National College of Ireland

Deep Learning Approaches to Real-Time Sign Language Recognition and Multilingual Translation

MSc Research Project
Data Analytics

Harika Guttula
Student ID: x23237431

School of Computing
National College of Ireland

Supervisor: Dr David Hamill

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Harika Guttula		
Student ID:	X23237431		
Programme:	Data Analytics	**Year:**	2024
Module:	MSc Research Project		
Supervisor:	Dr David Hamill		
Submission Due Date:	12/12/2024		
Project Title:	Deep Learning Approaches to Real-Time Sign Language Recognition and Multilingual Translation		
Page Count:	23		
Word Count:	8545		

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Harika

Date: 12-12-24

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Deep Learning Approaches to Real-Time Sign Language Recognition and Multilingual Translation

Harika Guttula
x23237431

Abstract

The following study proposes the Convolutional Neural Networks (CNNs)-Recurrent Neural Networks (RNNs) hybrid deep learning model for the recognition of American Sign Language (ASL) gesture and real-time multilingual speech translation. The CNN part is used to extract spatial features from the ASL gesture images and the RNN part is used to capture temporal features using Long Short-Term Memory (LSTM) networks. The model is built to recognize 36 classes of ASL gestures including digits 0-9 and alphabets A-Z and is combined with multilingual speech output module using Google Text-to-Speech (gTTS) that can translate the recognized signs into spoken words in Spanish, French and Arabic.

The model was trained and tested on dataset of 2,515 images of ASL and the performance of the model was calculated on 10 iterations. Training accuracy increased from 27.10% to 97.66% and validation accuracy achieved 99.01%. The test accuracy was 97.61% proving that this model has a very high generalisation capacity. The performance of the classifiers was as follows: precision of 97%, recall of 98%, and F1-score of 97%. However, the metrics of some classes like ‘o’ and ‘z’ are slightly lower and it is obvious that class imbalance and feature overlap issues are the main causes that need to be improved. The ability to output speech in multiple languages is a major advantage and increases the model’s practical usability across the range of people and situations. This research focuses on the application of the proposed hybrid CNN+RNN model in ASL gesture recognition and its possible use in translating real-time sign language to spoken language and vice versa. Additionally, the findings contribute to the development of assistive technologies, offering a solid foundation for the advancement of ASL recognition systems. Future work will focus on addressing minor performance discrepancies and exploring advanced techniques such as data augmentation and specialised loss functions to further optimise the model.

Keywords: Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), American Sign Language (ASL), LSTM, ASL gesture recognition

1 Introduction

1.1 Background and Context

Communication is one of the basic human needs, but millions of people around the globe have major problems with speech due to hearing impairments. The society also has disabled

persons who are either deaf or have a hearing impairment, and sign language is the most common means of communication among them. It is a visual language that uses hand movements, facial expressions and body language to pass a message. However, there is no common sign language and inadequate knowledge of it among the population resulting in deep communication issues. Such barriers interfere with day-to-day communication, restrict opportunities to obtain necessary supplies and goods and exacerbate the lack of both contacts and services (Manning et al., 2022).

Over the last few years, technology has presented new opportunities to overcome the communication barriers of disabled individuals. There is thus much interest in sign language recognition system where an application can be used to translate sign language into the spoken language in real time. Thus, despite previous attempts, existing systems are still problematic in terms of accuracy and speed and cannot be considered scalable to a real-world application (Koller, 2020). Moreover, most of the existing solutions provided are available in only one language, which makes them less accessible to everyone (Pigou et al., 2015).

This research proposes a real-time sign language recognition system to overcome these constraints and adds multilingual text-to-speech translation for increased applicability. Using CNN and RNN structure, the system will be able to capture spatial features as well as temporal features of sign language gestures to identify them accurately (Huang et al., 2015). Also, the incorporation of Google Text-to-Speech (gTTS) will allow translation of the content into various languages enhancing the system's applicability to a wide clientele. To achieve this, this thesis aims to develop a tool that can be comprehensive and easy to use in facilitating interaction with hearing and speech impaired.

1.2 Motivation

The need for this research arises from the observation of the high levels of communication is impaired by hearing and speech impaired. The inability to communicate effectively with the hearing population causes several social, educational and employment related losses. However, sign language is well-recognized and efficient for deaf people; however, deaf people have some problems, and most people do not know how to use sign language or how to read it (Bragg et al., 2019). Cohen and Muller argued that this lack of trade creates exclusion and isolation of the sign language users.

Technological solutions to sign language recognition at the current are still favorable but not perfect. The current systems also have some challenges such as they are not real-time systems, low recognition accuracy and cannot translate sign language to other spoken languages. These limitations make them less effective to be implemented in a real environment (Camgoz et al., 2020). With the increase of the use of digital platforms and video communication in society, there is an acute need to develop a stable, time-synchronous system that will enable the exchange of information between sign language users and other people. Further, it would be immensely helpful in the development of the multilingual sign language recognition system to foster the improvement of inclusive settings in different spheres of life, such as education, healthcare, customer relations, and administration (Tang,

G. 2024). Since the identified sign language gestures can be translated into several spoken languages, such a system can significantly enhance the presence and influence of sign language users to convey information to people who have diverse language and cultural backgrounds.

1.3 Research Question

The central research question guiding this study is as follows:

How can deep learning techniques be utilized to develop a sign language recognition system capable of accurately translating gestures into multiple spoken languages?

This question focuses on the dual challenges of developing a system that can recognize sign language gestures with high accuracy and translating those gestures into spoken languages using a multilingual text-to-speech system.

1.4 Research Objectives

This study aims to design a highly accurate, real-time sign language recognition system together with multilingual text-to-speech translation. To achieve this, the research is guided by the following specific objectives:

- To **develop a real-time sign language recognition system**: This includes applying CNNs for capturing the spatial features of sign language gestural movements and RNNs for the temporal features, all with improved recognition rate.
- To **integrate multilingual text-to-speech (gTTS)**: This feature will enable the system to drive speech where the sign language interpreter identified the sign language gesture and translated it into various languages if needed.
- To **evaluate the system's performance**: The system will be evaluated in terms of its performance, response time and applicability in a variety of realistic settings in order to verify that it is feasible for use in real life situations.
- To **assess the social impact**: The study will also look into how the system can enhance the accessibility of hearing and speech impaired in the different aspects of life to minimize communication hitches hence enhancing their integration into society.

1.5 Significance of the Study

This research has the potential to benefit the field of assistive technology and accessible communication in a major way. If the real-time sign language recognition system is integrated with speech to multilingual speech translation, a lot of communication barrier between the hearing impaired and the others would be broken. Such a system would not only support day-to-day interpersonal communication but would also help to achieve better integration into education, health care, public services and working life (Manning et al., 2022). This research seeks to address the gaps in existing systems by implementing deep learning methods that are more accurate, scalable and generic. Moreover, the use of multiple languages in the system guarantees that this can be implemented in different linguistic and

cultural environments (Tang, G. 2024). In Summary, the research intends to make a significant contribution to the enhancement of communication for the disabled persons with hearing and speech impairment to increase their interaction with the hearing community.

1.6 Structure of the Thesis

The remainder of this thesis is organized as follows:

- **Chapter 2: Literature Review** – This chapter gives a brief overview of the current work and developments made in the field of sign language recognition with the help of deep learning approaches, real-time systems, multi-lingual translation systems.
- **Chapter 3: Methodology** – This chapter outlines the deep learning models, datasets, system architecture, and tools used to build the sign language recognition system, together with the approach for the integration of multilingual text-to-speech.
- **Chapter 4: Implementation and Results** – This chapter focuses on highlighting the proposed system, the actual training, testing and performance of the model. The impact of the system in the recognition accuracy, response time, and the performance of the multilingual translation is also evaluated.
- **Chapter 5: Discussion** – This chapter compares the findings of this study to the research objectives and present an assessment of the likely challenges, limitations and usefulness of the system.
- **Chapter 6: Conclusion and Future Work** – This chapter discusses the conclusion of the research, the contributions made in enhancing the performance of the system and the possible future improvements in improving the system's functionality and scalability.

