TALKSIGN: A REAL TIME AMERICAN SIGN LANGUAGE TEXT AND AUDIO INTERPRETER FOR VIDEO CONFERENCING

A Research Project

Presented to the Faculty of the
Computer Engineering Technology Department
College of Engineering Technology
Technological University of the Philippines Visayas
Capt.Sabi St.,Bgry. Zone 12,
Talisay City, Negros Occidental

by

ROSS MATHEW D. NEGRIDO
SEAN MATTHEW C. VILLALOBOS

In Partial Fulfillment of the Requirements for the Degree
Bachelor of Engineering Technology Major in
Computer Engineering Technology

June 2025

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

Background of the Study

American Sign Language (ASL) has long served as the primary means of communication for many deaf and hard of hearing individuals. The recent improvements in the digital space have highlighted the gaps of accessibility for these individuals. This problem often arises when other individuals lack the familiarity of ASL, leading to barriers in effective communication. As digital communication becomes more essential in work, education, and social settings, the absence of inclusive tools can lead to social exclusion and limited participation. In today's society there is a growing need to make and introduce these tools that can give a voice and help be more inclusive of deaf and hard of hearing individuals in the digital world.

Over the years, online video conferencing and digital communication platforms have proven themselves to be an essential tool, with society steadily increasing its reliance on these services. Despite their widespread use, the support for real-time ASL interpretation remains largely unavailable. Platforms such as Google Meet, Zoom, and Microsoft Teams. may incorporate text-based features like captions, but they do not address individuals that rely on visual gestures for input. Moreover, integration of gesture recognition with speech output is mostly unexplored. This leaves a gap in accessibility for sign language users in these scenarios or environments.

This study aims to develop a utility software tool that translates ASL gestures into both text and audible speech in real time. A webcam will serve as the primary input, enabling the system to capture gestures, process and generate the corresponding text and speech output. Integration of virtual camera and virtual microphone will enable the users to use the software as a source in various applications. The system will be designed to function as a utility tool, and will run in the background, it is developed to be used on video platforms such as Zoom, Google Meet, and Microsoft Teams.

The system will aim to display advancements of artificial intelligence and computer vision in recent years. Real-time gesture recognition has been one of the areas with great improvement, it has become more reliable and accurate. The system will try to make use of this new understanding to create a software tool that can improve accessibility for deaf and hard of hearing individuals in the virtual space. It bridges the gap of communication and contributes to the growing desire of the society to be inclusive and accessible for every individual.

The model will be trained on a selected set of vocabulary that is available in the American Sign Language, these will be the words that are commonly used in everyday interactions. Natural language processing (NPL) methods will be used to arrange and create syntactically correct phrases or sentences from the gathered words. The output will act as a virtual camera and microphone, enabling the use in third-party applications. Hand and body movements will be the main points tracked by the gesture recognition model; the model will exclude the recognition of facial expressions. The system will be tailored towards Windows-based systems and its use in the digital space, such as video meeting and conferencing.

Objectives of the Study

General Objectives

The main objective of this research is to develop an **American Sign Language (ASL)** translator system that uses **computer vision** and **deep learning**. It aims to assist individuals with speech and hearing impairments in online video calls. This system will provide an **inclusive communication platform** that translates ASL gestures into readable text and spoken words, improving accessibility and inclusivity in live video calls.

Specific Objectives

1. Design a **gesture recognition** module using a webcam and computer vision to detect and track hand movements accurately in real-time.
2. Develop a machine learning-based **translation model** that converts recognized American Sign Language (ASL) **gestures into contextually appropriate text**.
3. Integrate a **text-to-speech (TTS)** system to enable spoken output of the translated signs, promoting smoother interactions in video conferencing.
4. Build a user-friendly desktop software that is compatible with commonly used video call platforms.
5. Evaluate the system's accuracy, and latency by using it in video call platforms available for desktops (Google Meet, Zoom, Messenger, etc.).

Keywords: Text and Speech Sign Language Interpreter, Gesture Recognition, Computer Vision, Deep Learning, User-friendly Desktop Software

Scope and Limitations of the Study

The primary focus of this study is to develop a utility software tool that translates American Sign Language (ASL) gestures captured from a webcam into both text and audible speech. The system will utilize natural language processing (NLP) to convert individual gestures and their respective translations into semantically correct phrases and sentences, and will be mainly optimized primarily for Windows-based systems. The system will incorporate a virtual camera and microphone function, which enables the user to use the output in any video conferencing platforms. The vocabulary supported will be limited to the most common and used ASL words relevant to day-to-day communication. Gesture recognition will mainly focus on hand movements and exclude facial expressions or full-body motions.

Significance of the Study

This study will showcase the use of deep-learning and artificial intelligence to aid communities in the digital space. Real-time ASL translation will help bridge the communication gap between deaf and hard of hearing individuals and those who are unfamiliar with the language. The system aims to offer a platform towards the individuals who were excluded from mainstream digital interactions. It can enhance inclusivity and accessibility in virtual classrooms, meetings, and social interactions. The help of natural language processing will aid the translations to be contextually accurate, creating a more humanized experience.

Chapter 2

CONCEPTUAL FRAMEWORK

Review Of Related Literature and Studies

Introduction

According to the National Institute on Deafness and Other Communication Disorders (NIDCD, 2021), American Sign Language (ASL) is a fully developed, natural language that shares the same attributes and fundamentals as any spoken language. It is communicated through hand movements, facial expressions, and body language; it serves as the primary language of many deaf and hard-of-hearing individuals. American Sign Language has played a vital role in bridging the gap for communication. Its influence has expanded and opened opportunities across multiple industries including healthcare, entertainment, education, technology, and government, creating new and equal opportunities. The increasing demand for ASL interpreters in the professional field has also been a great example of the changing society opting to be accessible and inclusive—regardless of hearing ability (Pandya, 2023.).

Communication with Deaf and hard-of-hearing individuals traditionally relied on human interpreters, which can present more obstacles for both parties. The rapid advancement of technology —accelerated by the global pandemic—has highlighted its potential to help solve and break down communication barriers in classrooms, meetings, and various other settings. Alsharif et al. (2023) discuss that the recent developments also bring awareness to what can be improved of the gaps within our society in terms of accessibility and inclusivity. Advances in machine learning and artificial intelligence have impacted our day-to-day lives. Studies are being conducted to find lapses and try to improve the quality of life and foster an inclusive society for the deaf and hard-of-hearing individuals in the digital world.

Related Readings

Accessibility to assistive technology is essential in its development and its benefits to the people in the long run. The Office of the United Nations High Commissioner for Human Rights (OHCHR) created the Convention on the Rights of Persons with Disabilities (CPRD). According to the OHCHR (n.d.), the Convention aims to create an environment for people with disabilities wherein their rights and dignity are to be respected. It also promotes the development of assistive technology that can allow these people to interact with the world without their impairments hindering them in doing so.

Assistive technology is a way to use technology in aiding people with disabilities by allowing them to communicate and interact with the world the same way people without disabilities do. According to the NIDCD (2019), assistive technology is categorized into tree types: (1) assistive listening devices (ALD), which amplifies sound, (2) augmented and alternative communication (AAC) devices, which provides a person with alternative forms of communication, and (3) alerting devices, which utilizes signals for notifications. This study aims to create an AAC that will be able to detect human gestures and interpret it to generate text and audio translation to aid people with disabilities in a video call environment.

According to Rodríguez-Correa et al. (2023), assistive communication devices highlights 5 main categories of technologies: (1) *gesture recognition system* for sign language translation, (2) *sign language teaching tools,* (3) *automatic captioning system,* (4) *online content platforms,* and (5) *text/illumination networks*. In the study, a critical gap is revealed, although these types of technologies exist, there is still a clear disparity in technology access. In developing countries where economic, infrastructural and educational barriers limit adoption, having access to these technologies is a challenge. The study also highlights that there is a predominance of tools designed for hearing and a lack of visual-centric adaptations tailored for Deaf learners' needs. This study aims to tackle the issue by creating a system that allows for a audio and visual centric form of communication for an inclusive design that is also accessible for anyone with a desktop/laptop.

Related Literature

Text and Speech Sign Language Interpreter

Sign language is the best mode of communication for deaf and mute people as it allows them to communicate their thoughts and feelings through the use of gestures. However, they are still limited with the number they could communicate with since sign language is not naturally learned and used by regular people for their daily lives. Devices that allow universal communication through the use of different technologies started to rise. In a study by Najib (2024), The author presented a comprehensive survey of machine learning, image processing, artificial intelligence, and animation tools used in sign-to-text/speech and speech-to-sign interpretation. The paper highlighted stages of sign language interpretation, from video capture and preprocessing through feature extraction and classification, outputting either speech or textual meaning of the gesture. Hand gestures, facial expressions, and lip-reading devices were studied for translating continuous sign inputs into text and speech.

