeping in mind about the off-angle of the hand gesture source. Through rigorous optimizations such as prepossessing the input into smaller frame, using histogram equalization on the input data, hand detection using skeleton method, transfer learning, extracting features SSTCN and optimizing it using Fuzzy logic

and finally using K-Ary tree for hand gesture recognition, our goal is to develop a model capable of real time SLT using video data from easily accessible devices such as smartphone cameras, laptop cameras etc.

### **The Primary Objectives of this research paper are:**

- To get an extensive knowledge on different algorithms and how they can ensure a lower computational cost while keeping the accuracy of the system to a moderately high level.
- To develop a model which works around the generally encountered data scarcity that exists in the field of sign language and overcome data imbalance and lack of variance in classification issues.
- Aim to build a model that will include different kinds of input data preprocessing techniques such as histogram matching, pixel median conversion etc which will help the mode to generalize well across different users, skin tones, hand shapes and appearance.

## **1.5 Problem Statement**

Sign language is a way of manual communication among people who are deaf and unable to convey their thoughts aurally. However, sign language is not universal. People from different geographical locations use different renditions of it where the gestures are represented in a linguistic manner and each gesture is commonly referred to as a sign. As we know English is one of the most widely spoken languages in the world, which is around 1.5 billion people. Some of them are native speakers and some of them have English as their second language. Naturally, to cover the most amount of ground it is the obvious choice to work with English sign gestures. Automated sign language recognition and interpretation breaks down the barrier between people with hearing disabilities and the hearing majority. There are currently more than 70 million people in the world who use sign language as the mean of communication. Through sign language, they are able to study, work, access resources, and take part in their communities. But it is tough for the people who uses sign language for communication to interact with their native people as most of them does not have sufficient knowledge of sign language. Not only that, it reflects on their career identity, earning power but also their personal and day to day life. Beside this, Some families rely on the HOH (hard of hearing) or the deaf member in the family to read lips, but this is incredibly challenging and frequently leads to an incorrect understanding of what has been said. It can be said that there is a massive communication gap between hearing impaired people and non signer communities, to reduce this gap, sign language interpreters are introduced. Unfortunately a large number of the population are not connected with certified sign language interpreters. So, there is a need for computer vision based recognition of sign language without the need of any interpreters. It will assist the hard of hearing to interact at various levels in society by bridging the communication gap. Hand gesture to real time character interpretation systems through computer vision based approach should be able

to recognize and learn the language and signs used by sign users in order to communicate with them. For this computer vision based approach to work perfectly, the recognition model/s should be able to run on real time with as little computational time as possible all while having a high accuracy rate. With the potential to improve communication for the deaf and speech-impaired, automatic gesture and sign language recognition has long drawn the attention of computer vision researchers. For the purpose of hand gesture recognition, researchers have employed a number of methods, including CNN (convolutional neural network), ANN (artificial neural networks) , RNN, CSL, HMMs. [15] [16]

However, there are major pitfalls for the conventional methods used by researchers. [8] One is a massive computational power requirement and the other is requiring expensive or non easily accessible hardware like a Microsoft Kinect or Dedicated sensors etc. Which pose two problems primarily. One of them being the accessibility to such enthusiast level equipment purely from a financial and availability standpoint. It circles back to the original problem of the lack of proficient sign language interpreters. The other problem is the raw computational power required in terms of the computer vision based alternatives to use on regular available hand-held devices in peoples hands. [8] Hence, there needs to be a way to overcome such issues without sacrificing accuracy and accessibility through a different approach without relying on the traditional models which mostly address static environments or objects in interest.

Furthermore, another major issue with hand gesture recognition models is that the available data-sets are not balanced meaning the distribution of samples across different classes or categories is highly skewed. The imbalance can be problematic as it can lead to biased model performance and reduced accuracy. Moreover, as the language itself is not that well explored and varies from place to place, there exists some wrong labeling in large data-sets which also hampers in bringing out the best accuracy score from a given data-set. Furthermore, collecting and annotating large amounts of hand gesture data is very time consuming, expensive and challenging due to the need for precise and consistent labeling. So, the labeled data available for training the hand gesture models is limited. This data scarcity hinders the development and performance of accurate models. Thus, both unbalanced data and data scarcity pose quite the challenge for hand gesture recognition models.

Therefore, the question that this research is trying to answer is

***How to optimize a computer vision based sign language recognition system without requiring incredible amount of raw computational power and easily accessible hardware?***

This research will answer the question stated above by processing the videos through a pipeline of different techniques and with the use of a pre-emptive predictive gesture recognition model.

## 1.6 Our Contributions

The previous studies the real time hand gesture recognition models have been improved significantly but there are some discrepancy in properly recognizing while dealing with data that has a variety of skin-tone.

**We make the following contributions to this study based on our research:.**

- A hand gesture recognition model that will recognise hand gestures in real time while minimizing the computational cost as much as possible.
- Making sure that the racial discrepancy that occurs because of the skin tone differences of the hand in the input images during recognition of the gesture does not take place.
- We propose this approach as it is a more inclusive model that has no boundaries in terms of skin tone.



# Chapter 2

## Literature Review

The concept of image recognition is a significant and ongoing research area in computer vision. Image recognition refers to classifying or identifying an object from an image or video feed. This idea has been used in many applications like medical image diagnosis, facial recognition, visual search, extracting text from image or optical character recognition etc. Sign language or hand gesture recognition is also a kind of image recognition which can be achieved with several approaches.

In sign language recognition we detect and interpret the hand gestures or signs to convert them into a meaningful text. This enables us to mitigate the communication gap with people who use sign language as their primary means of communication.

### 2.1 Wearable sensor based approach

Computer vision techniques can be used to analyze and process video sequences of sign language gestures. These approaches involve extracting relevant features from the video frames, such as hand shape, hand motion, or hand pose, using techniques like background subtraction, contour detection, or optical flow. Machine learning algorithms, such as Hidden Markov Models (HMMs), Support Vector Machines (SVMs), or Recurrent Neural Networks (RNNs), can then be trained to recognize and classify the extracted features into sign language gestures.

It has been found that using Euler angle (EULA) and quaternion (QUA) from IMU to represent complex hand rotation. (acceleration and angular velocity) then attention based encoder decoder model with multichannel CNN gives an error rate of 10.8% on continuous sentence recognition when implemented on smartphones [17]. The encoder in an attention-based encoder-decoder is a bidirectional long short-term memory (LSTM) network that receives the complete phrase as input and outputs a probability matrix for each word. The decoder may choose which portion of input should be given greater weight thanks to the attention mechanism.

### 2.2 Computer vision based approach

The idea of hand gesture recognition through a computer vision based approach in order to use in real time is not new. But, there are several limitations to it

and to overcome these limitations, time to time different methodologies have been proposed/used. Here, One solution is more optimum than the other and this optimization is a never ending cycle. Computer vision approaches involve extracting relevant features from the video frames, such as hand shape, hand motion, or hand pose etc.