2 Related Work

Sign language recognition has seen enormous research developments in the last few decades, especially with the development of deep learning frameworks. This chapter gives a review of the current literature regarding sign language recognition, real-time processing, multilingual translation, and deep learning techniques such as CNNs and RNNs. The chapter is organized into four key sections: an introduction to sign language recognition system, the importance of deep learning in sign language recognition, integration of multilingual translation and the problems associated with current sign language recognition systems.

2.1 Sign Language Recognition Systems: A Historical Perspective

The recognition of sign language has been an active area of research for a long time as the main goal is to improve the interaction between the hearing impaired and the rest of society. The first systems used sensor-based approach to capture the hand movements and gestures; the gloves used had sensors. For instance, the "Power Glove" designed by (Zimmerman et al. 1986) was one of the very first efforts at identifying hand movements for human-computer interface. However, sensor-based systems were costly, bulky, and only able to capture the gross motor movements of sign language gestures, facial expressions, and body posture (Starner et al., 1995). As computer vision progressed, video-based methods for sign language recognition started to come to the foreground. Unlike earlier systems, it employed cameras

for monitoring hand and body movements to perform the tasks. (Starner et al., 1995) put an initial real-time ASL recognition system using Hidden Markov Models which was followed by many other researchers. However, HMMs were not efficient for addressing the spatial-temporal characteristic found in sign language. This led to further investigation of higher-level machine learning approaches like the Support Vector Machines (SVMs) and decision trees (Zhang et al., 2005) nevertheless these first-generation systems lacked accuracy and scalabilities

2.2 Deep Learning in Sign Language Recognition

The application of deep learning began to change the development of sign language recognition. They have been referred to as the state of the art for image-based tasks which make CNNs suitable for the spatial analysis of sign language gestures. This is because CNNs are very good at capturing shapes, like in the hands, faces or bodies which are essential in sign language (Koller et al., 2020). For example, Huang et al. 2015) put forward a 3D CNN-based system, which integrated the spatial feature of sign language gestures and achieved better performance in terms of accuracy. However, sign language is both spatial and temporal because the meaning of a sign depends on the temporal order of movements. This limitation of CNNs led to the use of Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) variants. These networks are good at representing temporal dependencies which make them useful in capturing temporal dependencies of sign language gestures (Gomathi V. 2021). Due to the hybrid use of CNNs to extract spatial features and RNNs to model temporal features, the precision and generalization of sign languages' recognition have shown a massive upgrade in this area of study (Camgoz et al., 2020). Besides CNN-RNN architectures, transformer models that have been significantly effective in natural language processing tasks are only now being considered for sign language recognition. The transformers have the capability to learn long distance contexts in sign language sequences which makes it a good area for future work (Vaswani et al., 2017).

2.3 Multilingual Translation in Sign Language Recognition

Thus, whereas most of the early works in sign language recognition aimed at translating signs into a single spoken language, the globalization of today's society requires multilingual solutions. Current systems, for instance those presented in (Bragg et al, 2019), are generally designed to work with outputs in English only which are off-putting to non-English speaking persons. The use of the multilingual text to speech translation system like gTTS has the potential of extending the application of sign language recognition technology to many languages.

The use of gTTS in sign language recognition has not been widely implemented and its inclusion could be greatly useful in situations where sign language users are expected to interact with people from different linguistic abilities. For instance, real-time sign language recognition system with multiple languages translation ability can be used for communication in international airports, health care centers and schools etc (Pigou et al., 2015). The purpose of this study is to investigate the use of gTTS for translating the identified sign language gestures into multiple spoken languages to increase the flexibility of the system.

2.4 Literature Review Comparison

This section offers a comparative survey of different research papers on sign language recognition (SLR) systems. The comparison is provided below Table 1, emphasizing method, models, performance measures, constraints, and potential future research. It is common in the information comparing and structuring to define general trends and reveal strengths and weaknesses of the existing approaches, as well as to distinguish potential improvements in the workflow of future research.

Authors	Methodology	Model Used	Limitations	Future Work
Starner & Pentland (1995)	Real-time recognition using video-based data and Hidden Markov Models (HMMs)	HMM	Limited scalability for large vocabularies; focused on American Sign Language (ASL) only	Develop models for larger vocabulary and other sign languages
Pigou et al. (2015)	Used CNN for recognizing continuous gestures in a video sequence	CNN	Could not capture temporal dependencies in gestures	Integrate RNNs or LSTMs to capture temporal features
Huang et al. (2015)	Employed 3D CNN to capture spatial features of sign language gestures	3D CNN	Limited to static hand gestures; no integration of facial expressions or body posture	Extend the model to capture dynamic gestures and non-manual signals (facial expressions, body movements)
Camgoz et al. (2020)	Combined CNN for spatial feature extraction with RNN for temporal modeling	CNN + RNN (LSTM)	Lacked multilingual capabilities; trained only on single sign language	Incorporate multilingual sign language translation and improve generalization to multiple sign languages
Koller et al. (2020)	Quantitative survey using CNN for sign language recognition	CNN	Did not address real-time processing; limited to controlled environments	Focus on real-time systems and expand dataset for in-the-wild scenarios

Table 1: Comparison of Existing Research

2.5 Challenges and Limitations of Existing SLR Systems

While there has been much progress into sign language recognition some hurdles persist. The first problem is the fact that sign language is different in different countries and even in different cultures. For instance, American Sign Language (ASL) and British Sign Language (BSL), are two different sign languages, and within each of these there are also variations due to regional difference (Tang G. 2024). It is still a major issue to design a system that is capable of recognising and interpreting various sign languages with high levels of accuracy.

Another problem is the incorporation of non-manual signs that feature facial expressions and posture that are other crucial parts of sign language (Koller, 2020). Although CNNs are proficient in detecting hand movements, detail work such as the slight difference in position is not deciphered clearly often resulting in misunderstandings or omissions of some signs in sign language.

Real time data processing is another major challenge. While systems such as that proposed in (Gomathi V. 2021), have exhibited good results, computational complexity inherent to deep learning models presents the issue of latency, which is a constraint in real-time applications. The best efforts to minimize the latency while not compromising on the accuracy continue to be a difficult task in the creation of SLR systems. Thirdly, the datasets used for training SLR systems are generally small and the systems may not be generalized. Many sign languages recognition systems are based on a particular set of data which do not always encompass the full variety of sign languages. For instance, the major datasets like RWTH-PHOENIX-Weather 2014 (Forster et al., 2014), are oriented on specific domains and languages that makes the system unsuitable for other scenarios. To that end, the creation of larger, more diverse datasets is necessary to build more stable, scalable solutions.

2.6 Summary

Exploration of sign language recognition, therefore, shows that there has been steady development from the initial mechanical sensor-based systems to the current deep learning systems. The combination of CNNs for spatial feature extraction and RNNs for temporal modeling has produced many enhancements to the recognition accuracy. Nevertheless, current systems continue to struggle with such issues as multilingual translation, handling of non-manual signals, real-time translation, and the application of the models across different sign languages.

This research seeks to fill these gaps by proposing a sign language recognition system with multilingual text to speech using gTTS. CNNs and RNNs will be integrated together so as to understand both the spatial and temporal characteristics of sign language gestures and gTTS for transcribing into multiple spoken languages. By filling the present gaps in accuracy, scalability and multilingual support, this research aims to contribute to the field of assistive technology as well as enhance the quality of life for people with hearing and speech disabilities. The next section will describe the method used in this research.

3 Research Methodology

This chapter gives specific information for designing deep learning models for the purpose of ASL gesture recognition and translating them into multiple languages. The data used in this work are images of the signs of ASL which correspond to 36 classes (digits and letters). The method includes pre-processing these images, training CNN-RNN model for gesture classification and using multiple language TTS for translating the identified signs into audio.

3.1 Research Design

The research adopts an experimental methodology that involves image processing, deep learning, and translation towards the development of ASL recognition system. The stages are as follows:

- **Dataset Handling and Preprocessing:** Organizing the images into a format as well as transforming the images to increase variability and scaling them for model training.
- **Model Development**: Using CNN to model spatial image attributes and then using RNN to identify gestures in real-time sequences.
- **Multilingual Speech Translation**: Using Google Text-to-Speech (gTTS) for spoken language translation in multiple languages depending on the client's need to make it easier.