In a proposal study by Madahana et al. (2022), the researchers proposed AI-driven South African translators. The study investigates the use of AI for a real-time speech-to-text to sign language translation due to the COVID-19 limiting two-way communication between the hearing and hearing-impaired. The researchers conducted a systematic search across five major bibliographic databases (ScienceDirect, PubMed, Scopus, MEDLINE, and ProQuest) to find publications that discussed AI/ML solutions for speech-to-sign translations. The findings revealed a notable paucity of research and implementation of AI powered speech-to-sign systems, despite the challenges brought upon by the COVID-19 pandemic. The researcher proposed an AI-powered, real-time speech-to-sign solution made for South Africa’s eleven official languages, outlining its implementation and development in a roadmap. Concluding that a localized assistive technology is essential for enabling mutual communication between a hearing and a hearing-impaired person and bridging the identified implementation gaps.

A study by Banag (2023) where the development of sign language tutorial mobile applications through the use of animated 3D avatars and embedded quizzes. The application aims to help users learn Filipino Sign Language (FSL) alphabet, common phrases and thematic vocabulary. The functionality and usability of the application were evaluated by IT professionals, educators, parents, Deaf students, and hearing users. Across several criteria, the application scored the following in a 1-5 rating scale: (1) functionality (4.67), (2) usability (4.61), (3) reliability (4.61), (4) efficiency (4.57), and (5) portability (4.51), overall getting a rating of 4.59, which is equivalent to an "Excellent" resting. The application was able to meet its educational objectives and authors noted that the integration of other advanced technologies may benefit future developments for the application.

Gesture Recognition

A Study by Hosain et al. (2020) showed the new potential and improvements in the area of Gesture Recognition. In ASL recognition, hand shape is identified as the major aspect in gathering the data. The study approached the task of ASL recognizing differently than the conventional methods, aiming to improve its accuracy. The model uses Convolution Neural Networks (CNNs) for spatial feature extraction and Recurrent Neural Networks (RNNs) observing changes from the input in real-time. The research contributes to the improvement of systems designed to aid deaf and hard-of- hearing individuals; it provides a solid base for developing real-time ASL translation tools.

Locally, A study by Jucaban et al. (2022) developed an interactive mobile app that teaches Filipino Sign Language (FSL) with the use of gesture recognition. Sign for EverJuan is an android- based application with the goal to educate users of FSL. The app uses the device’s front camera as a source; the camera is used to capture gestures from the users to help aid the user for real-time practice. The application integrates MediaPipe, this is to track and extract points that serve as coordinates. Long Short-Term Memory (LSTM) neural uses these coordinates as inputs to accurately recognize the user’s performed gestures. The model was assessed and showed results that it is accurate and reliable in gesture recognition.

Among these studies, Convolutional Neural Network (CNN) seemed to have played a large role in recognizing and classification of sign language gestures. CNN classifies and recognizes objects through the use of three-dimensional data that are taken from these images (Ibm, 2025). Additionally, with the aid of MediaPipe’s landmark and hand tracking module, will allow the real-time tracking of a person’s hand (Google AI, n.d.).

Computer Vision

Computer Vision plays a big role in the development of AI technologies since it allows a device to recognize real-world objects and gather data from those objects (Ibm, 2021). A survey study by Chen et al. (2024) demonstrated a comprehensive review of over 100 publications from 1999 to 2024, focusing on the evolution of computer vision (CV) and visual simultaneous localization and mapping (SLAM) algorithms. CV has become a reliable technology due to the advancements in deep learning and hardware CV has also shown a notable impact in autonomous systems, pointing out how energy efficiency was improved through this technology. Optimization strategies such as pruning and quantization were tested to further improve the energy economy of autonomous systems like robots and factories. This simulated the trade-offs between algorithmic performance and energy consumption. The possibility of using visual SLAM algorithms in autonomous cars is also examined in the paper, highlighting the significance of striking a balance between processing demands and energy limits in practical applications.

Computer vision is like eyes for computers, it allows systems to recognize and process images (Ibm, 2021). The application of this technology to various industries can be essential to their development. A survey study by Sinha et al. (2023) looked at how computer vision benefits the field of data science and machine learning. It is capable of taking important information from images and coming up with computational models of real human eyes. Tools like OpenCV that enable the real-time processing of images and videos are important in supporting deep learning frameworks. Real-world applications of CV were also reviewed, showing that application of CV in several industries such as transportation, construction, healthcare, agriculture, and retail will allow these industries to flourish further as time goes by.

Using computer vision in any machine learning applications can be a challenging endeavor. In a book authored by Krig (2020), gave a structured description and matching methods on computer vision. The author explained several features in the interest-point detector and descriptor design. The book helps developers by sitting them within the broader theoretical landscape, contrasting community-driven OpenCV examples with deeper understanding on how algorithms work. The author also wrote about specialized topics such as search strategies, spectral analysis of descriptors, shape representation, distance metrics, and evaluation criteria to equip developers with the needed practical intuition when creating their own algorithms.

Deep Learning

In a review on sign language recognition using a convolutional neural network (CNN) by Ugale et al. (2023) the researchers show a comprehensive survey of CNN-based approaches for vision-based sign language recognition. The researchers structured their analysis around three core components: data acquisition methods, data environments, and hand-gesture representation techniques. The survey resulted in an accuracy range of 69% to 98%, averaging 88.8%, demonstrating strong performance achieved by the CNN architectures over classical image-processing pipelines. However, limiting factors recognized from the study may hinder it from real-world deployment. They discovered that datasets collected for this are in a highly controlled environment, which leads to poor generalization under unique conditions like variable lighting, backgrounds, and signer styles. As a way to step forward, the researchers encourage a more diverse sign-language corpora captured in “in-the-wild” settings to improve generalization. They also noted that the development of unified CNN+sequence-model architectures will allow it to handle continuous sign streams without manual processing.

Bhatia and Wadhawan (2019) presented the first systematic literature review of sign language recognition (SLR) systems. Covering research studies between 2007 and 2017. Their study revealed that SLR systems created to date are focused on camera-based recognition of static, isolated, single-handed signs. Neural networks also played an essential role for the classification approach of such systems with 65% of the studied systems achieving over 90% accuracy in recognizing sign language. The researchers noted that the lack of attention towards dynamic or continuous signing modes, multi-modal sensor setups, and signer-independent evaluation indicates gaps in robustness and real-world applicability.

Rajalakshmi and Kumar (2022), aimed to improve the most significant challenge in recognizing hand gestures continuously and accurately. The task on hand requires the model identifying the individual gestures from the user and the transitions between. The study proposed a deep learning approach that combines Convolutional Neural Network and Recurrent Neural Network, the pre-trained VGG16 CNN model is used for extracting visual patterns while the RNN is used to understand gestures and how they work over time. Their system not only challenges the continuous recognition but is designed to process and extract the information in real-time, it is tailored to capture both static gestures and dynamic transitions in between. The dual approach demonstrated a high accuracy and performance showing the systems potential, and how it can contribute in real-world scenarios and systems.

In a research article by Tipan et al. (2024), the researchers presented an SLR system that combines 2D CNNs with Transformers architectures to translate FSL into text. The system aims to explore Sign Language Translation to develop a much more robust model in the future. Due to the limitation of available datasets for FSL, translation for this sign language is a challenging task. That is why for this system, the researchers adapted pretrained Transformer models of high-resource language to learn the intricacies of low-resource language like FSL. The transformer model used was a Sign Language Transformer model that was trained on the ‘RWTH PHOENIX-Weather-2014T’ dataset, which is in German Sign Language. Analysis on the presented system showed that transfer learning allows training on low-resource datasets, however, data characteristics and distribution can affect the system’s performance.

User-friendly Desktop Software

A study conducted by Rastgoo et al. (2020) studied sign language recognition on computers. The authors noted that computers see sign language in two ways: (1) isolated, one sign at a time, and (2) continuous, one sign followed by another. The study also tackles the application front of SLR models where these pipelines are utilized. Common applications of this model are found in assistive mobile apps that work well by changing signs to words or sound with over 90% accuracy. They concluded SLR on computers can help people with disabilities communicate using their hands and other real-world applications such as video calls are possible.

Nathan, Hussain, and Hashim (2018) explored the usability of mobile applications designed and focused towards the deaf and hard-of-hearing community, the study revealed that the traditional approach and models often fall short in addressing needs unique to the community. The authors identified that good applications for people with speech and hearing disabilities need to be: (1) easy to learn, (2) fast to use, (3) easy to remember, (4) has little-to-no mistakes, and (5) appeals to the user. The study serves as a framework for assessing mobile applications and their usability, the literature supports the development of more inclusive and effective applications tailored towards the needs of the deaf and hard-of-hearing community.