Vision based approach gives users a degree of freedom which decreases the chance of false data collection. In [5] the process used to do hand gesture recognition does not rely on pattern recognition as it takes more computation time than that's needed to use in real time. Thus, they went for more of an eccentric approach by counting active fingers 7 which can be achieved by using the euclidean distance between the centroid point and the furthest point as the radius of a circle that will mask the inner parts of the circle. After that use belabel and regionprop the number of active fingers. But it worked only on RGB images by reducing its resolution to 160x120. It was able to recognize up to 6 frames per second.

For the case of [1], Salma Begum and Md. Hasanuzzamana proposed a Principal Component Analysis (PCA) based pattern matching system for recognizing 16 bangla signs. This system used skin color based segmentation for preprocessing. Moreover to make computation faster the images were scaled down to 0.40 and then turned into HSV from RGB format. For feature extraction a threshold for skin color is set in HSV. Anything outside the threshold is set to black and inside is white. This way the hand pose is separated from the background. For processing the data in [1] Turk and Pentland's trick was used to get the eigenvectors of  $AA^T$  from the eigenvectors of  $A^T A$ . Only K-eigenvectors of  $V_i$  are used to form PCA. These vectors define the eigenspace of hand images . For recognition or matching in this model, the test image is scaled and separated from background just like training data and then it is projected onto the eigenspaces. Afterwards a set of weight vectors is calculated using [3]. This test was done with 480 samples of bangla vowel signs and 800 samples of bangla numbers. While the precision rate for number signs were from 81-87%, vowels were 59-89%.

However [6] uses systems with better performance than PCA. In [6], Mahmood Jasim and Md. Hasanuzzaman tackled the challenges that were faced when separating hand gestures from the background and interpreting different gestures. Just like in [5] it also uses images rather than videos as a data-set. The main difference here is the process of recognition and the algorithms used for it. Here, the gesture features are extracted using Latent Dirichlet allocation (LDA) and Local Binary patterns (LBP). Moreover, the gesture classifications are done using the Nearest Neighbor (NN) algorithm. For pre-processing the Image containing the hand is first divided into multiple segments, gray-scaled and then resized. LDA takes scatter into account and uses clustering methods to separate classes and a projection is used to maximize class separation. While LBP checks a pixel against its neighboring pixels to determine edges and a 8 bit code is generated depending on a threshold. A histogram of LBP codes is then generated by calling an LBP pattern. The image is primarily split into 8x8 parts, and from each sector, a local histogram is formed. All of these local histograms are then accumulated together to create a final histogram which is utilized to identify each gesture. The classification of sign gestures is then performed on both models using the NN method. The model employs the euclidean norm as the dissimilarity metric to determine the closest match in terms of LDA. While the dissimilarity metric in LBP is the chi-square difference. On the Chinese

numerical gesture dataset, the mean accuracy found for the LDA-based sign language interpretation system was 92.417%, while on the Bangladeshi dataset, it was 88.55%. The LBP findings, on the other hand, were 87.13% and 85.1%, respectively. But just like spoken language, sign languages also differ from country to country. In order to overcome that language barrier a Multilingual Sign Language Recognition system was proposed [12] where Hamburg Notation System (HamNoSys) annotation is used in order to make a global classifier. In [12] hand shape and movement information were extracted from video files with the help of OnePose, DeepHand and HMM. The idea was to use HMM to get a discrete value for hand movement information. These discrete representations are called subunits. In this model feature observation was made probabilistic. They used the Kullback-Leibler divergence HMM (KL-HMM) model for this. The hand movement subunits extraction step was done separately according to each sign language leading to a stack of posterior probabilities. Where each element of the stack denoted the probabilistic features corresponding to hand movement subunits derived from sign language. The literature combinedly used hand shape subunit and hand movement subunit to obtain a recognition accuracy of 96.67% in case of language independent recognition.

## 2.3 Depth based approach

Depth sensors, such as Microsoft Kinect or Intel RealSense, can capture depth information in addition to RGB data. Depth-based approaches leverage this additional information to estimate 3D positions and motions of hands or body parts involved in sign language gestures. Depth information provides valuable cues for accurate hand tracking and gesture recognition. Techniques like depth image processing, hand skeleton modeling, or point cloud analysis can be combined with machine learning algorithms to recognize sign language gestures.

Roel Verschaeren proposed a CNN model that recognizes a set of 50 different signs in the Flemish Sign Language with a very less amount of error while using Microsoft kinect but has a limitation that it only works on a single person in a static environment [9]. Here, the main idea was that the video should be segmented according to each hand gesture and then put through algorithms to extract the features from it. This feature extraction can be done automatically with convolutional neural networks (CNN) while classification is done with Artificial neural networks (ANN) [9]. This model was tested on the data-set obtained from CLAP14 which basically has of 20 different Italian gestures that were carried out by 27 users in a dynamic environment that goes through preprocessing by creating noise free depth map and removing background using user index and median filtering. Then the video is segmented using temporal segmentation and goes through 2 CNN, each consisting of a depth of 3 layers. Using this 3D to 2D convolution and pattern matching it gets over 90% accuracy score on the testing data.

Microsoft Kinect is also used in the proposed model of [4] to recognize 239 words from CSL but here the 3D movement trajectory of the hand and a language model has been used to construct sentences which seemed to take up way too much computation time.

For continuous SLR Zafrulla et al [2] built 4-state hidden mark PV models (HHMs) and used viterbi alignment for word+sentence recognition. They used a Kinect depth

mapping camera to detect skeletal joints of hand, shoulder and elbow position and used the joint angles for HMM. Recently deep neural Networks with Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) have been used for capturing temporal relations in vision based continuous SLR. According to [11] Using the attention mechanism can improve the performance of aligning and sequence learning as well.

## 2.4 Deep learning based approach

Deep learning techniques like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have shown remarkable results in sign language recognition. CNNs can be used to learn spatial features from video frames or depth images, while RNNs, such as Long Short-Term Memory (LSTM) or Gated Recurrent Units (GRUs), can capture temporal dependencies and model sequential nature of sign language gestures. Deep learning models can be trained on large-scale sign language datasets to achieve high accuracy in recognition.

Like for providing real time sign to text conversion [10] uses a software based approach where a personal database, Deep Learning and CNN are used for increased efficiency and cost reduction. In this paper, pre-processing images are generalized and each gesture is mapped to a string, then a database is created using OpenCV. Here hand or sign detection is done using Skin Color Segmentation, Median blur and Contour detection. In the skin color segmentation part we extrapolate the hand and fill the dark spots within, then blur the image using Median blur to reduce noise in the image. The outcome from this is a hand histogram. Then using the 'find-Contours' function of OpenCV the contour with the maximum area is found. [10] classified gestures with CNN and had a prediction accuracy of 99%.