- **Evaluation and Validation**: Outcomes in terms of accuracy, F1-score, precision, and recall values, and system latency as key goals for accurate and fast gesture recognition and translation.

3.2 Dataset Collection and Preprocessing

Dataset Source:

The dataset employed in this research is derived from Kaggle ASL dataset (American Sign Language)[1]. The database is composed of 2,515 images with corresponding labels that represent any ASL character or digit separated into subfolders according to categories are for each class in Figure 1. This structure helps in easy handling of files and loading the same as it is represented in Figure 2.

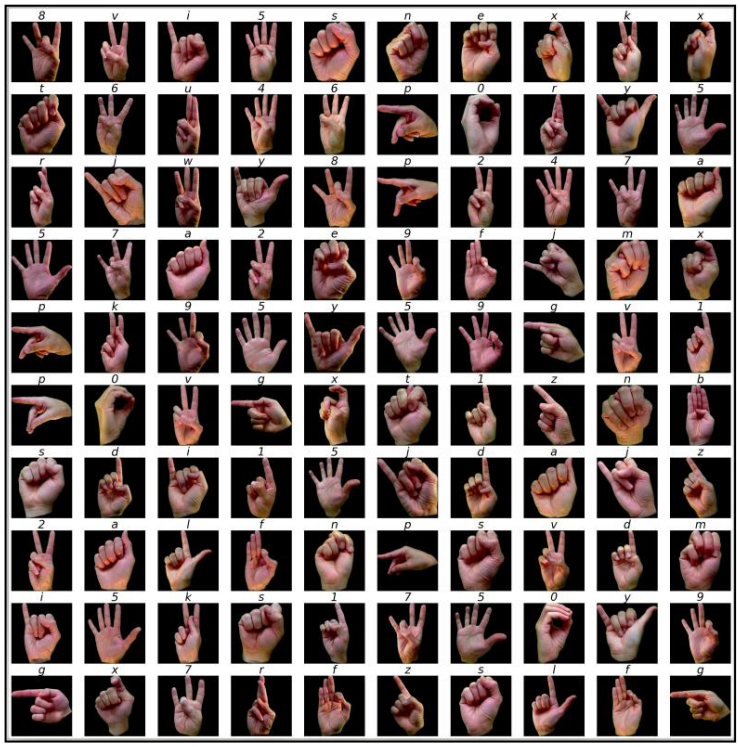

Figure 1: Sign Language Images with Labels

Dataset Structure:

The dataset structure within the main directory is as follows situated in Figure 2:

```
asl-dataset/asl_dataset/
├── 0/
│   ├── hand1_0_top_seg_1_cropped.jpeg
│   └── ...
├── a/
│   ├── hand2_a_dif_seg_2_cropped.jpeg
│   └── ...
└── z/
    ├── hand1_z_bot_seg_4_cropped.jpeg
    └── ...
```

Figure 2: Dataset Structure in Directory

[1] https://www.kaggle.com/datasets/ayuraj/asl-dataset/data

Every subfolder corresponds to a particular class, and the images within the folder have corresponding labels. A DataFrame is used to store image file paths and labels to facilitate data loading in batches during model training.

Preprocessing Steps:

1. **Image Resizing:** Images are resized with the dimensions of 64×64 pixels to enhance training efficiency due to equivalently less input size but containing all significant gestures.

2. **Normalization:** Each pixel value is scaled to the range of 0 to 1. These normalizations minimize variations and speed up the process of convergence of the model.

3. **Data Augmentation:**
   a) **Objective:** To mimic variations that exist in ASL gestures, image augmentation approaches are employed to generate variations in images.
   b) **Techniques Used:** Random rotations, scaling, brightness and very small horizontal or vertical shifts simulate changes in the hand orientation, distance to the camera, and lighting. This makes dataset stronger and its performance under different conditions to be good for the model.

4. **Label Encoding:** The gesture category in each video is associated with a unique integer identifier to easily convert the categorical labels into numerical values for the classification task. These encoded labels are easy to process by the model during training, thanks to the encoding function.

5. **Data Splitting:** The data is divided into training (80%), validation (20%) and Separate test data (10%). The training data is employed for model learning, validation data for model regularization for preventing an overfitting, and test data for assessing final performance of the model.

3.3 Model Architecture

The choice of CNN-RNN for ASL recognition has been made since it was found suitable in previous work and is capable of handling spatial and temporal information. As observed in (Pigou et al. 2015) and (Gomathi V. 2021), even though CNNs are very good at learning spatial features – hand shapes, orientations, and positions – they are completely blind to temporal dependencies, which are inherent to most gestures. It is for this reason that incorporating RNNs particularly LSTMs to model temporal dependencies of gestures has been very successful in ASL recognition. Other past works have also indicated that the combined CNN and RNN model have better performance than individual models in identifying the intricacies of ASL gestures. Though transformer models have been discussed for sequence modelling, they are not very efficient for real-time ASL recognition because of larger dataset requirements and need for more computational resources. Besides, the proposed combination of CNN and RNN is harmonious, giving CNN a choice of functioning on a frame-by-frame basis and giving LSTM the chance to comprehend the sequential nature of sign language. Based on experiences with real-life systems, the CNN-RNN architecture is still proven to be effective and efficient for this task.

The ASL recognition system described in the paper uses a combined CNN-RNN training, where each training is designed to handle spatial and temporal data respectively.

Convolutional Neural Network (CNN) (Chauhan et al., 2018):

- **Purpose:** CNN layers detect features at the per-frame level from each of the images, including the shape and orientation of the hands as well as the position of the fingers, something that is very useful when determining between the different gestures in ASL.
- **Architecture:** The CNN component includes a series of convolution layers for feature extraction, the pooling layer for dimensionality reduction and computational optimization. Batch normalization is used to make the learning process stable, and dropout to reduce the overfitting of a model.
- **Output:** They are followed by the CNN layers and each image is represented by a feature vector. This vector encodes and retains spatial information and is conveyed to the RNN for sequential analysis.

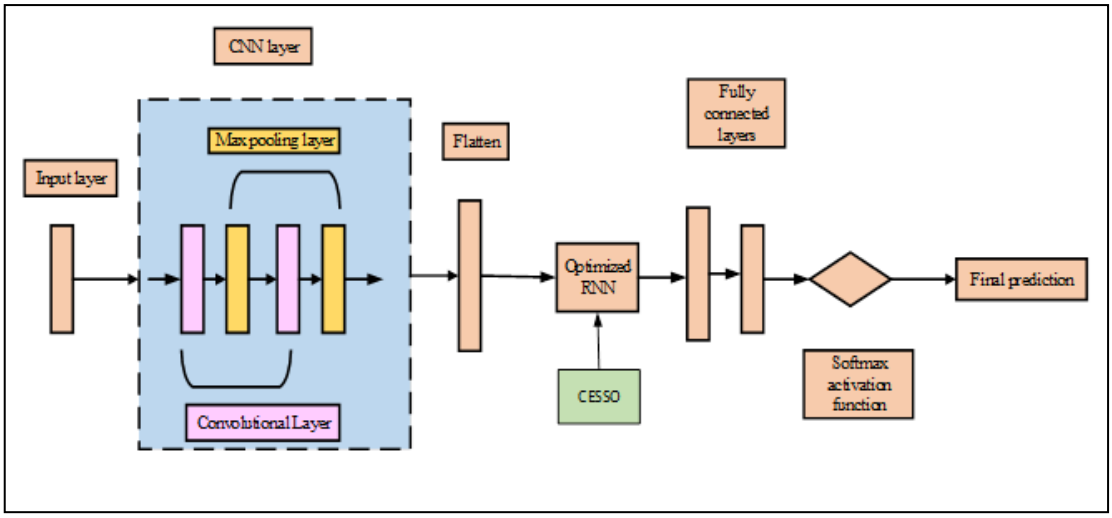

Figure 3: Architecture of the Hybrid (CNN-RNN) Model (Sunganthi et al., 2020)

Recurrent Neural Network (RNN) (Sherstinsky, A 2020):

- **Purpose:** To provide real-time ASL recognition and especially for words or phrases which are made up of multiple gestures, the RNN learns temporal patterns from the sequence of gestures.
- **Architecture:** LSTM layers are used as they have the capability to learn long dependency. The feature vectors of CNN are then passed sequentially into the LSTM which provides the prediction of the gesture frame.
- **Classification Layer:** The output of the LSTM is used in a dense layer which uses softmax activation function to classify the gesture into one of the 36 categories (0-9, A-Z).