In a local study conducted by Murillo et al. (2021), the researchers created a web-based system "Speak the Sign". The website changes the Filipino Sign Language (FSL) gestures it receives and turns them into words. To evaluate the systems, the researchers used purposive sampling for data gathering. They interviewed 30 respondents and there were 11 Special Education students, 10 Special Education teachers, and 14 non-disabled individuals. The goal of the data gathering is to gather data on the content, design, and functionality acceptability of the system via a rating-scale questionnaire and the interviewees gave the website a "Very Highly Acceptable" rating. The respondents liked the website mainly because it was easy to use and it is good at recognising signs. The author noted that the website needed more work, but it is already capable of helping people talk with signs

Related Studies

Text and Speech Sign Language Interpreter

New ways to understand sign language are important. Atri et al (2025) made a new way of approaching sign language translation technology. It goes straight from video to words, eliminating the need for intermediate gloss annotations and instead directly translates cuisign language video sequences into text using an Adaptive Transformer (ADTR) architecture. This architecture uses 3 main modules: (1) Adaptive Masking (AM), which takes out unnecessary video clips, (2) Local Clip Self-Attention (LCSA), which captures both local and spatiotemporal features through self-attention mechanism, and (3) Adaptive Fusion (AF), which makes the system robust and strong. The ADTR framework boasted state-of-the-art performance with a 15% increase in translation accuracy, processing efficiency, and real-time applicability when being evaluated using the ArabSign dataset. In conclusion, this new way of translating sign language is a potentially better way and it can also work well in real-life applications

Advancements in sign language interpreting technology demonstrated significant progress in AI and mobile technologies to tackle communication gaps for the deaf community. A study by Alday and Torres (2024), features the use of a mobile application using Kotlin in creating an application, as a Filipino sign language translator with CNN and MediaPipe. The application works by capturing live video frames through the phones built in camera. The app preprocesses the image to isolate hand landmarks, which are then fed through a trained CNN model for gesture classification and creates the corresponding translation and outputs it on the screen. The researchers achieved over 90% accuracy in classifying the image and the process can take under 200ms for it to create an output. The study shows the application's suitability for real-world communication scenarios, while demonstrating great potential for scalability.

Gesture Recognition

A study by Cui et al. (2019) proposes an end-to-end deep learning architecture that directly translates videos of continuous sign-language sentences into ordered gloss sequences. Traditional approaches in sign language translation rely on Hidden Markov Models. However, the proposed approaches make use of convolutional neural networks (CNN) with stacked temporal fusion layers for spatiotemporal feature extraction, followed by bidirectional recurrent neural networks for sequence modeling. After evaluating the proposed framework on two challenging continuous sign-language benchmarks, the proposed framework highlighted significant discoveries. The framework boasted a 15% relative improvement compared to prior state-of-the-art systems. It demonstrated both higher accuracy and robustness in real-world scenarios.

To address a challenge of sign language gesture classification in continuous sign language recognition systems, Eunice et al. (2023) introduced a posed-based Transformer model for word-level ASL recognition. Their system aims to improve the speed and accuracy of gloss prediction from dynamic video sequences. In order to do this, pose estimation was used instead of raw video frames, wherein a pose data is extracted from a single frame instead of using the whole frame for classification. This approach dramatically improved computational efficiency for the system while also maintaining a high classification performance from the WLASL dataset.

Jarabese et al. (2021) developed a real-time Filipino Sign Language (FSL) recognition system that converts them into speech. The study utilizes an inflated 3D Convolutional Neural Network (13D-CNN) to recognize the gestures. The model was trained with a dataset containing 237 video clips, featuring 20 different FSL gestures. The dataset was collected and augmented for training the model. The model achieved an accuracy of up to 95% in classifying and recognizing the different hand gestures. The study also explored the idea of implementing the model. Using Rapid Application Development (RAD) the authors were able to create and refine the application. The study where able to introduce a method that achieves great accuracy and explore the idea of implementing into a user-friendly application. It demonstrated the potential of the system and what it can provide for the deaf and hard-of-hearing community.

Computer Vision

Alsharif et al. (2023) a python-based system integrates computer vision and machine learning techniques in order to recognize hand gestures and accurately translate them into text. The system uses a webcam to continuously capture the hand movements, and processes with the help of OpenCV to detect and isolate the user's hands. The classification of each gesture is aided with the model which is trained with a gesture dataset that contains basic words and alphabets. The system achieved a 93.05% recognition accuracy, with limitations that the data should be in stable lighting conditions. The system explored the idea of real-time feedback which is important to deaf and hard-of-hearing individuals who rely on this type of system as their primary way of communicating. The system showcases how computer vision and machine learning have the potential to create accessible assistive applications and systems for the community.

The research paper featured in the title "Filipino Sign Language Hand Gesture Recognition Using MediaPipe and Machine Learning" presents the picture of the role of computer vision using OpenCV in a real-time recognition system for FSL gestures, where

Pilare et al (2024) are the authors. This is a program capable of identifying 27 sign language letters and three basic words, employing MediaPipe for hand and joint tracking and an LSTM algorithm for gesture classification. This system was developed by using a custom dataset containing 1,050 recordings from seven different signers which were utilized to provide an extensive set of data for training and evaluation by the authors.

Deep Learning

American Sign Language (ASL) fingerspelling was effectively and accurately recognized by the system developed by Arifiandi (2022). The study utilized a Convolutional Neural Network (CNN) approach, with the use of MobileNetV2 architecture. The model was trained on a dataset of 65,574 images that represent 24 static ASL alphabet signs, ensuring a wide range of diverse training samples. The model achieved a training accuracy of 99.60%, a validation accuracy of 98.66%, and a testing accuracy of 96.8%. This shows the effectiveness of lightweight CNN models, they can deliver highly accurate results in gesture recognition tasks when properly optimized and trained. The study also highlights the potential of deep learning architectures to develop sign language recognition systems that are accurate and used in real time.

In a way to empower the deaf community in healthcare communication, Bellil, Ghiri, and Boulesnane (2024) created a system that utilized 1D CNN in recognizing Algerian Sign Language. The authors used a pose-based approach in recognizing and classifying essential healthcare-related Algerian Sign Language gestures. They use MediaPipe's pose estimation framework to extract body landmarks and key points and used

1D-CNN deep learning architecture in training a model that achieved a 100% accuracy in classifying medical terms. The model demonstrated the effectiveness of landmark-based 1D-CNN models in real-world application.

Cayme et al. (2024) developed a real-time Filipino Sign Language (FSL) recognition system that uses convolutional neural network (CNN) for spatial feature extraction and a long short-term memory (LSTM) module to model temporal dynamics. The system was trained using a custom dataset of 2,100 images per class that covers a broad set of FSL gestures. The benchmarks in the system resulted in the model achieving validation and test accuracies of 97.62% and 97.29%, respectively, loss values were below 0.082. The study showcases the framework’s robustness in distinguishing visually similar signs, while also identifying residual errors in a few multi word phrases using confusion matrix.

User-friendly Desktop Software

This study aims to create an application that will allow the SLR in video conferencing (Zoom, Google Meet, etc.). A system created by Hautasaari et al. (2024) makes use of a virtual-camera to enable the user to capture live audio from their microphone to generate text captions on the user’s feed while in a call. The user is able to modify the text caption in their live feed as they please. The system pipeline of the application comprises (1) real-time ASR to transcribe spoken word, (2) a lightweight sentiment/emotion analyzer to tag each caption segment, and (3) a caption-rendering engine that overlays the text into the user’s live feed. The study bridges the application of

SLR systems into video conferencing, promoting real-world application for this type of technology.

In another another by Kim and Lee (2021) where they presented a solution to a problem that is not commonly addressed when it comes to video conferencing. There are occasions in video calls wherein participants may not be able to speak with their voice due to various factors. The system that the researchers developed will be able to address this problem by making a voice output communication aid (VOCA) for video conferencing which allows users to chat without sound.

A study was done by Tupal (2023) to tackle the challenges of limited high-quality datasets and scarce real-world applications in Filipino Sign Language (FSL). The first step in the study is to assemble a novel FSL video corpus comprising over 2,000 clips across 105 carefully selected "introductory" signs with the use of MediaPipe for joint-location extraction. The researcher evaluates graph convolutional networks (GCN) for frame-wise feature learning and gated current units (GRU) for sequence classification to model both spatial and temporal dimensions of signing. The study led to the development of an FSL E-learning desktop app that is powered by the top-performing MediaPipe-GRU model. The application was able to achieve a perfect 100% top-5 accuracy on newly created datasets. The author concluded that the study was not only able to create a model with such a strong classification result, but also the application was able to aid the effort in bridging the gap between SLR systems and real-world applications.