# Chapter 3

## Dataset

### 3.1 Dataset Description

For the datasets we have chosen WLASL, American Sign Language which has two variations. One with 100 (WLASL100) classes and the other with 2000 (WLASL2000) classes. This dataset is unbalanced in terms of skin tones and sign videos per class. The backgrounds and lighting conditions are also different, making it a challenging dataset to work with. To get a complementary idea of the accuracy we have also included SLR500 which is a Chinese sign language dataset containing 500 classes and AUTSL226 which has 226 classes within it. Both of these datasets have pristine backgrounds, consistent lighting conditions to gauge the best case scenario in terms of the pipeline performance.

# Chapter 4

## Comparative Analysis

The table below shows the comparison between the model, method, dataset used and the accuracy of 6 papers where SLR (Sign Language Recognition) is done.

Table 4.1: Comparative Study of Models

Ref No	Method	Architecture/ Model	Dataset	Accuracy
[17]	Armbands with IMU and multi-channel sEMG sensors	Attention-based encoder-decoder model with a multi-channel CNN	CSL dataset	89.2%
[5]	iball C12 Webcam	Row 2	Own Dataset	83.33%-100%
[1]	CCD camera	PCA based pattern matching	Own Dataset	51%-89%
[6]	Logitech 310 webcam	Linear Discriminant Analysis and Local Binary Pattern based feature extractors	Chinese Numeral Gesture Dataset, Bangladeshi Numeral Gesture Dataset	92.417%, 88.55%, 87.13%, 85.10%
[12]	Vision based	Kullback Leibler divergence-based Hidden Markov Model (KL-HMM)	SMILE Swiss German Sign Language Database, Turkish Sign Language HospiSign Database, DGS Database	96.67% ( $\pm 1.80$ ), 66.8%
[15]	Vision based	Holistic visual appearance based approach, 2D human pose based approach	WLASL2000	62.63%

Table 4.1 Contains the comparative Study of methods, architecture, data-sets and accuracy after using each model.

## 4.1 Visual representation of papers

For better understanding we made visual representations of the papers we studied and compared. We studied 7 paper and below is attached their architectural flowcharts.

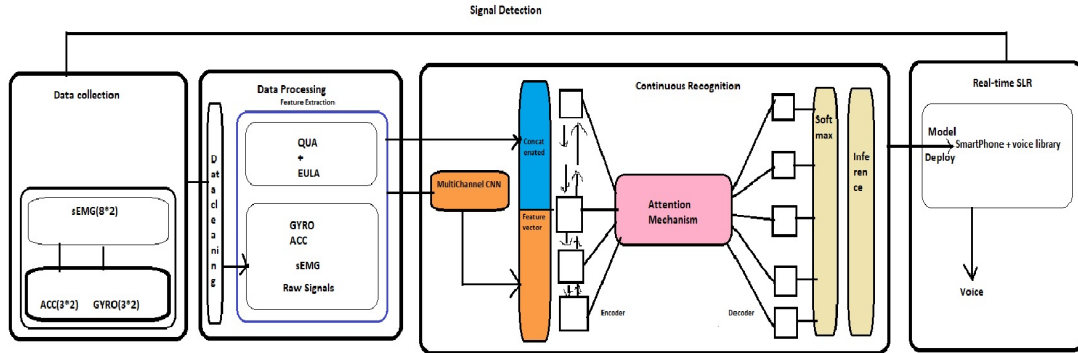


Figure 4.1: Architecture from study [17]

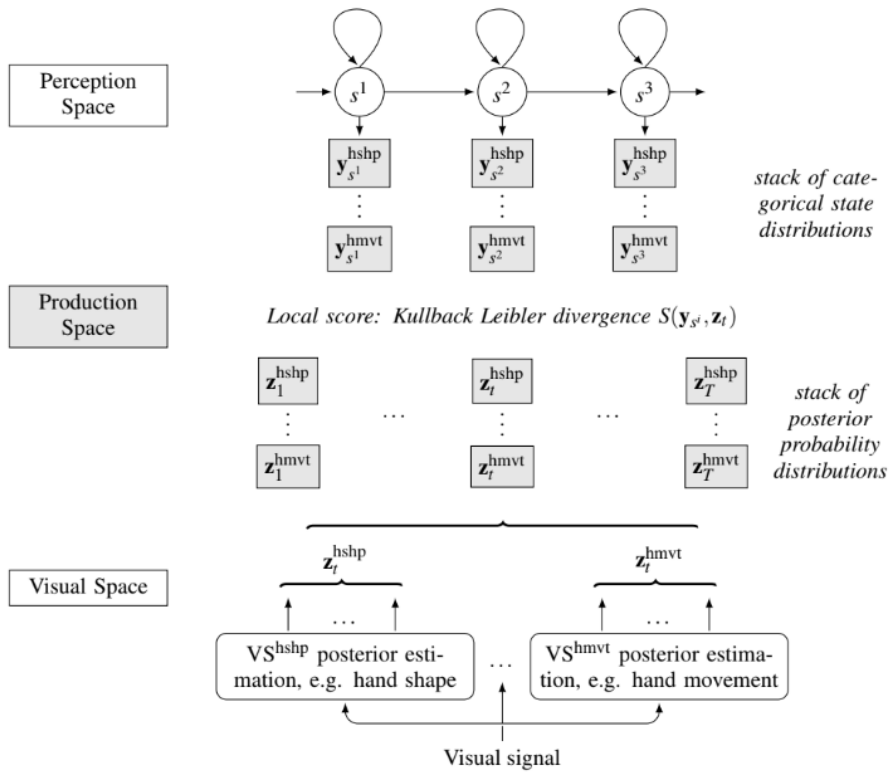


Figure 4.2: Architecture from study [12]