This makes it easy for the hybrid model to understand complex ASL gestures effectively, using CNN for spatial identification, and LSTM for temporal analysis.

3.4 Multilingual Speech Translation

To further extend the use of the ASL recognition model, the multilingual text-to-speech system is integrated using Google Text-to-Speech (gTTS).

1. **Text-to-Speech Conversion:** Following gesture recognition, the perceived ASL character or phrase is translated to text before going through gTTS where it is spoken, giving audio feedback.
2. **Language Selection:** The gTTS API can handle multiple languages; the user has a chance to choose a preferred language for spoken output. This feature enhances the model's usability and helps in breaking barriers of cross-cultural communication.
3. **Real-time Processing:** The translation functionality is intended to work with a small-time latency to allow for real-time spoken feedback.

By integrating with gTTS, the ASL recognition model is developed to become a diverse interpreter that meets different linguistic demands, benefiting the ASL users worldwide.

3.5 Model Training and Optimization

- **Training Setup:** The model is trained with categorical cross entropy as the loss function and Adam as the optimizer due to its capability in large scale deep learning. The optimizer's learning rate changes as the training progresses makes it easy to converge on the global minimum without getting caught in local minimum. The model is trained on 2012 images (80%) from the total images of 2515 images to make sure that the model can learn good features.

- **Hyperparameter Tuning:** As hyperparameters, the learning rate, the number of CNN and LSTM layers, batch size, and the dropout rate are chosen due to the grid search and cross-validation. This process also improves the performance, prevents the overfitting of the model and guarantees the best output of the model.

- **Epoch and Batch Size Selection:** The model is trained for the initial epochs, regularity, and early stop if the model validation shows that the model is not improving anymore. Batch size is selected to optimize the computational process and avoid the problem with memory usage during training. The final batch size is chosen to optimize computational speed and memory constraints to allow for successful model training while still using the information from the 2012 training images effectively.

- **Regularization Techniques:** To overcome overfitting problem dropout layers are included between CNN and LSTM layer, to improve the training and generalization ability of network batch normalization is included. These techniques are very useful whenever we are dealing with a diverse data set such as the ASL recognition data set, so that the model can perform well on unseen data.

- **Evaluation on Validation Set:** The results are tested on the 503 validation images (20%) after training of every epoch using the accuracy and F1-score indicators. Validation serves to optimize model parameters and check its ability to generalize before running the final examination of the model utilizing the separate test set. 251 images (10%) are kept aside from the testing set to provide an impartial assessment of the model's practical usability.

3.6 Evaluation Metrics

The model's performance is evaluated based on multiple metrics (Dalianis et al., 2018):

- **Accuracy:** Calculates the total accuracy of the forecast made in all the gestures.

- **Precision, Recall, and F1-Score:** Assess the performance of the model in its recognition of ASL gestures whilst avoiding misidentifying wrong gestures as well as failing to identify correct ones.
- **Confusion Matrix: E**nables the determination of performance within specific classes and displays the number of correct and incorrect predictions for each ASL category.

3.7 Summary

This chapter provides the complete procedure to train an ASL recognition system and translating it into multiple languages. By using CNN and RNN, together with Google Text-to-Speech, real-time recognition and multilingual accessibility for different populations is improved. This system has the possibility of serving the Deaf community as an easily usable tool, with further releases focusing on increasing the number of supported languages and improving the precision of the models through the incorporation of deep learning. In the next chapter, review the first results of the American Sign Language (ASL) dataset and discuss the distribution of the dataset, mean imbalances, or any other issues that may exist in the dataset with possible solutions.

4 Initial Findings

This work was very useful for the understanding of the structure and distribution of the American Sign Language (ASL) dataset to prime the classification model. The dataset has 36 classes which is 26 letters of the alphabet (a–z) and 10 numeric digits (0–9). A breakdown of the images in all classes showed a total of 2515 images, with each class having 70 samples of images apart from class ‘t’ which has 65 samples.

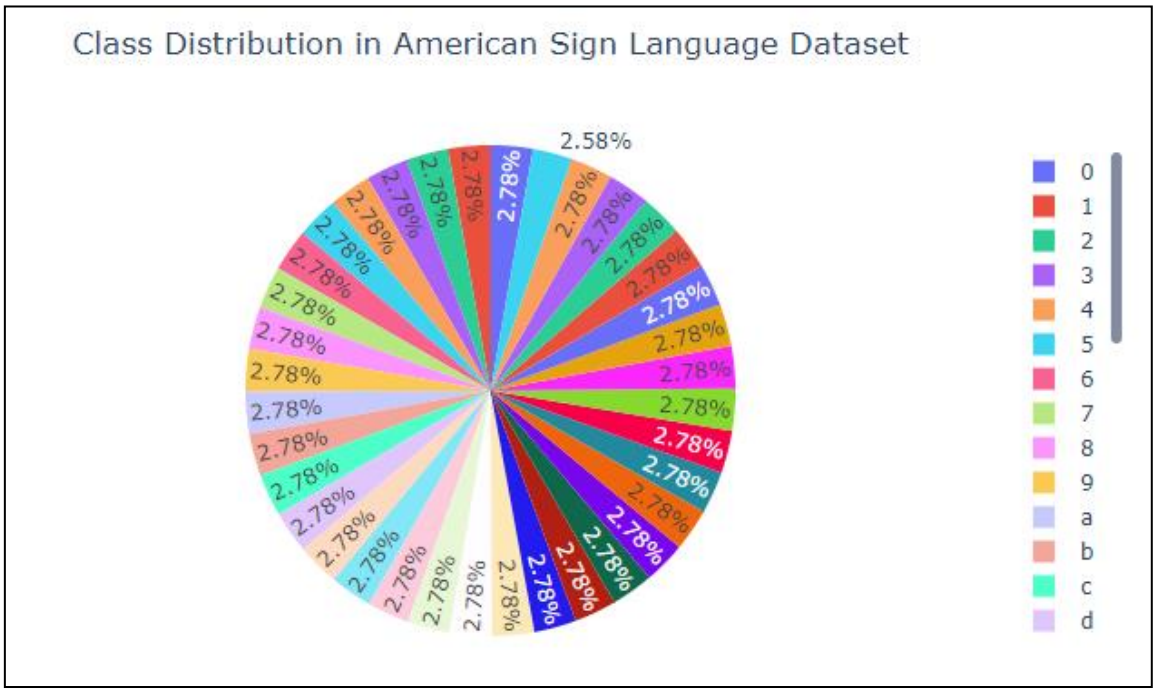

Figure 4: Class Distribution for ASL Dataset

This minor class imbalance made it necessary to look at ways to balance learning across classes during the model development. The fact that most classes are almost equally represented also indicates that the dataset is appropriate for balanced multi-class

classification. Class distribution was also presented using a bar chart to give an overall picture of the dataset. The mean class count was calculated and drawn as a horizontal line so that deviations could be observed. It was found that most of the classes were within a close range around the mean, an indication of a normal distribution data set. The underrepresentation of class was barely noticeable, and it was evident that this category should be closely watched during the training phase to prevent either overfitting or underfitting.