Synthesis and Justification

The related literature showcases the increase of interest in bridging the communication gaps between the deaf and the hearing communities, which is aided by the use of technological innovations. Both foreign and local studies focused on recognizing and converting gestures into text or speech. The researchers used different techniques such as Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), MediaPipe, OpenCV, and other machine learning models that are capable. Additionally, through the use of app-development technologies, easily accessible assistive apps through desktop and mobile devices demonstrated the real-world application of the Sign Language Translation system. These approaches have made significant strides in how we recognize hand gestures and signs accurately and efficiently. The studies showcase how effective machine learning is, serving as a base for systems related to language translations.

These technologies provide the researchers with the direction for real-world application, specifically for video conferencing. The review of related literature justifies the feasibility of creating a system that can recognize and translate ASL gestures into readable text and audible speech. Knowing that the sudden popularity of using video conferencing as a form of communication, assistive technology such as this will be able to give hearing and speech impaired people a platform to connect with others in the virtual space.

Theoretical Framework

This study is grounded in the following theories and concepts:

Sign Language Translation

Sign Language Translation (SLT) is the process of converting sign language gestures into text or speech using computational methods (Atri et al., 2025). Study from Alday and Torres (2024) demonstrated the direct translation of FSL gestures into readable text using the phone's built-in camera. From this study, leveraging translation models into video conferencing is essential for two-way communication.

Gesture recognition

Mediapipe Gestures Recognizer enables a system to recognize hand movements and gestures in real time and detects hand location with the help of landmarks (Google AI, n.d.). It operates on image data and operates with either static or continuous stream of data. With that in mind, the system's ability to recognize hand movements and gestures will allow it to identify which gestures translate to which word/phrase.

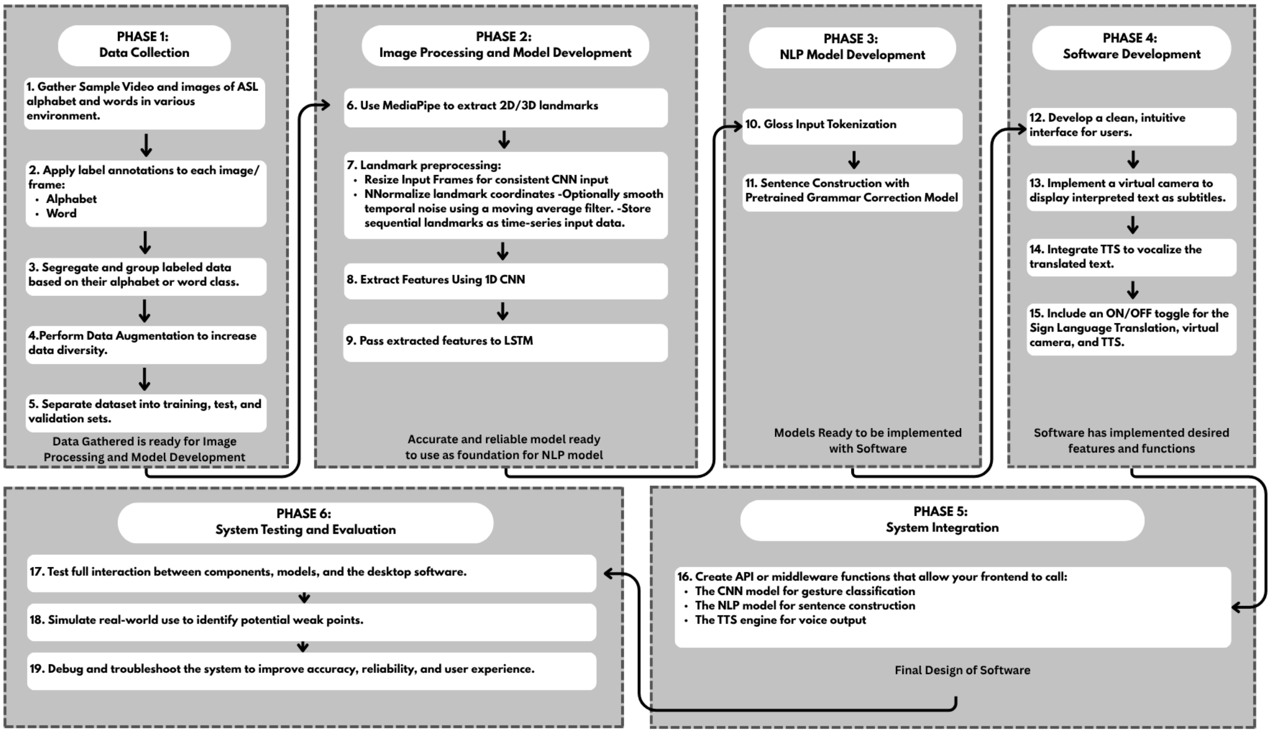
Computer Vision

Computer Vision enables the extraction of meaningful data from digital videos and images (Sinha et al., 2023). A model properly trained using the right datasets is capable of processing videos and images with high accuracy. In line with this, computer vision is essential in recognizing sign language and non-sign language gestures in the system.

Virtual Camera & Audio

Studies from Hautasaari et al. (2024) and Kim and Lee (2021) demonstrated the use of alternative means of communication during video conferences through the use of virtual camera and virtual audio (TTS), respectively. Virtual cameras can be used to act as visual aid for what the user wants to communicate with a microphone using readable text that can be seen from the user's fees. Virtual audio is used as a text-to-speech, where users can input the words, they wanna say and the application is going to convert those text into audible speech. In line with this, virtual camera dn virtual audio will enable the system to utilize sign-to-text and sign-to-speech translation in video conferencing.

Conceptual Model of the Study


PHASE 1:
Data Collection
1. Gather Sample Video and images of ASL alphabet and words in various environment.
2. Apply label annotations to each image/frame:
• Alphabet
• Word
3. Segregate and group labeled data based on their alphabet or word class.
4.Perform Data Augmentation to increase data diversity.
5. Separate dataset into training, test, and validation sets.
Data Gathered is ready for Image Processing and Model Development
PHASE 2:
Image Processing and Model Development
6. Use MediaPipe to extract 2D/3D landmarks
7. Landmark preprocessing:
• Resize Input Frames for consistent CNN input
• NNormalize landmark coordinates -Optionally smooth temporal noise using a moving average filter. -Store sequential landmarks as time-series input data.
8. Extract Features Using 1D CNN
9. Pass extracted features to LSTM
Accurate and reliable model ready to use as foundation for NLP model
PHASE 3:
NLP Model Development
10. Gloss Input Tokenization
11. Sentence Construction with Pretrained Grammar Correction Model
Models Ready to be implemented with Software
PHASE 4:
Software Development
12. Develop a clean, intuitive interface for users.
13. Implement a virtual camera to display interpreted text as subtitles.
14. Integrate TTS to vocalize the translated text.
15. Include an ON/OFF toggle for the Sign Language Translation, virtual camera, and TTS.
Software has implemented desired features and functions
PHASE 5:
System Integration
16. Create API or middleware functions that allow your frontend to call:
• The CNN model for gesture classification
• The NLP model for sentence construction
• The TTS engine for voice output
Final Design of Software
PHASE 6:
System Testing and Evaluation
17. Test full interaction between components, models, and the desktop software.
18. Simulate real-world use to identify potential weak points.
19. Debug and troubleshoot the system to improve accuracy, reliability, and user experience.

The theoretical concept shows how the development of the system will flow, from collecting the necessary data for model training to testing and troubleshooting the system for possible problems. The development starts with the gathering of the appropriate dataset for the model's training and development. This involves annotating, segregation, and augmentation of the gathered data. The system will have a pose-based approach in classifying sign language gestures; therefore, appropriate tools and methods are to be used in the second phase. Once the model could successfully classify the gestures, the sequence of glosses that come from the classification model will be passed through another pre-trained model that will turn the raw gloss sequence into a contextually correct sentence. To provide the user with an interface to control the system in, the researchers will also develop a desktop software that is equipped with the necessary tools to give the users full control over the system. To bring all the components together, both frontend and backend, APIs or middleware will be used to give the system a seamless communication with the frontend and backend of the system and an efficient flow of data within the system. Lastly, the system will undergo real-life testing to analyze its capabilities in a real-life scenario. Appropriate tools will be used to measure its performance and any arising problems from the system will be troubleshooted.