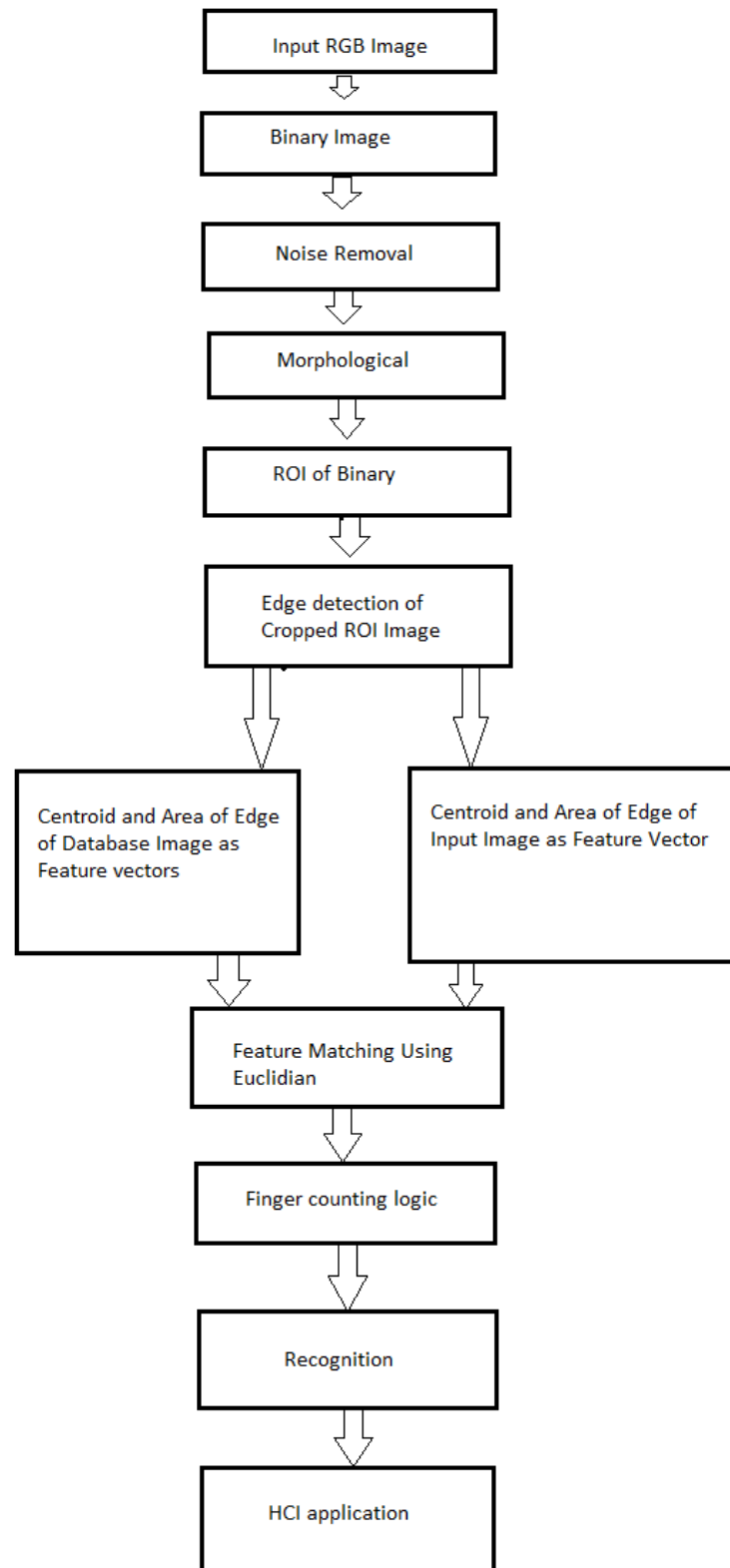


Figure 4.3: Architecture from study [5]



# Chapter 5

## Architecture of Proposed System and Methodological Analysis

### 5.1 Pipeline:

Signing is a variation of human action primarily expressed through hand gestures and because of that, action recognition and pose estimation are quite similar to it. In the following section, we present an overview of the separate modalities of our ensemble method and also evaluate the performance of deep models based on different modalities in recognizing sign language. In addition to that we will examine how usable our collected datasets are. We have approached the problem with a multi-modal ensemble method. Including different ways of extracting various features for each of the classes to get a more cohesive and holistic set of parameters to work with.

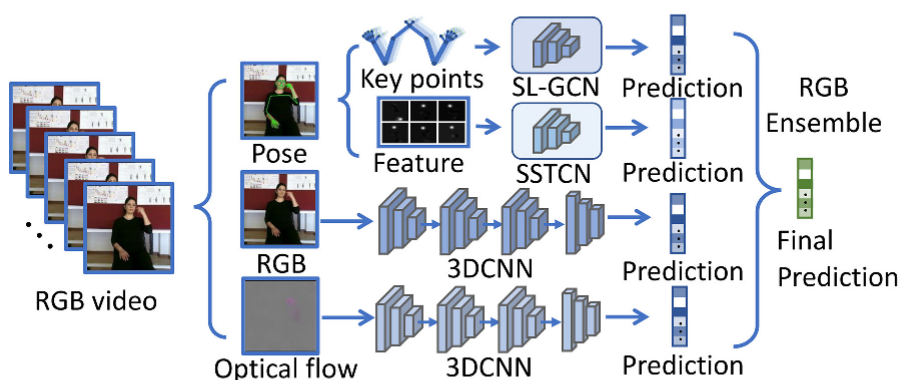


Figure 5.1: RGB Ensemble [14]

## 5.2 Graph Reduction and Convolution:

One of the more important parts in Sign Language recognition are hand gestures. Hand gestures typically include the whole body ranging from the arms to the fingers which generate a lot of key-points. Some ground truths need to be set in order to validate the detection cases. Thus we've included a pre-trained body pose estimation network for the whole body to provide 100+ key points in each of the instances. It generates a spatio-temporal graph to estimate the movements. However, these large numbers of nodes cause a lot of noise in the model. Nodes too far away from each other also can not establish a proper relationship within the model which results in low accuracy. A graph reduction is applied on the nodes to reduce the numbers of excessive, irrelevant nodes. Only the essential 10 nodes for each hand and 7 nodes for the upper body allows for the relevant information to be trained on. Graph reduction causes the models to converge faster and have higher recognition rates. We used the spatio-temporal GCN with spatial partitioning technique to model the dynamic skeletons in order to capture the pattern present in the skeleton graph. To enhance the GCN output, we used a more extensive variant of the spatial graph convolution known as decoupling graph convolution [14]. The channels of graph features are divided into groups during decoupling graph convolution, and each group's channels share a separate trainable adjacent matrix. As the output feature, the decoupling groups' convolution results are concatenated.

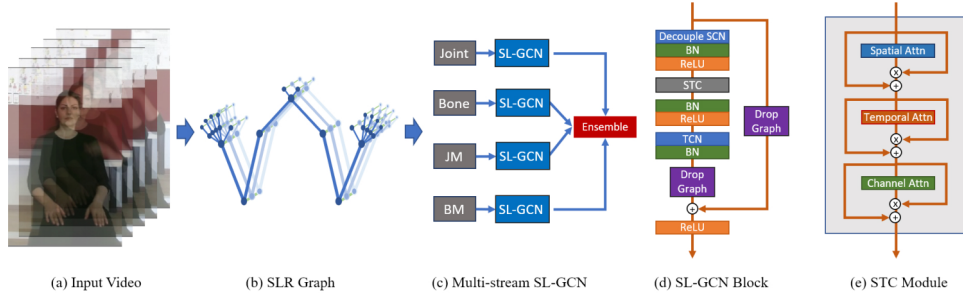


Figure 5.2: Multi-streamed modality [14]

### 5.3 Multistream SL-GCN:

From the keypoints the joints, bones, joint motion and bone motions are multi-streamed through an SL-GCN [14] network. The data are generated Using vectors pointing from source joints to their target joints present in the human body. In both joint and bone motion streams, motion data are produced by computing the difference between adjacent frames. Each stream is trained independently and its combined predicted result comes out through the weighted ensemble method.