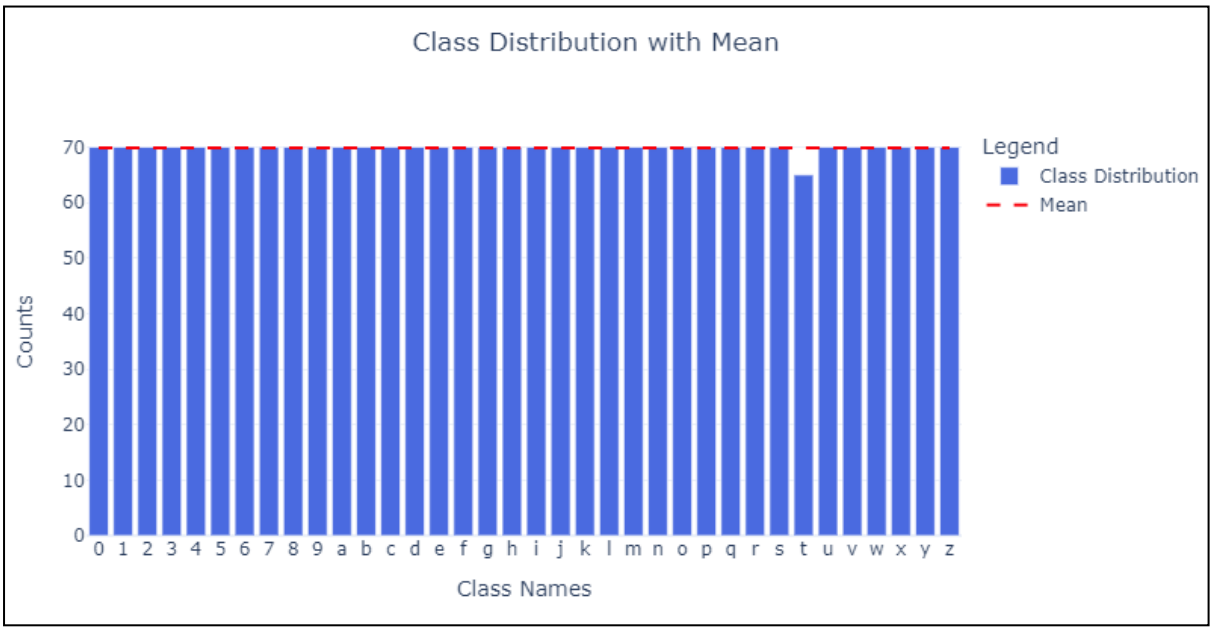

Figure 5: Class Distribution of Classes through Mean

The next step is a division of obtained dataset into the training, validation, and test sets. Out of the total of 2515 images, 2012 images (80%) were used in the training process to allow the model enough data to capture good features and 503 images (20%) were used in the validation process to allow periodic checks on the generalization of the model. Moreover, a distinct set of 251 images (10%) was selected for testing purposes only so as not to interfere with the model's performance and prove the effectiveness of the work done. The preparation of such a diverse dataset was made easier by TensorFlow's image_dataset_from_directory function that creates the dataset and is reproducible through proper seeding.

```
Found 2515 files belonging to 36 classes.
Using 2012 files for training.
```

```
Found 2515 files belonging to 36 classes.
Using 503 files for validation.
```

```
Found 2515 files belonging to 36 classes.
Using 251 files for validation.
```

Figure 6: Splitting of the Dataset (80:20)

The first thing I noticed: the sizes of the resulting subsets were quite like each other, which meant that the model was being trained and validated equally in all classes. The records in each of the subsets were randomized to reduce the chances of bias by providing random and equally distributed data. Another important finding was obtained from the preprocessing pipeline that is used on the dataset. Images were reduced to the target size to meet the input layer of the model while preserving important characteristics. The decision to utilize mini-batches during training (batch size defined programmatically) was reasonable with regard to both, the computation time and model performance. Also, the application of standard

approaches, for instance, the image normalization helped to achieve the unified distribution of pixels and, therefore, increase convergence during learning.

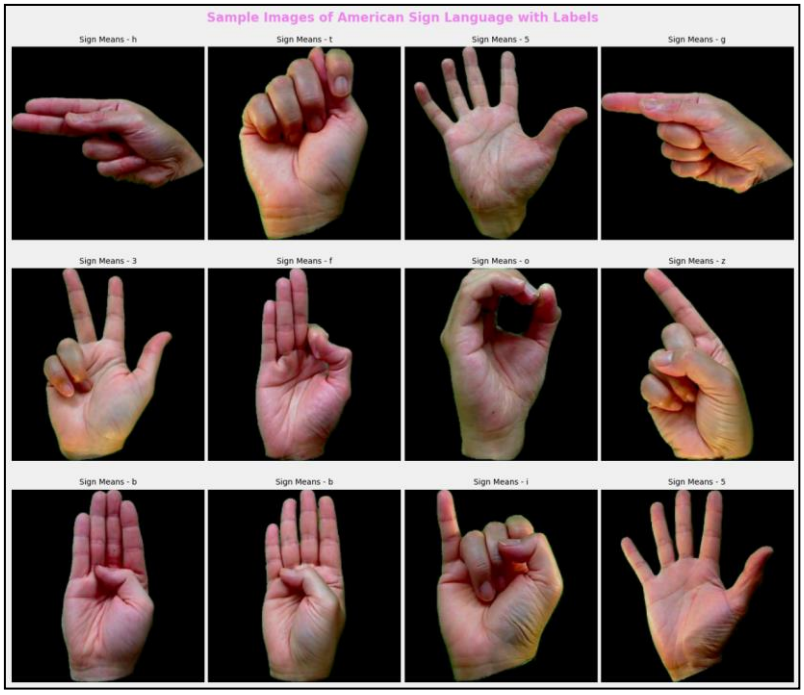

Figure 7: Sample Images of ASL with Labels

As observed on the first data set, it was evident that the dataset was ready for training, though class ‘t’ was slightly imbalanced requiring oversampling or augmentation. So, these strategies could improve the model’s performance for underrepresented classes without deteriorating the underlying training process. The initial findings also emphasized the need to have a proper EDA before developing the models so that any issues that might cause problems are noticed early enough.

```
Epoch 1/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 145s 2s/step - accuracy: 0.2710 - loss: 2.9534 - val_accuracy: 0.8290 - val_loss: 1.3338
Epoch 2/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 123s 2s/step - accuracy: 0.7781 - loss: 1.2561 - val_accuracy: 0.9145 - val_loss: 0.6242
Epoch 3/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 125s 2s/step - accuracy: 0.8765 - loss: 0.6624 - val_accuracy: 0.9682 - val_loss: 0.3332
Epoch 4/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 129s 2s/step - accuracy: 0.9317 - loss: 0.4138 - val_accuracy: 0.9602 - val_loss: 0.2406
Epoch 5/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 126s 2s/step - accuracy: 0.9349 - loss: 0.3141 - val_accuracy: 0.9781 - val_loss: 0.1537
Epoch 6/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 127s 2s/step - accuracy: 0.9638 - loss: 0.2246 - val_accuracy: 0.9861 - val_loss: 0.1256
Epoch 7/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 115s 2s/step - accuracy: 0.9703 - loss: 0.1973 - val_accuracy: 0.9801 - val_loss: 0.1012
Epoch 8/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 118s 2s/step - accuracy: 0.9577 - loss: 0.1816 - val_accuracy: 0.9901 - val_loss: 0.0804
Epoch 9/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 123s 2s/step - accuracy: 0.9777 - loss: 0.1402 - val_accuracy: 0.9742 - val_loss: 0.0896
Epoch 10/10
63/63 ━━━━━━━━━━━━━━━━━━━━ 126s 2s/step - accuracy: 0.9766 - loss: 0.1289 - val_accuracy: 0.9901 - val_loss: 0.0597
```

Figure 8: Model Training

This step showed that before developing the models it requires a good understanding of the provided dataset. The integrity of the dataset was assured, class distribution was analyzed, and a sound approach for data division was provided, which formed a solid foundation for an efficient and scalable classification solution. These basic actions gave the confidence needed to continue to the model training phase with the knowledge of the strong and weak points of

the data set. The following chapter contains the assessment outcomes and conclusions on the model with the emphasis on the training, validation and testing procedures.

5 Model Evaluation Results and Findings

This section presents the analysis of the performance of the proposed model during the training, validation, and the testing phases in detail, success and challenges. The findings are derived from the evaluation of such factors as accuracy, loss, precision, recall, and F1 score, as well as the model's capacity to classify the 36 different classes successfully.