Definition of Terms

Conceptual Definition

- **Sign Language Interpreter:** Refers to the system designed to recognize ASL gestures and translate it into text and audio in video conferencing platforms.
- **Gesture Recognition:** Refers to the system's ability to detect and recognize continuous sign language gestures from a person.
- **Computer Vision:** Refers to the technological framework used to allow the system to perceive the user's body and hand movements through the computer's webcam.
- **Deep Learning:** Refers to the use of layered neural network architectures to create a model that can classify sign language gestures into its corresponding word.

- **Desktop Software:** Refers to a user interface developed to provide users with control over the translation system and its output.
- **Gloss:** A classified representation of the meaning of a sign in translation systems.
- **Pose Estimation:** The computer vision technique using MediaPipe of detecting and tracking body landmarks (e.g., hands, arms, face) in real time, used to extract meaningful motion features from signers.
- **1D Convolutional Neural Network:** Refers to a type of neural network that performs convolutions over sequential data (like pose key points over time) to extract temporal patterns from sign motions.
- **Long Short-Term Memory:** Refers to a type of Recurrent Neural Network (RNN) capable of learning long-term dependencies in sequential data, such as the order and timing of sign movements.
- **Natural Language Processing:** Refers to a model that generates grammatically correct and contextually relevant natural language output from a sequence of glosses.

Theoretical Definition

- **Sign Language Interpreter:** An automated or human system that translates between sign language and spoken or written language, enabling real-time communication across modalities (NCDHHS, 2013).
- **Gesture Recognition:** The computer-based process of detecting and interpreting human gestures from visual or sensor data, classifying them into meaningful symbolic or linguistic representations (ScienceDirect, 2023).

- **Computer Vision:** An area of artificial intelligence and image processing focused on enabling machines to "see" by extracting, analyzing, and understanding information from images or video (Ibm, 2021).
- **Deep Learning:** A branch of machine learning utilizing multi-layered neural networks to automatically learn hierarchical features from data (Google Cloud, n.d.).
- **Desktop Software:** A software application that installs and runs locally on a personal computer, offering direct access to hardware such as webcams and microphones with minimal latency (Christensson, 2022).
- **Gloss:** A written label used in linguistic transcription to represent individual signs (Lifeprint, n.d.).
- **Pose Estimation:** A technique within computer vision, it locates and tracks key landmarks of the body (e.g., hands, elbows, and face) from visual input in real time (Google Developers, n.d.).
- **1D Convolutional Neural Network:** A type of neural network architecture designed to process one-dimensional sequential input, such as time series or sequences (Imperial College London, n.d.).
- **Long Short-Term Memory:** A specific type of recurrent neural network designed to model long-range dependencies in sequential data through gated memory cells (GeeksforGeeks, 2025).
- **Natural Language Processing:** A subfield of computer science and artificial intelligence focused on enabling machines to understand, interpret, and generate human language using computational techniques (De Wolf, 2024).

Chapter 3

METHODOLOGY

Introduction

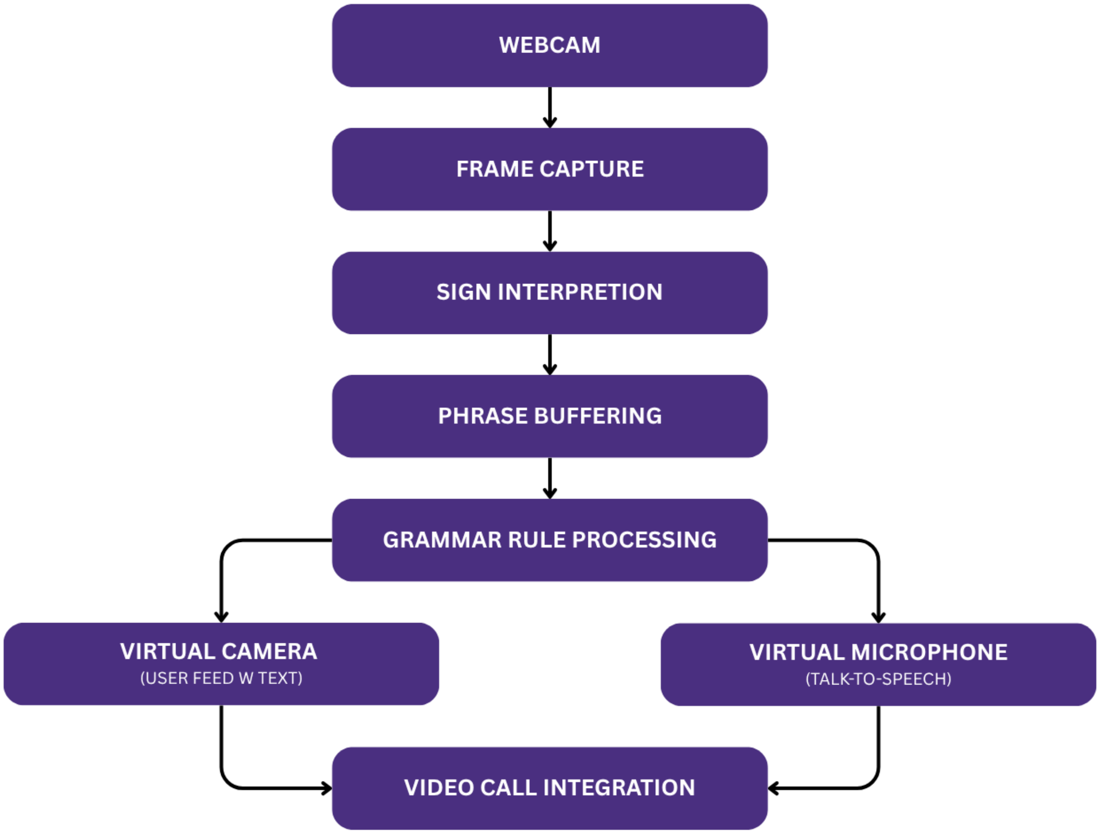


Figure 3.1. System Block Diagram

This research is going to use a two-stage pipeline for a real-time sign language translation for video conferencing. **Figure 3.1** shows the flow diagram of the system; the system receives continuous ASL gestures from a webcam. A trained image/video processing model will then be tasked in capturing and translating the corresponding signs. The translated words are then turned into contextually correct sentences. In order to apply this into video conferencing, the researchers developed a desktop software that makes use of a virtual camera and microphone. The virtual camera will allow the user to put captions on their feed of the translated signs and the virtual microphone will turn the translated text into speech for audio.

Project Development

Phase 1: Data Collection

The first phase of the system's development involves the collecting of a broad and robust dataset for training the sign language recognition model. The researchers will utilize both image and video datasets in training the model, therefore, the dataset will be taken from multiple sources to ensure its robustness. Also, to ensure the system's accuracy in real-world scenarios, the collected dataset must also reflect a variation in the signer's appearance, background, lighting, and signing speed. The data collected will be preprocessed to only extract the necessary data for the model's training.

Gather Sample Video and Images

This system aims to translate ASL alphabets, numbers, and words that are commonly used in everyday communication. With this in mind, the researchers will be getting their dataset from one of the biggest datasets for ASL words, 'WLASL' (Word-Level American Sign Language). WLASL is a large-scale video dataset that includes multiple signers, offering diversity in signing style, speed, and hand shapes. For numbers and alphabets, the datasets will be taken from the 'ASL Alphabet Dataset' and 'Sign Language for Numbers'.