### 5.4 SL-GCN Block:

An SL-GCN block is built with a self-attention, graph dropping, and decoupled spatial convolutional network. An STC with spatial, temporal, and channel fitting attention module, a decoupled spatial convolutional layer, a temporal convolution layer, and a Drop Graph module make up the network. With the modules set up in a cascaded arrangement and the GCN block having 10 blocks, the SCN lowers the cost. Prior to classification using a fully linked layer, global average pooling is applied on the spatio-temporal dimensions.

### 5.5 SSTCN:

Additionally, an SSTCN [14] model is introduced to identify sign language from physical characteristics. The model is fed 33 critical points from 60 frames of each film which includes 22 landmarks on hands, 2 landmarks on wrists, 2 landmarks on elbows, and 2 landmarks on shoulders, In order to downsample the features,  $24 \times 24$  max pooling is used. Using a 2D convolution layer reduces the parameters compared to a 3D one for faster convergence. There are a total of 4 stages in the network. The features are reshaped into  $60 \times 792 \times 24$  from  $60 \times 33 \times 24$  and  $1 \times 1$  convolution layers process them, meaning it only filters out the temporal information. Then the features are shuffled into 60 groups to make them go through  $3 \times 3$  convolution to extract the spatio-temporal information among the same key point features from random frames. The features are then once again shuffled to 33 groups with the same convolution specification but it only filters the spatial information from each frame. A number of  $3 \times 3$  fully connected layers are introduced to generate prediction features. Dropout layers are used to avoid overfitting the model with Swish activation

function. In a few scattered circumstances, using one-hot labels with cross-entropy loss results in overfitting. To resolve such difficulties, label smoothing methods are used. Utilizing optical flows and RGB frame modalities, a baseline for 3D CNN is produced. The most effective 3D CNN design is ResNet2+1D, which decouples spatial and temporal convolution in 3D CNNs and performs them sequentially. The performance is not improved by increasing the architectural depth, and the network becomes overfit. The identification rate for RGB frames is further improved using ResNet2+1D-18 with weights which was pre-trained on the Kinetics dataset. We swap out the ReLU activations with Swish activations, much like SSTCN.

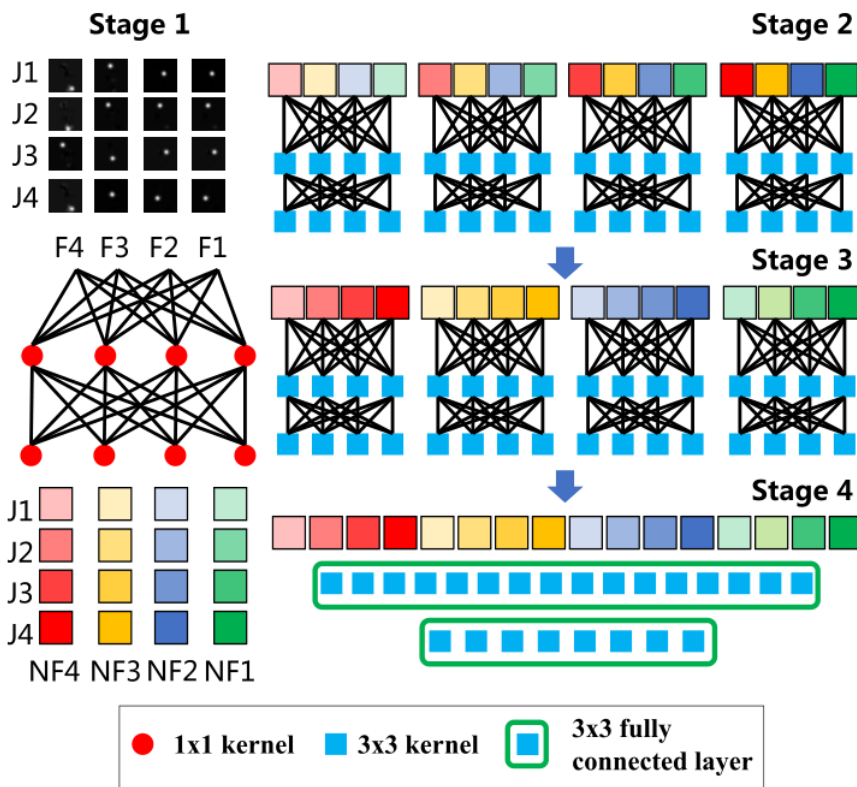


Figure 5.3: Connected Networks [14]

## 5.6 2D Convolution with Recurrent Neural Network:

2D Conv RNN is a baseline model which consists of a 2D Convolution Neural Network(VGG16 pretrained on ImageNet) to extract spatial features and a RNN(Gated Recurrent Unit) used to derive temporal features from input images and videos. Stacked recurrent layers are set to 2 in the GRU with its hidden sizes set to 64, 96, 128 and 256 for the four subsets respectively to avoid overfitting the training set. During training, up to 50 consecutive frames are randomly selected from each video, and cross-entropy loss is applied to the outputs at all time steps as well as the output feature obtained from average pooling of all the output features. In the testing phase, all the frames in the video are taken into consideration and predictions are made based on the average pooling of output features.

## 5.7 3D Convolution Network:

3D convolutional networks have the ability to capture both the overall representation of each frame and the temporal relationships between frames in a hierarchical manner.

Carreira et al. [16] have proposed a method to convert 2D filters from the Inception network into well-initialized 3D filters by inflating them. These 3D filters are then fine-tuned on the Kinetics dataset to improve their ability to capture spatial-temporal information in videos.

Inception 3D(I3D) network architecture, based on the work of Carreira et al., is used as the second baseline model for image appearance. The architecture is adapted to the specific task, focusing on the hand shapes, orientations, and arm movements in sign language. The original I3D network is trained on ImageNet and fine-tuned on the Kinetics-400 dataset. To suit the WLASL subsets with varying class numbers, only the last classification layer is modified accordingly.

## 5.8 Pose-based Recurrent Neural Network:

Pose-based approaches primarily rely on Recurrent Neural Networks (RNNs) to analyze human motions by modeling pose sequences. The RNN is used to record the temporal sequential information of posture movements, and after that sign the output representation from the RNN is used for sign recognition. To implement this approach, OpenPose is utilized to extract 55 2D key points of the body and hands from each frame of the WLASL dataset. 13 joints in the upper torso and 21 joints in the left and right hands make up these critical locations/ key points. Each joint's 2D coordinates are added together to create the input feature, which is then fed into a two-layer, stacked GRU. The hidden sizes of the GRU are empirically optimized as 64, 64, 128, and 128 for the four subsets, respectively. 50 consecutive frames are randomly chosen from the input video for training, following a similar procedure to the training and testing methods outlined in the 2D Conv RNN section. In training, cross-entropy loss is used. All of the frames in a movie are used for categorization during testing.