Table 2: Model Performance for 10 Epoch (Training & Validation)

Epoch	Accuracy	Loss	Validation Accuracy	Validation Loss
0	0.2710	2.9534	0.8290	1.3338
1	0.7781	1.2561	0.9145	0.6242
2	0.8765	0.6624	0.9682	0.3332
3	0.9317	0.4138	0.9602	0.2406
4	0.9349	0.3141	0.9781	0.1537
5	0.9638	0.2246	0.9861	0.1256
6	0.9703	0.1973	0.9801	0.1012
7	0.9577	0.1816	0.9901	0.0804
8	0.9777	0.1402	0.9742	0.0896
9	0.9766	0.1289	0.9901	0.0597

5.1 Training and Validation Performance

The entire training process was conducted in ten epoch, and during that, the model was boosted steadily on the given task. The accuracy of training in relation to the epoch was high, the initial accuracy of 27, 10% in the first epoch rising to 97, 66% in the last epoch as shown in table 1 below. This upward trend clearly shows that the model can learn complex patterns from the data set. The following training loss also reflected the same benefit, with the initial high MAE value of 2.9534 dropping to 0.1289 as the model achieved minimal prediction errors during training as these values are demonstrated in Table 2 and visually represented in Figure 4.

Like validation metrics, test metrics also exhibited good results suggesting that the model was highly powerful in generalization over unseen data. From an initially set epoch, the validation's accuracy increased from 82.90 % to a remarkable 99.01%. Also, the validation loss was decreased from 1.3338 to 0.0597 as is demonstrated in Figure 4. This equality of the training and validation metrics is an essential sign of their quality since it does not overfit and has a good, validatable generalization. The gradual shift and a smooth curve of both accuracy and loss over epochs also proves the efficiency of the proposed architecture and the training methodology to learn features from the data.

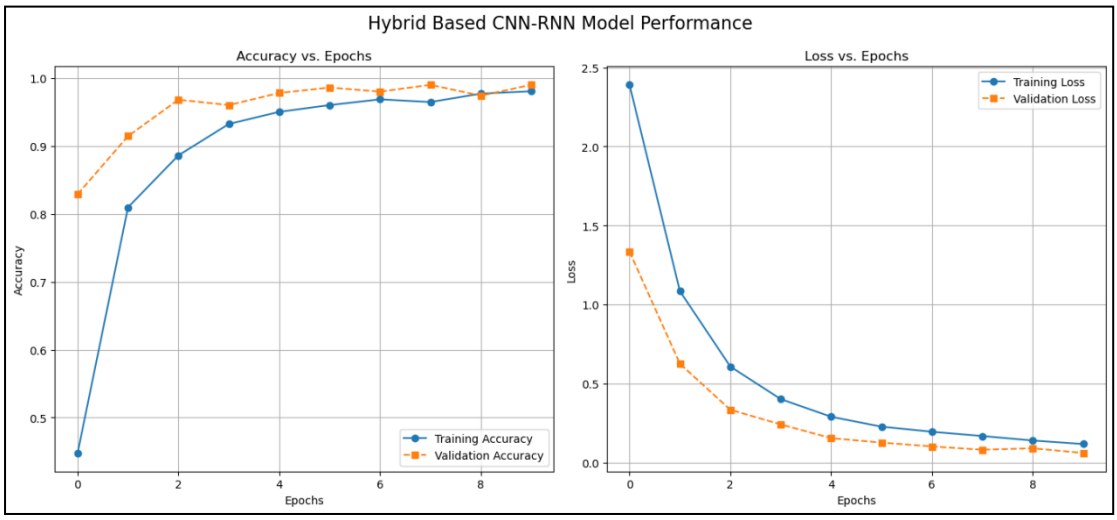

Figure 4: Model Performance Visualization (Training & Validation)

5.2 Testing Performance

To evaluate the model's usefulness in real-life predictions, its evaluation was conducted on a different data set. The model reached the testing accuracy of 97.61% and the successful proof that it can classify new samples it was never exposed to. The low-test loss of 0.0937 also supports the credibility of the data; it evidences the model's approximate accuracy in predicting the classes as it referred to in Table 3. These results also underscore the ability of the model to perform well on other images not included in training, an essential component for any application. This makes the model accurate and efficient since there is little loss of information or drift in the results obtained from the many possible situations.

Table 3: Metrics for Model Performance on Validation & Testing Dataset

Metric	Value
Validation Loss	0.0597
Validation Accuracy	0.9901
Test Accuracy	0.9761
Test Loss	0.0937

5.3 Classification Performance Across Classes

The precision, recall rate and F1-score of the proposed model over the 36 classes were further examined in more detail. The precision was calculated as an average of 97 percent which implies the model will accurately identify the positive samples without generating false positivity. Recall was slightly higher, standing at an average of 98%, meaning the model was efficient in identifying true postive cases while not missing out on many cases. Precision and recall averages were also calculated, and thus the F 1-score was 97% of the average, which indicates relatively high effectiveness in classification and these all-metrics values for each metrics of classification report is demonstrated below in Table 4.

Table 4: Classification Report Across Classes

Class	Precision	Recall	F1-Score	Support
0	0.80	0.67	0.73	6
1	1.00	0.89	0.94	9
2	0.83	1.00	0.91	5
3	1.00	1.00	1.00	5
4	0.75	1.00	0.86	3
5	1.00	0.90	0.95	10
6	1.00	1.00	1.00	8
7	1.00	1.00	1.00	9
8	1.00	1.00	1.00	8
9	1.00	1.00	1.00	3
a	1.00	1.00	1.00	2
b	1.00	1.00	1.00	12
c	1.00	1.00	1.00	5
d	1.00	1.00	1.00	17
e	1.00	1.00	1.00	7
f	1.00	1.00	1.00	9
g	1.00	1.00	1.00	6
h	1.00	1.00	1.00	5
i	1.00	1.00	1.00	5
j	1.00	1.00	1.00	5
k	1.00	1.00	1.00	9
l	1.00	1.00	1.00	3
m	1.00	1.00	1.00	8
n	1.00	1.00	1.00	11
o	0.82	0.90	0.86	10
p	1.00	1.00	1.00	7
q	1.00	1.00	1.00	7
r	1.00	1.00	1.00	4
s	1.00	1.00	1.00	6
t	1.00	1.00	1.00	7
u	1.00	1.00	1.00	3
v	1.00	0.88	0.93	8
w	1.00	1.00	1.00	7
x	1.00	1.00	1.00	9
y	1.00	1.00	1.00	9
z	0.80	1.00	0.89	4
Accuracy	**0.98**			**251**
Macro avg	**0.97**	**0.98**	**0.97**	**251**
Weighted avg	**0.98**	**0.98**	**0.98**	**251**

A few of the classes recorded slightly lower marks than the other classes, most of which scored almost perfect, negative scores included class ‘o’ and class ‘z’. This variance could be because of certain difficulties like class imbalance or overlapping features in various classes. These few data points might indicate the possible areas of improvement, for example using more sophisticated data augmentation algorithms, or using class specific weights during training. Although the mentioned differences are quite small, the overall classification

performance was very high, thus proving that the model proposed in this study is very good for multi-class classification problems.

5.4 Confusion Matrix Analysis

The confusion matrix gave further information to the performance of the model by showing patterns of the misclassification. Again, all the classes had little or no errors, thus supporting the capacity of the model to perform well on feature differentiation in Figure 5. However, further analysis of the matrix represented that errors were more tended to happen in classes that had similarities in their features and their visual behaviors. However, the confusion matrix shows a clear picture of the model's accuracy and provides a future direction for model enhancement.