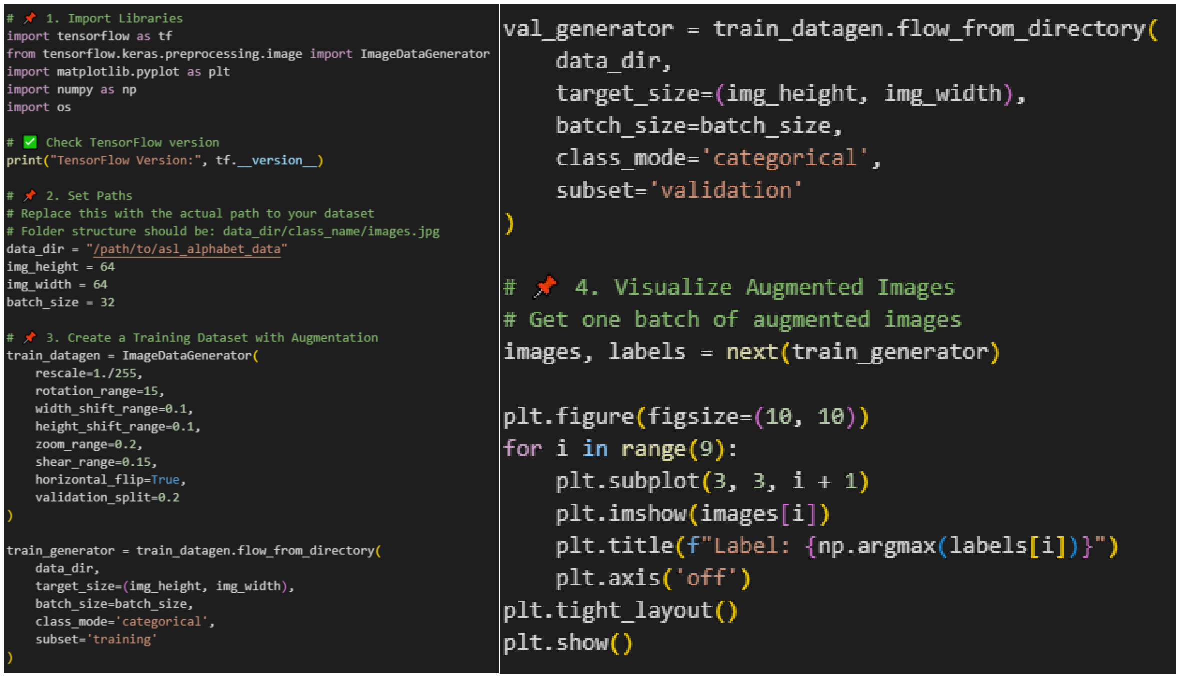
Label Annotation

To ensure an organized dataset, videos and images would be annotated since the dataset is taken from several sources. Accurately annotating labels into the dataset is essential for supervision of the model and enabling it to know the right mappings between visual gestures and the correct sign classification.

Segregation

The videos and images are to be grouped based on their corresponding categories (alphabet, word, number). In doing so, training the model will be smoother for the researchers as they won't have to bother with a disorganized set of data.

Data Augmentation



```
# 📌 1. Import Libraries
import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import matplotlib.pyplot as plt
import numpy as np
import os

# ✅ Check TensorFlow version
print("TensorFlow Version:", tf.__version__)

# 📌 2. Set Paths
# Replace this with the actual path to your dataset
# Folder structure should be: data_dir/class_name/images.jpg
data_dir = "/path/to/asl_alphabet_data"
img_height = 64
img_width = 64
batch_size = 32

# 📌 3. Create a Training Dataset with Augmentation
train_datagen = ImageDataGenerator(
    rescale=1./255,
    rotation_range=15,
    width_shift_range=0.1,
    height_shift_range=0.1,
    zoom_range=0.2,
    shear_range=0.15,
    horizontal_flip=True,
    validation_split=0.2
)

train_generator = train_datagen.flow_from_directory(
    data_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='training'
)
```

```
val_generator = train_datagen.flow_from_directory(
    data_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical',
    subset='validation'
)

# 📌 4. Visualize Augmented Images
# Get one batch of augmented images
images, labels = next(train_generator)

plt.figure(figsize=(10, 10))
for i in range(9):
    plt.subplot(3, 3, i + 1)
    plt.imshow(images[i])
    plt.title(f"Label: {np.argmax(labels[i])}")
    plt.axis('off')
plt.tight_layout()
plt.show()
```

Figure 3.2 Snippet Code for Data Augmentation

In order to create a model that is robust, data augmentation is applied to achieve this goal. Data augmentation involves rotating, flipping, brightness changes, and noise additions to help the model learn and have a better performance. **Figure 3.2** shows a sample code for augmenting the dataset.

Dataset Splitting

Dataset splitting is a standard practice in model training. It allows you to train, tune hyperparameters on unseen data, and evaluate the model on its generalization rigorously on the test set (Muktha, 2024). Standard split of a dataset into train, test, and validation subsets that is mostly done is: 70% training, 15% validation, and 15% test (ÇetiN, 2024).

Phase 2: Image Processing and Model Development

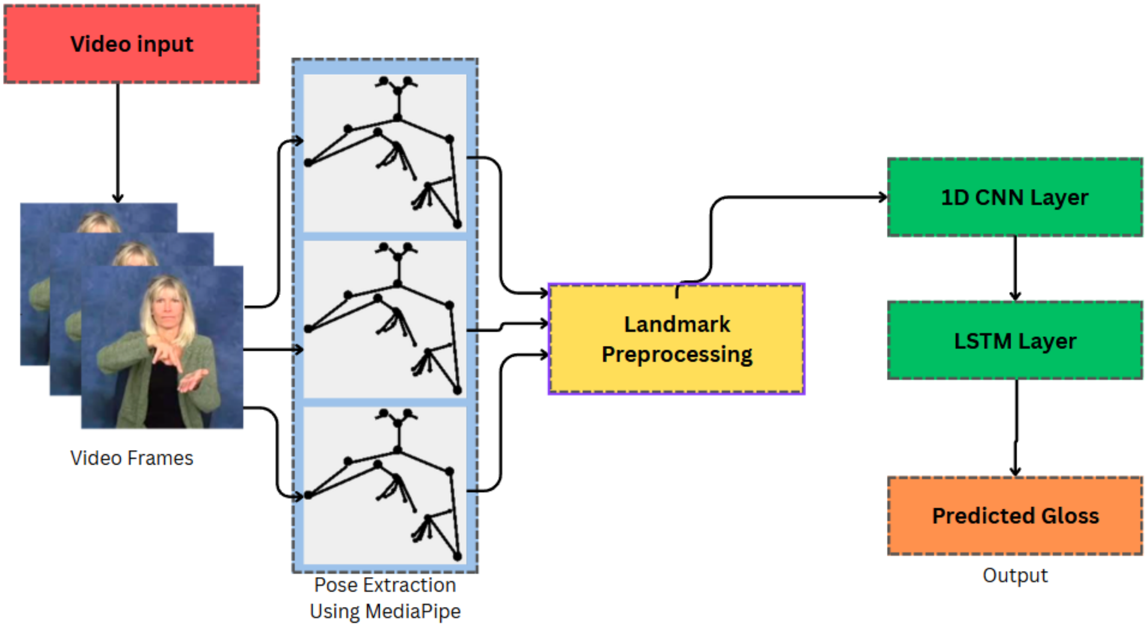


Figure 3.3. Overview of Sign-to-Gloss Classification Model

Video Frame and Pose Extraction images form: https://www.mdpi.com/1424-8220/23/5/2853#

This phase is where the development of a model that classifies continuous sign language into a sequence of glosses. The approach that the researchers will follow is a pose-based approach to sign language translation. A pose-based approach will allow the researchers to reduce the amount of data being processed since it only extracts the essential body, hand, and face key points. It is suitable for real-time applications like this, an SLT system for video conferencing.

Landmark Extraction

The system will utilize Google’s real-time framework, MediaPipe, in order to get the body and hand landmarks from each of the video frames. The landmarks represent essential joint or key point coordinates (like finger tips, wrist, elbows, etc.). A landmark-based representation will eliminate the need to process entire video frames, dramatically reducing the resource consumption from the system, making it lightweight and efficient for real-time applications like video conferencing.

Preprocessing

The extracted landmarks will be normalized to ensure consistency all throughout the video frames, regardless of the signer’s position, scale, or movements within it. Scaling the video frames uniformly and centering landmarks are also essential in this step. To reduce jitters caused by noisy key point detection, temporal smoothing may also be considered as a solution to the problem. The processed landmarks are then formatted as sequences over time, creating structured input data suitable for sequence modeling.

1D Convolutional Neural Network

The preprocessed landmarks are then passed through a 1D CNN to identify and learn short-term motion patterns. 1D CNN is used because unlike 2D CNN that is used to process spatial patterns in images, it operates over the temporal dimension, which is effective in recognizing trends in movements (such as finger flocks, directional transitions, etc.) (Jain, 2025). The features extracted from this will help capture nuanced motions and allow the model to distinguish different glosses.

Feature Classification Using LSTM

The temporal features from the 1D CNN will be passed and further processed using Long Short-Term Memory (LSTM) networks. This type of network specializes in learning long-range temporal dependencies (GeeksforGeeks, 2025). LSTM will analyze the full sequence of gestures to interpret continuous signs as glosses. If consecutive alphabet letters are detected, the group of letters are automatically merged into a single word to represent names or out-of-vocabulary words. Ultimately, the output is the classification corresponding to a gloss, which serves as the input for the next stage, converting the classified alphabets/words into a contextually correct sentence.