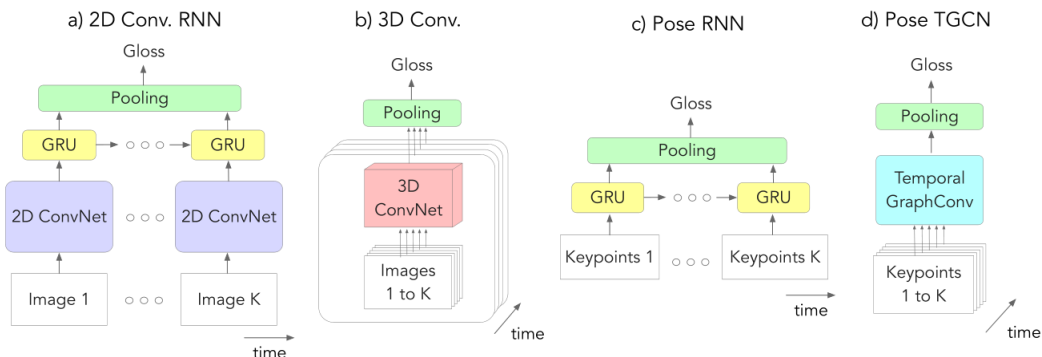


Figure 5.4: Multilayered Pose TGCN [15]

## 5.9 Pose-based Temporal Graph Neural Network:

Li et al. [16] presented a pose based approach for Image-based Sign Language Recognition (ISLR) using Temporal Graph Convolution Networks (TCGN). They proposed a graph network architecture that captures the spatial and temporal dependencies of a pose sequence represented by concatenated 2D keypoint coordinates.

In this approach, the human body is treated as a fully-connected graph with vertices representing body keypoints and the edges are represented by a weighted adjacency matrix. Through this, the model can learn dependencies among the joints through a graph network. The TCGN consists of multiple residual graph convolution blocks which take the average pooling result along the temporal dimension to obtain a feature representation of pose trajectories. The classification is performed using a softmax layer followed by an average pooling layer.

## 5.10 Ensemble:

A simple ensemble method is introduced to ensemble all of the modalities. Every output before the softmax layer is saved and assigned weights to each of the modalities based on the validation set. The sum is then predicted as the final score. Process TVL1 algorithm is used to obtain optical flow features implemented with OpenCV and CUDA. The flow maps of the spatial dimension are concatenated in channel dimension. RGB and optical flow frames are resized to 256x256 owing to the key points from pose estimation. It is applied to the rest of the modalities as well. Training is done using sampling 32 consecutive frames from each video randomly. Data is augmented using random sampling, mirroring, rotation, shifting and jittering to avoid overfitting. The learning rate during training at the beginning is 1e-3 with weight decay 1e-4. At epoch 70 the learning rate is 1e-4 with weight decay being 0. At epoch 150 the learning rate is 1e-5. A total of 230 epoch is trained.

# Chapter 6

## Result Evaluation

### 6.1 Results:

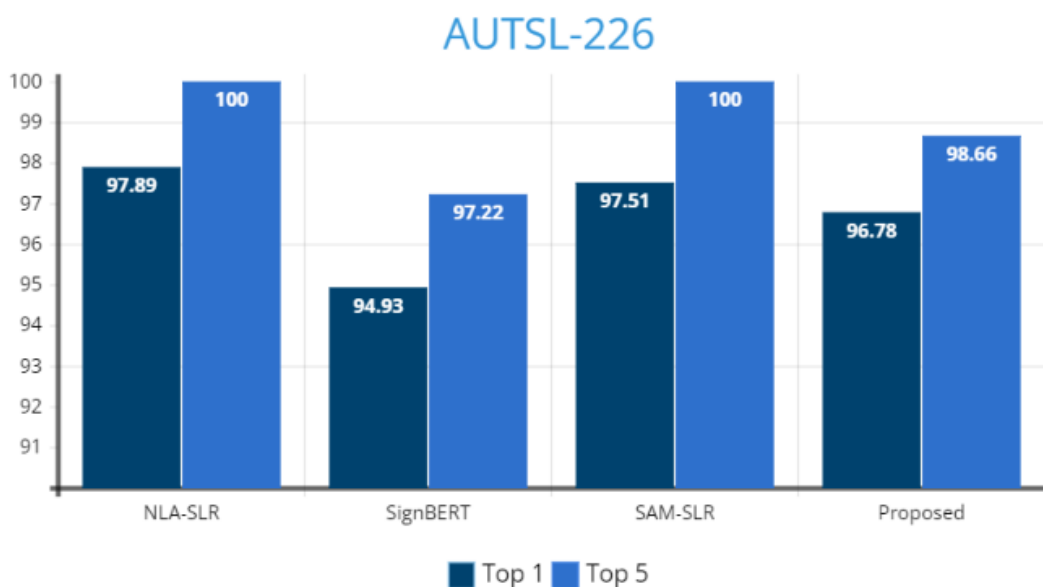


Figure 6.1: Model Results Based on AUTSL226 Dataset

Evaluating the model with pristine testing and training datasets such as SLR-500 and AUTSL-226 produces favorable results. NLA-SLR and SAM-SLR both produce top-1 accuracy at 97.89% and 97.51% respectively, while our proposed pipeline trails at 96.78% accuracy. Which is better than SignBERT's 94.93% accuracy. Similarly with SLR-500, NLA-SLR and SAM-SLR both edge out with 98.10% and 98.98% top-1 accuracy with our proposed model coming at 96.32% accuracy ahead of SignBERT at 95.56%. The high scores in accuracy is owed to good lighting conditions in the datasets with clear backgrounds which allows for better separation and the datasets have no variance in terms of color, jitter and tones.