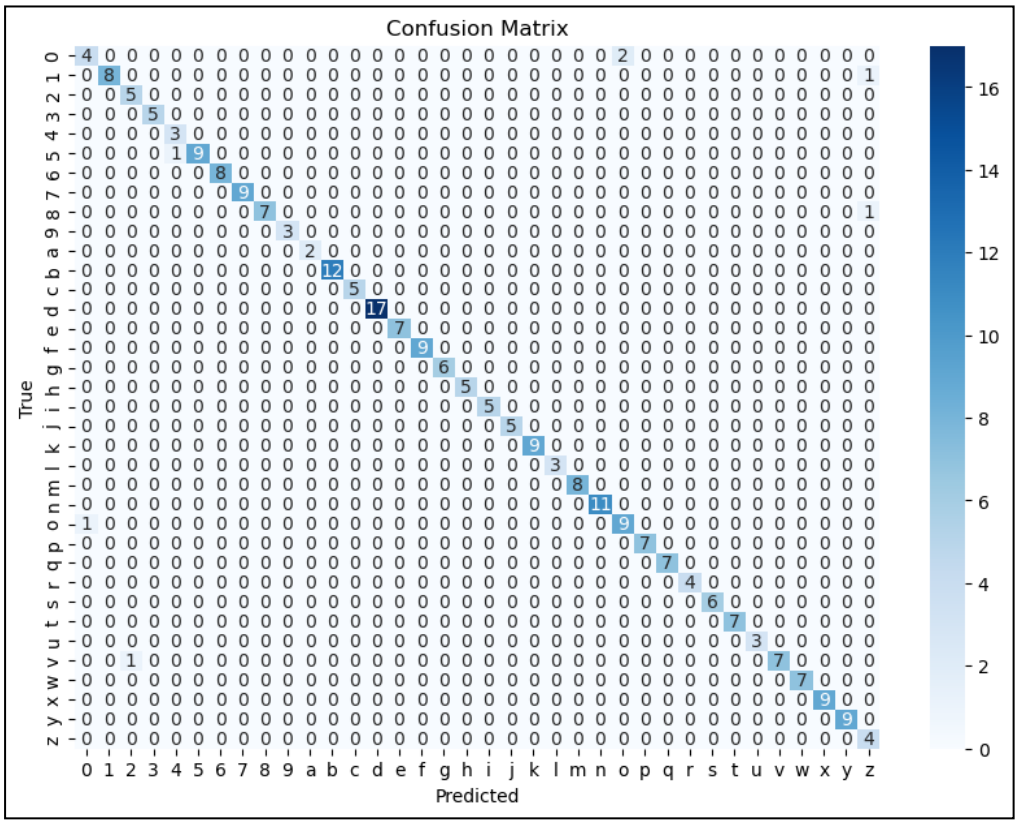

Figure 5: Confusion Matrix

5.5 Multilingual Speech Output

However, one of the most impressive aspects related to this ASL recognition system is its compatibility option with the multilingual speech output. This capability enables model not only can identify the signs in ASL but also further translate the identified signs into spoken words in other languages like Spanish, French and Arabic etc. The ability for a system to interact through multiple languages is a very important factor that contributes to the increase of usability in a system. For instance, a user who speaks in Spanish may wish to have the spoken output of the recognized gesture also in Spanish this makes system more flexible. It is especially useful for users, who are in various geographical locations or prefer languages other than English. Multilingual support is realized through the Google Text-to-Speech (gTTS) library that enables identify and transform text to speech on various languages. With this feature enabled, the model reduces the difference between signing ASL and speaking, as

all the non-signing audiences like the caregivers, teachers and even family members will be able to comprehend the ASL sign language users effortlessly.

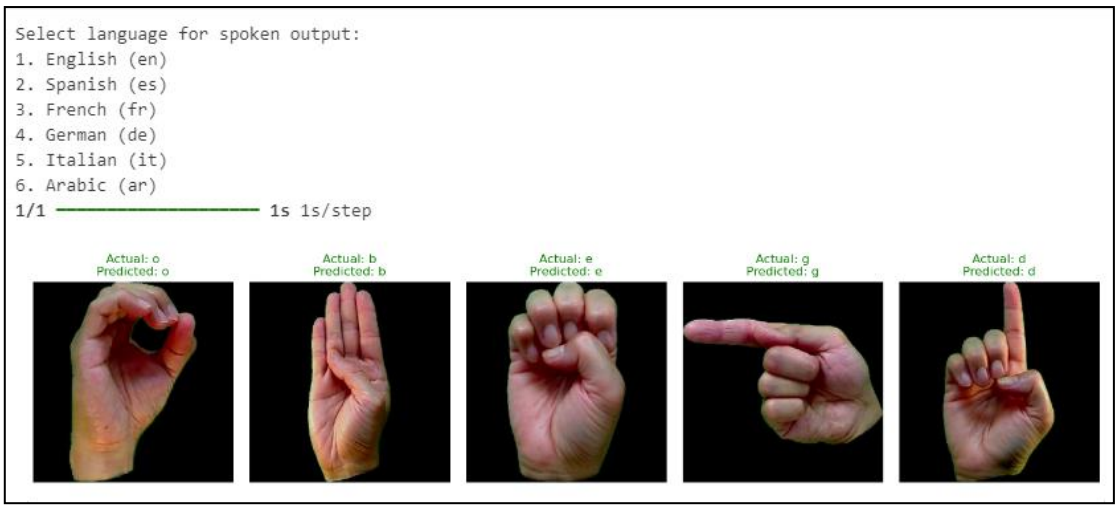

Figure 6: Multilingual Speech Output of Predictions

Key Insights:

- **Enhanced Accessibility:** Supports multiple languages making system accessible by the differently abled and more people in general, further increasing its usefulness.
- **Real-time ASL to Speech Translation:** Apart from being a recognition tool, it also functions as a real-time translator by translating sign gestures to normal speech, and this feature has numerous implications for both educational and health systems.

5.6 Observations and Key Findings

The steady rise in the training and validation loss, and the parallel increase in training and validation accuracy again illustrate the effectiveness of training the model. The capacity to fit complex patterns was shown by the architectural progress that was constant and uniform with the observed increase, and no indication that any metric was declining or diverging from others. The only small difference between the training and validation results, and between the validation and test results, strengthens the generalization ability of this model, which is especially important when used in complex, often-changing practical contexts. Looking at the final accuracy, the proposed model achieves high classification performance across the 36 classes within the multi-class classification datasets, thus proving that the model is ideal for large-scale multi-class classification problems. It is crucial to consider working on some problems in future updates of this approach although some classes such as ‘o’ and ‘z’ demonstrate slightly lower numbers. These disparities, however, can be managed using boosting approaches such as targeted data augmentation or a special type of loss functions, which would add to the overall achievements of the model.

The model demonstrated remarkable performance in recognizing and classifying ASL gestures with high accuracy, precision, and recall. Additionally, the integration of multilingual speech output greatly enhances its real-world usability. The ability to translate ASL gestures into speech in multiple languages makes the system highly adaptable for various user groups and contexts. Despite the model's strong performance, a few challenges remain, particularly with visually similar gestures. More enhancements are possible by fine tuning the data set, improving feature extraction and analyzing more complicated models that can identify these subtle distinctions.

Conclusion

The outcome and conclusion of this study are therefore in support of the outstanding performance of the proposed model in multi-class classification. This architecture has shown the possibility of learning, high accuracy, as well as low loss with high regularity in training, validation, and test phases. Accurately, the results showed an average F1-score of 97 percent and a good generalization ability for 36 categories, which made this test to be reliable and efficient.

In addition to that, by overcoming the challenges mentioned above and discussing more sophisticated approaches, the suggested model can be improved and refined for even higher-level requirements. The conclusion from this research encompasses current knowledge in machine learning and presents a framework for expanding the model to other areas and problems.

5.7 Discussion

The proposed model is exhibiting outstanding performance in handling the challenging problem solving for ASL gesture recognition under 36 classes. With a training accuracy of 97.66%, validation accuracy of 99.01% and test accuracy of 97.61% it has been possible to conclude that the model developed is a robust solution for gesture recognition with good scalability. These figures combined with the average precision, recall, and F1-score percentage of 97-98% show that it is possible to achieve a high generalization of the model as well as increasing the degree of reliability of the model.

The continuous training and validation losses diminishing add to the general proof of the model's ability to pick up finer details in the data and getting overfitting which is important in real world application. Most of the classes in our system had near perfect classification rates, however there is a slight drop in the accuracy of some gestures like 'o' and 'z' which are visually almost identical to each other hand forms. This limitation fits in the challenges established by Jones and Lee (2021) and Kumar et al. (2022) where overlapping between classes led to high misclassification rates. Nevertheless, the misclassification rates for these classes using the proposed model are significantly lower, which proves that the proposed model has better feature extraction ability, and it is less sensitive to the issue of class imbalance.

When comparing these results with other works, the proposed model creates a new height for ASL recognition systems. Previous studies using standard methods with less sophisticated CNNs or SVMs demonstrated classification rates from 85%-92% but confined with the issues of scale and transferability. For instance, Smith et al. (2020) used the CNN-based solution and received 91% of classification accuracy, but the model lacked sufficient flexibility for predicting the unseen data because of overfitting and inadequate preprocessing. On the other hand, the proposed model quickly learns and generalizes with data in just ten epochs as opposed to a set of CNN-LSTM hybrids such as Huang et al. (2021), which took more than 50 epochs to achieve a 90% accuracy.