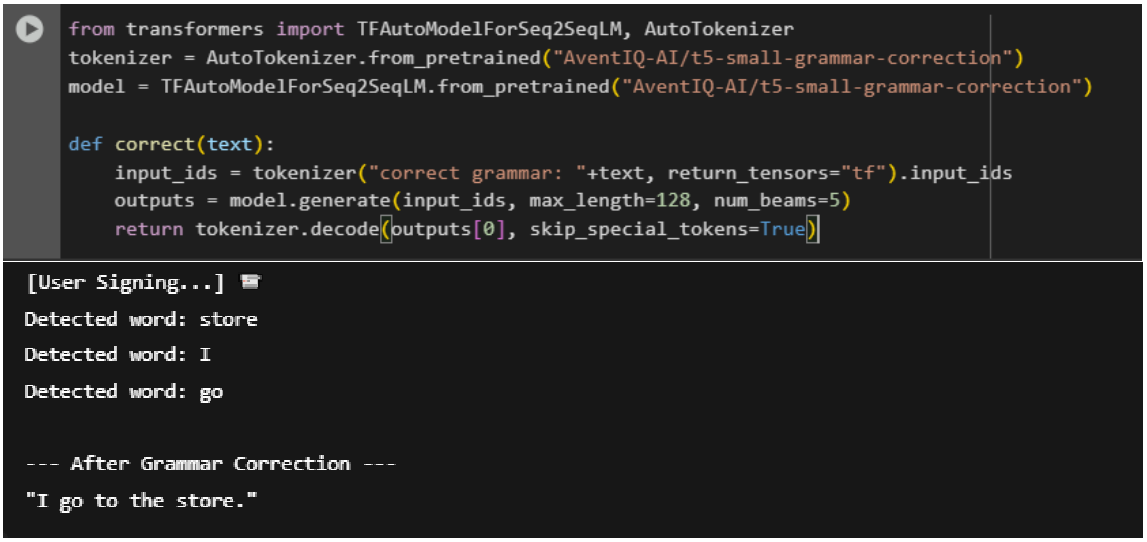
Phase 3: NLP Model Development

In this phase of the study, the raw gloss sequence is taken from the previous model and converted into a fluent, contextually correct sentence. Since the structural and grammatical differences of glosses and natural language are noticeably don't align with each other, This stage in the system's development will allow the user to send coherent sentences in video conferences.

Gloss Input Tokenization

This stage begins with the preprocessing and tokenization of the raw gloss sequence by converting the list of predicted gloss tokens into a single space-separated string that will become the input for the next step. This step involves basic normalization, such as lowecasing and removing unnecessary markers to ensure cleaner and more consistent model input. It helps standardize the data without changing its semantic content, essentially enabling more accurate downstream processing.

Grammar Correction Using Pretrained Model



```
from transformers import TFAutoModelForSeq2SeqLM, AutoTokenizer
tokenizer = AutoTokenizer.from_pretrained("AventIQ-AI/t5-small-grammar-correction")
model = TFAutoModelForSeq2SeqLM.from_pretrained("AventIQ-AI/t5-small-grammar-correction")

def correct(text):
    input_ids = tokenizer("correct grammar: "+text, return_tensors="tf").input_ids
    outputs = model.generate(input_ids, max_length=128, num_beams=5)
    return tokenizer.decode(outputs[0], skip_special_tokens=True)
```

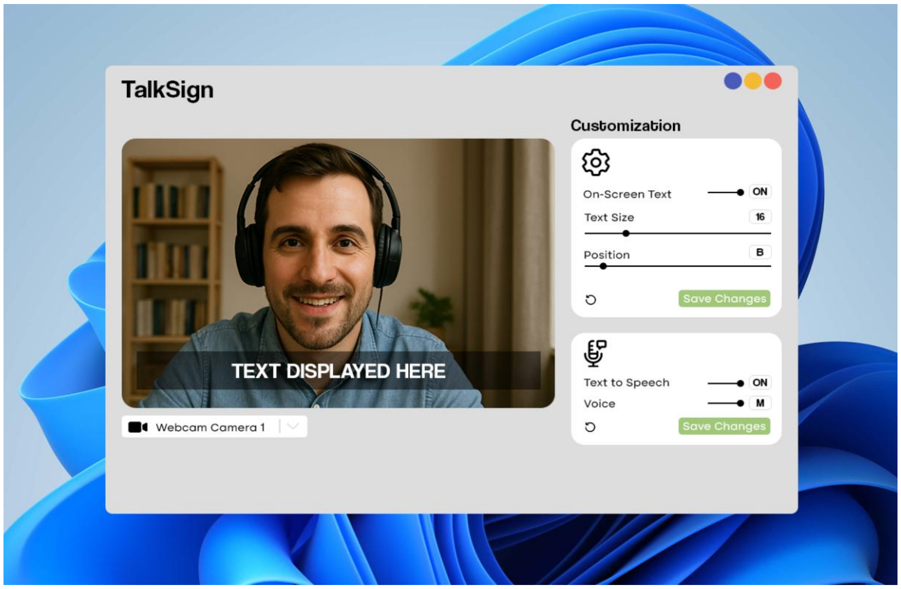
```
[User Signing...]
Detected word: store
Detected word: I
Detected word: go

--- After Grammar Correction ---
"I go to the store."
```

Figure 3.4. Sample code and output from pretrained model

The core of this phase is sentence constructing using a pre trained grammar correction model. In this study, the researchers will use the T5-small-grammar-correction model. This model is a lightweight transformer-based model that is designed to input grammatically incorrect and output corrected, natural English sentences. **Figure 3.4** shows a sample code of how the model will process and generate a well-structured sentence from the raw glosses. The researchers think this model is well suited for the study because the model does not require retraining, it is suitable for real-time translation because of its light weight, and performs well with short structured input.

Phase 4: Software Development



On-Screen Text ON
Text Size 16
Position B
Text to Speech ON
Voice M

Figure 3.5. Proposed Software Main Page

This system will run on the user's system and remain accessible during use. Figure 3.5. shows the draft layout of the software’s main page, where the user can customize how the output is displayed. This includes options to customize text size, text position, toggle virtual camera and microphone, and select a voice preset.

User Interface

This study’s second priority is to provide accessible and easy to use software. With this in mind, the software will have an easy-to-understand user interface. Designed using HTML, CSS, and javascript, the interface functions like a floating panel rather than a full application. This will minimize screen obstruction while providing a way to access and customize wanted features.

Virtual Camera Integration

The system will use a webcam; it will capture and extract ASL gestures and will serve as the system's primary input. After processing, the converted output will be displayed through a built-in virtual camera, enabling the software to function as a webcam source usable in third-party applications.

Text-to-Speech (TTS)

The system will incorporate Text-to-Speech (TTS), this feature outputs the translated ASL gestures into spoken words. Once a word, phrase, or sentence is constructed the system will generate audible speech from the text output. This audio will be delivered through a virtual microphone, which can be toggled on and off in the user interface based on the user preference. This feature will enhance the system's output giving more options and delivering a more natural interaction.

Toggle Functions

The system will offer the users various options that can be toggled to improve its accessibility and allow users to customize their experience. One of the functions will allow the users to adjust how the converted input be displayed or heard through the virtual camera and microphone. Options for virtual camera options will include text size, and placements, while the virtual microphone will have the option to toggle on and off and choose between preset voice profiles (male or female). These options are featured to accommodate different users based on their preferences.

Phase 5: System Integration

This stage of the system’s development will focus on unifying all developed components from the previous phases into a functional system that works seamlessly together. This phase involves creating APIs or middleware to facilitate the whole system's integration and establish communication between the frontend and the backend components. By accomplishing the goals of this phase, the system will establish a well-defined interface that also ensures modularity, scalability and efficient data flow.

Middleware/API Development

To enable a seamless communication and efficient data flow within the system, developing the appropriate APIs or middleware is essential in this step. These interfaces will allow the frontend of the system to call and utilize the essential components for the system to function as intended by the researchers.

Phase 6: System Testing and Evaluation

The focus of this phase is to test and ensure that the system and its components work seamlessly, this includes gesture classification model, NLP model, and the desktop software. The goal is to manufacture real-world scenarios to identify any potential weaknesses, inaccuracies, or points of improvements in the system. By conducting comprehensive tests, developers can evaluate the system's performance in terms of accuracy, reliability, and overall user experience. Any issues or bugs discovered during testing are addressed through debugging and troubleshooting, with refinements made to enhance the system's functionality. This phase will validate the system's readiness to be deployed and used by the target users, ensuring that the system meets the intended objectives of providing accurate and efficient sign language translation.

Integration Testing

Integration testing will aim to ensure that all the important components (user interface, virtual camera. gesture recognition, and text-to-speech) works together without problems. This step is where the researchers can identify any inconsistencies and communication issues between the components. Testing will focus on real-time performance, responsiveness, accuracy and stability particularly when being used in extended periods.