With WLASL-2000 having that many classes and variances in the dataset causes the overall accuracy to drop across the board. The proposed model settles at 54.55% for top-1 accuracy while NLA-SLR and SAM-SLR settle at 58.16% and 57.13% respectively. WLASL-100 with its reduced classes and more balanced instances for each class helps with the recognition issue, to better teach the network of the differences

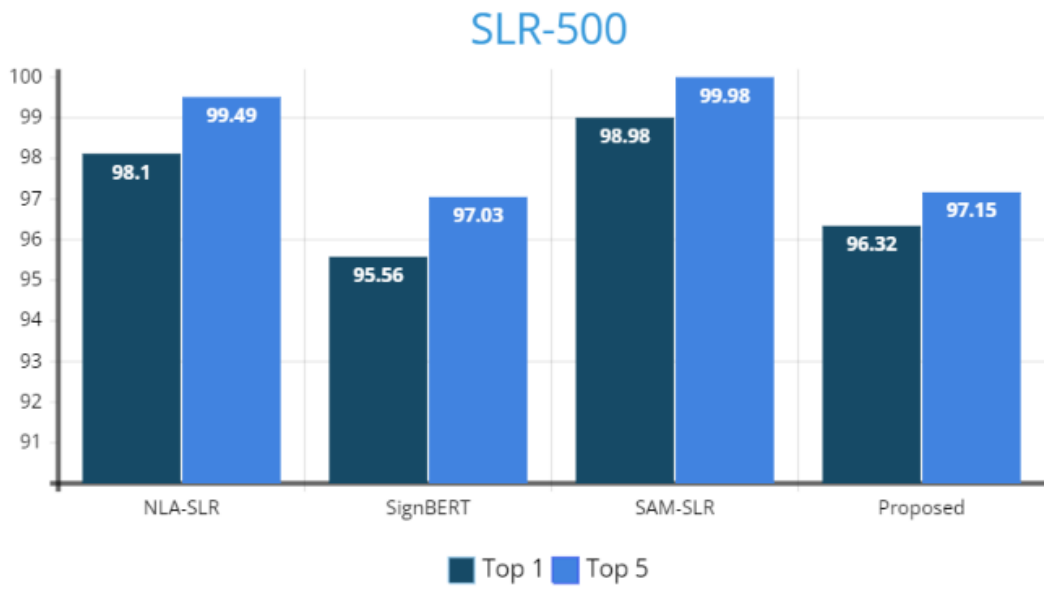


Figure 6.2: Model Results Based on SLR500 Dataset



Figure 6.3: Model Results Based on WLASL100 Dataset





Figure 6.4: Model Results Based on WLASL2000 Dataset

among classes. However there are still lighting issues, jitters, angle and background separation issues present in the model. The overall top-1 score increases a bit but it is not marginal. The proposed architecture accomplishes 63.42% accuracy while NLA-SLR and SAM-SLR produces 67.81% and 65.30% respectively. The primary causes are an inability to recognize landmarks in uneven lighting and the RGB flow feature extraction stops working as intended due to the sudden changes in the spatial dimension. Which is left for future studies to figure out.

## 6.2 Checking for Skin-tone Biases in the Models:

In order to evaluate the presence of bias in the models arising from variations in skintone, a specific investigation was conducted by narrowing down the datasets to videos exclusively featuring individuals with darker skin. However, it became apparent that a significant limitation was encountered due to the scarcity of available data in this category. Notably, the SLR-500 and AUTSL-226 datasets proved unsuitable for this analysis since they lacked any videos showcasing individuals with darker skintones.

Consequently, the focus shifted to the WLASL-100 and WLASL-2000 datasets, where the impact of skintone variance on model accuracy was examined. Strikingly, the results unveiled a noteworthy decrease in accuracy across all comparable models. This suggests that the models may indeed be influenced by bias stemming from skintone variations, emphasizing the need for further investigation and mitigation strategies to ensure fair and unbiased performance in sign language recognition systems. One of the few ways of mitigating such issues is with histogram matching to bring out the proper pixel values along with contrast stretching and transfer learning.

## 6.3 Improvement:

Histogram matching is a process used to adjust the color distribution of an image or video to match a specified reference histogram. A lighter toned reference is used to calculate its histogram to match with the data consisting of darker skinned people. The histogram is then used to capture the frequency distribution of pixel values in each color channel. It is then normalized to ensure it represents probability distributions by dividing the bin values by the total number of pixels in the image or video frame. After normalization, the cumulative distribution functions (CDFs) are calculated for both the reference and target histograms of the data. The CDF represents the cumulative probability of pixel intensity values. The histogram matching algorithm then adjusts the pixel values in the target video frames to align their CDFs with those of the reference. This is achieved by mapping pixel values from the target CDF to the reference CDF. By applying this process, the color distribution of the target image or video frames is transformed to closely resemble that of the reference, leading to improved color consistency and alignment for better feature extraction. On top of that contrast stretching is applied on the data along with color balancing to ensure the most optimal points of interest are being focused.

The pre-trained ResNet-50 model is imported to help with general multipurpose object detections. Weights are updated in the last layer to better fit the feature extraction of dark skinned subjects with an output layer. The training time weights of ResNet-50 are frozen to keep the pre-trained learned features intact and weights of the new output layer are updated using Adam optimizer. The ResNet-50 weights are finetuned with 0.001 learning rate after unfreezing the model. The outputs are then normalized to feed back into the ensemble method multi modality for improved detections.



Figure 6.5: Model Results Based on WLASL100 Dataset Without Bias



Figure 6.6: Model Results Based on WLASL Dataset Without Bias

# Chapter 7

## Epilogue

### 7.1 Insufficient Resources:

Despite the remarkable precision and robustness exhibited by state-of-the-art hand gesture recognition models, they encounter significant challenges when applied to larger datasets. These models often struggle to achieve high accuracy and performance on such datasets due to inherent limitations. **The Limitations are caused by:**

- One major issue lies in the lack of balance within the available datasets. The distribution of hand gesture samples across different classes or categories is often highly skewed, with certain gestures being overrepresented while others are underrepresented. This data imbalance introduces bias into the training process, causing the models to favor the majority gestures and perform inadequately on minority ones. Consequently, the overall accuracy and generalizability of the models are compromised.
- It often needs to process and store large amounts of data, including input images or video frames, intermediate feature representations, and model parameters. Insufficient memory can limit the size of the input data that can be processed or the complexity of the model that can be employed. It may require compromises in terms of data resolution, model architecture, or batch sizes, which can impact the model's accuracy and overall performance.
- The larger datasets used for hand gesture recognition can suffer from incorrect or erroneous labeling. The process of annotating hand gestures is complex and prone to human error, especially when dealing with a large volume of data. Incorrect labels in the dataset can mislead the model during training, negatively impacting its ability to accurately recognize and classify gestures.

### 7.2 Conclusion:

This paper presents an overview of the methods currently used to recognize sign language. This work significantly advances the field of hand gesture to real-time character interpretation systems using a computer vision-based methodology for sign language recognition. A recognition system can easily process and understand

the majority of the signs of other sign languages that have been studied to date because they are based on hand gestures. But if you use a video as your input and then display all of your sign gestures for a recognition system, it becomes very challenging to identify each sign with accuracy. The experimental findings suggest that rather than the work's complexity, the system's recognition performance is adequate. This paper's ultimate goal is to enhance the proposed sign language recognition system. Rather than the difficulty of the task, it can be inferred from the experimental results that the system's recognition performance is satisfactory. Here in this paper, we have created a model which is a combination of models that run simultaneously on the input data while we use the basis of ensemble technique to choose the final result from all the produced outputs. Moreover, our system also deals with the existing skin tone bias in the models by using histogram equalization, data regularization and contrast stretching on the input images and by using transfer learning. The models could be further improved with better resources in terms having adequate samples per class in different lighting conditions and having a variable amount of diversity within the training subjects.