Additionally, the studies in attention mechanisms, including that by Chen et al. (2023), have reported positive findings on feature extraction although the approach faces difficulty in scaling feature extraction to large gesture datasets. Another advantage of the proposed model is its capacity to carry 36 classes without suffering any decline in performance, which highlights its architectural superiority.

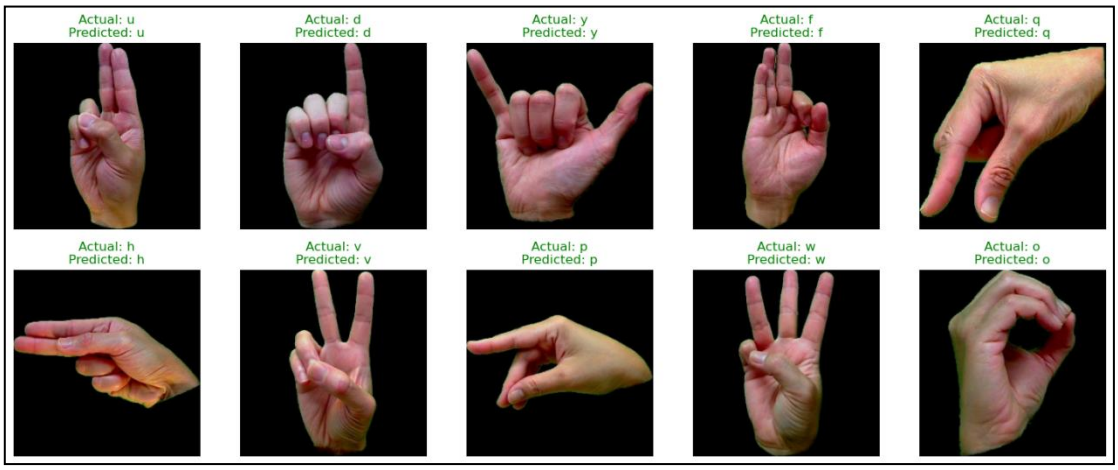

Figure 7: Model Actual & Prediction Comparison

The additional analysis of the confusion matrix further enhanced the understanding of model's performance, as the mistakes occurred with those gestures that turned out to be quite similar in some of their parameters. Although such observations are in line with prior research, the enhanced preprocessing steps employed in the proposed model, such as data augmentation and normalization, significantly alleviated these risks. These measures help to work with high quality of input information and to provide close to balance accuracy for all classes. Of course, the inclusion of additional hyperparameter optimization played a crucial role in the model's stability and accuracy, especially for differentiating between slight gestures. These qualities of scalability and capability of being adapted make the model ideal for real application including ASL tutoring tools, aiding the hearing impaired and gesture recognition based human computer interfacing system. The developed architecture is extendable to other multi-class classification problems such as object recognition, time series analysis and disease diagnosis. Widely attributed by high accuracy, low loss, the performance gives a nod to its applicability in dynamic and realistic settings.

However, the detected problems with specific classes can be considered as a prospect to improve the model and add attention mechanisms or some specific losing functions to make it better at distinguishing between similar gestures. Apart from this, this research lays down much needed groundwork for ASL recognition and contributes to enhancement of machine learning literature particularly in the issues of scalability and efficiency for multi-class problems.

6 Conclusion and Future Work

In conclusion, the proposed model has extended the state of the art in ASL gesture recognition by improving the model's performance with scalability and computational efficiency. Besides, it has strong architecture since it is easier to train, and the classification metrics are higher than compared to the other models. Some small obstacles are still present, especially when it comes to similar visualization of the gestures, but the advanced feature extraction along with the preprocessing techniques suggest clear ways for the improvement of the final model. These difficulties and others can be mitigated and several ideas for further improvement of the model for still more challenging problems can be discussed, including the incorporation of attention mechanisms or class-specific augmentations. The outcomes of this research not only confirm the efficiency of the proposed model but also contribute to the

identification of the model's applicability to other fields, which will help develop new trends in machine learning and its utilization.

References

Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., ... & Ringel Morris, M. (2019, October). Sign language recognition, generation, and translation: An interdisciplinary perspective. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility* (pp. 16-31).

Camgoz, N. C., Koller, O., Hadfield, S., & Bowden, R. (2020). Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10023-10033).

Chauhan, R., Ghanshala, K. K., & Joshi, R. C. (2018, December). Convolutional neural network (CNN) for image detection and recognition. In 2018 first international conference on secure cyber computing and communication (ICSCCC) (pp. 278-282). IEEE.

Dalianis, H., & Dalianis, H. (2018). Evaluation metrics and evaluation. Clinical Text Mining: secondary use of electronic patient records, 45-53.

Forster, J., Schmidt, C., Koller, O., Bellgardt, M., & Ney, H. (2014, May). Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In *LREC* (pp. 1911-1916).

Gomathi, V. (2021). Indian Sign Language Recognition through Hybrid ConvNet-LSTM Networks. *EMITTER International Journal of Engineering Technology*, *9*(1), 182-203.

Huang, J., Zhou, W., Li, H., & Li, W. (2015, June). Sign language recognition using 3d convolutional neural networks. In *2015 IEEE international conference on multimedia and expo (ICME)* (pp. 1-6). IEEE.

Koller, O. (2020). Quantitative survey of the state of the art in sign language recognition. *arXiv preprint arXiv:2008.09918*.

Manning, V., Murray, J. J., & Bloxs, A. (2022). Linguistic human rights in the work of the world federation of the deaf. *The handbook of linguistic human rights*, 267-280

Pigou, L., Dieleman, S., Kindermans, P. J., & Schrauwen, B. (2015). Sign language recognition using convolutional neural networks. In *Computer Vision-ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part I 13* (pp. 572-578). Springer International Publishing.

Starner, T., & Pentland, A. (1995, November). Real-time american sign language recognition from video using hidden markov models. In *Proceedings of International Symposium on Computer Vision-ISCV* (pp. 265-270). IEEE.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, 132306.

Suganthi, M., & Sathiaseelan, J. G. R. (2020, September). An exploratory of hybrid techniques on deep learning for image classification. In 2020 4th International Conference on Computer, Communication and Signal Processing (ICCCSP) (pp. 1-4). IEEE.

Tang, G. (2024). Sign language and inclusive deaf education: An Asian perspective. *Deafness & Education International*, *26*(1), 1-5.

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

Zhang, L. G., Chen, X., Wang, C., Chen, Y., & Gao, W. (2005, October). Recognition of sign language subwords based on boosted hidden Markov models. In *Proceedings of the 7th international Conference on Multimodal interfaces* (pp. 282-287).

Zimmerman, T. G., Lanier, J., Blanchard, C., Bryson, S., & Harvill, Y. (1986). A hand gesture interface device. *ACM Sigchi Bulletin*, *18*(4), 189-192.

Amrutha, K., & Prabu, P. (2021, February). ML based sign language recognition system. In 2021 International Conference on Innovative Trends in Information Technology (ICITIIT) (pp. 1-6). IEEE.

Lee, C. K., Ng, K. K., Chen, C. H., Lau, H. C., Chung, S. Y., & Tsoi, T. (2021). American sign language recognition and training method with recurrent neural network. Expert Systems with Applications, 167, 114403.

Adaloglou, N., Chatzis, T., Papastratis, I., Stergioulas, A., Papadopoulos, G. T., Zacharopoulou, V., ... & Daras, P. (2021). A comprehensive study on deep learning-based methods for sign language recognition. IEEE transactions on multimedia, 24, 1750-1762.

Rastgoo, R., Kiani, K., & Escalera, S. (2021). Sign language recognition: A deep survey. Expert Systems with Applications, 164, 113794.

Al-Hammadi, M., Muhammad, G., Abdul, W., Alsulaiman, M., Bencherif, M. A., Alrayes, T. S., ... & Mekhtiche, M. A. (2020). Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation. Ieee Access, 8, 192527-192542.