# Bibliography

- [1] S. Begum and M. Hasanuzzaman, “Computer vision-based bangladeshi sign language recognition system,” *ICCIT*, pp. 414–419, 2009. DOI: 10.1109/ICCIT.2009.5407274. eprint: <https://doi.org/10.1109/ICCIT.2009.5407274>. [Online]. Available: <https://doi.org/10.1109/ICCIT.2009.5407274>.
- [2] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, and P. Presti, “American sign language recognition with the kinect,” pp. 279–286, 2011. DOI: 10.1145/2070481.2070532. eprint: <https://doi.org/10.1145/2070481.2070532>. [Online]. Available: <https://doi.org/10.1145/2070481.2070532>.
- [3] B. C. Karmokar, K. M. R. Alam, and M. K. Siddiquee, “Computer vision-based bangladeshi sign language recognition system,” p. 43, 2012. eprint: <https://www.academia.edu/17972640>. [Online]. Available: <https://www.academia.edu/17972640>.
- [4] X. Chai, G. Li, Y. Lin, *et al.*, “Sign language recognition and translation with kinect,” *AFGR*, vol. 655, p. 4, 2013.
- [5] V. Bhome, R. Sreemathy, and H. Dhumal, “Vision based hand gesture recognition using eccentric approach for human computer interaction,” *ICACCI*, pp. 949–953, 2014. DOI: 10.1109/ICACCI.2014.6968545. eprint: <https://doi.org/10.1109/ICACCI.2014.6968545>. [Online]. Available: <https://doi.org/10.1109/ICACCI.2014.6968545>.
- [6] M. Jasim and M. Hasanuzzaman, “Sign language interpretation using linear discriminant analysis and local binary patterns,” *ICIEV*, pp. 1–5, 2014. DOI: 10.1109/ICIEV.2014.7136001. eprint: <https://doi.org/10.1109/ICIEV.2014.7136001>. [Online]. Available: <https://doi.org/10.1109/ICIEV.2014.7136001>.
- [7] P. Premaratne, “Historical development of hand gesture recognition,” in *Human Computer Interaction Using Hand Gestures*. 2014, pp. 5–29. DOI: 10.1007/978-981-4585-69-9\_2. eprint: [https://doi.org/10.1007/978-981-4585-69-9\\_2](https://doi.org/10.1007/978-981-4585-69-9_2). [Online]. Available: [https://doi.org/10.1007/978-981-4585-69-9\\_2](https://doi.org/10.1007/978-981-4585-69-9_2).
- [8] H. Jie, Z. Wengang, L. H, and L. W, “Sign language recognition using 3d convolutional neural networks,” *ICME*, pp. 1–6, 2015. DOI: 10.1109/ICME.2015.7177428. eprint: <https://doi.org/10.1109/icme.2015.7177428>. [Online]. Available: <https://doi.org/10.1109/icme.2015.7177428>.
- [9] L. Pigou, S. Dieleman, P. J. Kindermans, and B. Schrauwen, “Sign language recognition using convolutional neural networks,” in *Lecture Notes in Computer Science*. 2015, pp. 572–578. DOI: 10.1007/978-3-319-16178-5\_40. eprint: [http://dx.doi.org/10.1007/978-3-319-16178-5\\_40](http://dx.doi.org/10.1007/978-3-319-16178-5_40). [Online]. Available: [http://dx.doi.org/10.1007/978-3-319-16178-5\\_40](http://dx.doi.org/10.1007/978-3-319-16178-5_40).

- [10] T. Bohra, S. Sompura, K. Parekh, and P. Raut, “Real-time two way communication system for speech and hearing impaired using computer vision and deep learning,” *ICSSIT*, pp. 734–739, 2019. DOI: 10.1109/ICSSIT46314.2019.8987908. eprint: <https://doi.org/10.1109/ICSSIT46314.2019.8987908>. [Online]. Available: <https://doi.org/10.1109/ICSSIT46314.2019.8987908>.
- [11] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, “Sign language transformers: Joint end-to-end sign language recognition and translation,” 2020. DOI: 10.48550/arXiv.2003.13830. eprint: <https://doi.org/10.48550/arXiv.2003.13830>. [Online]. Available: <https://doi.org/10.48550/arXiv.2003.13830>.
- [12] S. Tornay, M. Razavi, and M. Magimai.-Doss, “Towards multilingual sign language recognition,” *ICASSP*, pp. 6309–6313, 2020. DOI: 10.1109/ICASSP40776.2020.9054631. eprint: <https://doi.org/10.1109/ICASSP40776.2020.9054631>. [Online]. Available: <https://doi.org/10.1109/ICASSP40776.2020.9054631>.
- [13] M. Turk and V. Athitsos, “Gesture recognition,” in *Computer Vision: A Reference Guide*. 2020, pp. 1–6. DOI: 10.1007/978-3-030-03243-2\_376-1. eprint: [https://doi.org/10.1007/978-3-030-03243-2\\_376-1](https://doi.org/10.1007/978-3-030-03243-2_376-1). [Online]. Available: [https://doi.org/10.1007/978-3-030-03243-2\\_376-1](https://doi.org/10.1007/978-3-030-03243-2_376-1).
- [14] S. Jiang, B. Sun, L. Wang, Y. Bai, K. Li, and Y. Fu, “Skeleton aware multi-modal sign language recognition,” *CVPRW*, pp. 3408–3418, 2021. DOI: 10.1109/CVPRW53098.2021.00380. eprint: <https://doi.org/10.1109/cvprw53098.2021.00380>. [Online]. Available: <https://doi.org/10.1109/cvprw53098.2021.00380>.
- [15] D. Li, C. R. Opazo, X. Yu, and H. Li, “Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison,” *WACV*, pp. 1448–1458, 2021. DOI: 10.1109/WACV45572.2020.9093512. eprint: <https://doi.org/10.1109/wacv45572.2020.9093512>. [Online]. Available: <https://doi.org/10.1109/wacv45572.2020.9093512>.
- [16] Y. Chen, F. Wei, X. Sun, Z. Wu, and S. Lin, “A simple multi-modality transfer learning baseline for sign language translation,” *CVPR*, pp. 5110–5120, 2022. DOI: 10.1109/CVPR52688.2022.00506. eprint: <https://doi.org/10.1109/cvpr52688.2022.00506>. [Online]. Available: <https://doi.org/10.1109/cvpr52688.2022.00506>.
- [17] Z. Wang, T. Zhao, J. Ma, *et al.*, “Hear sign language: A real-time end-to-end sign language recognition system,” *TMC*, vol. 21, no. 7, pp. 2398–2410, 2022. DOI: 10.1109/TMC.2020.3038303. eprint: <https://doi.org/10.1109/TMC.2020.3038303>. [Online]. Available: <https://doi.org/10.1109/TMC.2020.3038303>.