Simulated Real-World Use

Real-world usage will help uncover unresolved bugs and inconsistencies within the system. This step will help the researchers understand the system and evaluate to lessen errors before deployment. This step involves replicating common user environments and use cases, it can provide data on how the system operates under realistic load. Data in the areas of responsiveness, accuracy, and stability will help the researchers fix and improve the systems overall performance. This step can reveal potential issues that usually aren't evident during testing in controlled conditions.

Debugging and Troubleshooting

During the development process, testing, debugging and troubleshooting the researchers will be able to identify and fix issues that can cause errors or hinder the system’s performance. In this system, some bugs that can be encountered are misclassifications, errors in text-to-speech and integration failures between the components. The researchers will utilize the use of testing tools and logs to see and trace problems. In this stage the researchers will try to improve the system’s accuracy, responsiveness, reliability, stability and user experience.

REFERENCES

Alday, R., & Torres, J. C. (2024). *Sign Language Translator via Smartphone Image Analysis using Convolutional Neural Network.* Philippine E-journals. https://ejournals.ph/article.php?id=25780

Alsharif, B., Altaher, A. S., Altaher, A., Ilyas, M., & Alalwany, E. (2023). Deep learning-based systems for automatic sign language recognition: A review. *Sensors*, *23*(18), 7970. https://www.mdpi.com/1424-8220/23/18/7970

Arifiandi, W. (2022). *Kestrel: American Sign Language fingerspelling translator based on machine learning and CNN* (Bachelor's thesis, Institut Teknologi Sepuluh Nopember). ResearchGate. https://www.researchgate.net/publication/359082886_Kestrel_American_Sign_Language_Fingerspelling_Translator_Android_Application_Based_On_Machine_Learning_Convolutional_Neural_Network_and_TensorFlow

Atri, Y. S. &. S. B. &. S. M. a. &. A. a. a. &. M. (2025). Adaptive Transformer-Based Deep Learning framework for continuous sign language recognition and translation. *ideas.repec.org*. https://ideas.repec.org/a/gam/jmathe/v13y2025i6p909-d1608156.html

Banag, C. T. (2023). DEVELOPMENT OF SIGN LANGUAGE TUTORIAL MOBILE APPLICATION FOR FILIPINOS. *International Journal of Research in Education Humanities and Commerce*, *04*(02), 124–131. https://doi.org/10.37602/ijrehc.2023.4213

Bhatia, P., & Wadhawan, A. (2019). Sign Language Recognition Systems: A Decade Systematic Literature Review. *ResearchGate*. https://www.researchgate.net/publication/353571514_Sign_Language_Recognition_Systems_A_Decade_Systematic_Literature_Review

Cayme, K. J., Retutal, V. A., Salubre, M. E., Astillo, P. V., Cañete, L. G., & Choudhary, G. (2024). Gesture Recognition of Filipino Sign Language Using Convolutional and Long Short-Term Memory Deep Neural Networks. *Knowledge*, *4*(3), 358–381. https://doi.org/10.3390/knowledge4030020

ÇetiN, K. R. (2024, November 18). The Importance of Splitting Datasets into Training, Validation, and Test Sets. Medium. https://ruveydakardelcetin.medium.com/the-importance-of-splitting-datasets-into-training-validation-and-test-sets-417caaeae91d

Chen, L., Li, G., Xie, W., Tan, J., Li, Y., Pu, J., Chen, L., Gan, D., & Shi, W. (2024). *A Survey of Computer Vision Detection, Visual SLAM Algorithms, and Their Applications in Energy-Efficient Autonomous Systems. Energies,* 17(20), 5177. https://doi.org/10.3390/en17205177

Christensson, P. (2022, February 19). *Desktop software*. https://techterms.com/definition/desktop_software

Cui, R., Liu, H., & Zhang, C. (2019). A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training. *IEEE Transactions on Multimedia*, *21*(7), 1880–1891. https://doi.org/10.1109/tmm.2018.2889563

De Wolf, M. (2024, November 5). *NLP (Natural Language Processing)*. Techopedia. https://www.techopedia.com/definition/natural-language-processing-nl

Empowering Deaf Community in Healthcare Communication: 1D-CNN-Based Algerian Sign Language Recognition System. (2024). In L. Bellil, M. G. Ghiri, & A. Boulesnane (Eds.), Proceedings of the 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS 2024). IEEE. https://ieeexplore.ieee.org/document/10541233

Eunice, J., J, A., Sei, Y., & Hemanth, D. J. (2023). Sign2Pose: A Pose-Based approach for gloss prediction using a transformer model. *Sensors*, *23*(5), 2853. https://doi.org/10.3390/s23052853

GeeksforGeeks. (2025, May 28). What is LSTM Long Short Term Memory? GeeksforGeeks. https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/

Google AI. (n.d.). *Gesture Recognition Task Guide*. Google AI for Developers. https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer

Google Cloud. (n.d.). What Is Deep Learning? Applications & Examples. https://cloud.google.com/discover/what-is-deep-learning

Google Developers. (n.d.). *Pose landmark detection guide*. Google AI for Developers. Retrieved June 9, 2025, from https://developers.google.com/mediapipe/solutions/vision/pose_landmarker

Haritha, N., Vashnavi, A., & Rajesh, G. (2025). *Python-based real-time sign language interpreter using computer vision and machine learning*. ResearchGate. https://www.researchgate.net/publication/391394231

Hautasaari, A., Aramaki, M., Chujo, R., & Naemura, T. (2024). EmoScribe Camera: A Virtual Camera System to Enliven Online Conferencing with Automatically Generated Emotional Text Captions. *ACM Digital Library*, 1–7. https://doi.org/10.1145/3613905.3650987

Hosain, A. A., Santhalingam, P. S., Pathak, P., Rangwala, H., & Kosecka, J. (2020). *FineHand: Learning hand shapes for American Sign Language recognition*. IEEE. https://doi.org/10.1109/FG47880.2020.00062

Ibm. (2021, July 27). Computer Vision. *What is computer vision?* https://www.ibm.com/think/topics/computer-vision

Ibm. (2025, April 16). Convolutional Neural Networks. *What are convolutional neural networks?* https://www.ibm.com/think/topics/convolutional-neural-networks

Imperial College London. (n.d.). *Convolutional 1D network classification* [Web page]. ReCoDE – AI For Patents. Retrieved June 2025, from https://imperialcollegelondon.github.io/ReCoDE-AIForPatents/5_Convolutional_1D_Network_Classification/

Jain, A. (2025, January 7). Understanding the 1D convolutional layer in deep learning. Medium. https://medium.com/@abhishekjainindore24/understanding-the-1d-convolutional-layer-in-deep-learning-7a4cb994c981

Jarabese, M. B. D., Marzan, C., Boado, J. Q., & Lopez, R. R. M. F. (2022). *Sign to speech convolutional neural network-based Filipino sign language hand gesture recognition system*. ResearchGate. https://www.researchgate.net/publication/357287630_Sign_to_Speech_Convolutional_Neural_Network-Based_Filipino_Sign_Language_Hand_Gesture_Recognition_System

Jaucian, S. A., Polano, K. M., & Divinagracia, R. D. (2023). Hand gesture detection technology using Raspberry Pi converting hand signals into Tagalog words. *International Journal of Computer Science and Research*, *10*(9), 243–249. https://stepacademic.net/ijcsr/article/view/618

Jucaban, J. P., Requina, M. M., Duhaylungsod, K. B., Nalda, K. S., & Bermudez, D. T. (2022). *Sign For Everyjuan: An interactive Android application that teaches Filipino Sign Language using hand gesture recognition*. ResearchGate. https://www.researchgate.net/publication/382811335

Kim, W., & Lee, S. (2021). "I Can't Talk Now": Speaking with Voice Output Communication Aid Using Text-to-Speech Synthesis During Multiparty Video Conference. *2024 8th International Conference on Natural Language Processing and Information Retrieval (NLPIR '24)*, 1–6. https://doi.org/10.1145/3411763.3451745

Krig, S. (2020, June 22). *Computer Vision Metrics: Survey, Taxonomy, and Analysis*. PHL CHED Connect - We Educate as One. https://phlconnect.ched.gov.ph/content/view/computer-vision-metrics-survey-taxonomy-and-analysis

Kumar, R., Singh, S. K., Bajpai, A., & Sinha, A. (2023). Mediapipe and CNNs for Real-Time ASL Gesture Recognition. *arXiv*. https://arxiv.org/pdf/2305.05296

Lifeprint. (n.d.). Gloss in American Sign Language (ASL). https://lifeprint.com/asl101/topics/gloss.htm

Madahana, M. C., Khoza-Shangase, K., Moroe, N., Mayombo, D., Nyandoro, O., & Ekoru, J. (2022). *A proposed artificial intelligence-based real-time speech-to-text to sign language translator for South African official languages for the COVID-19 era and beyond: In pursuit of solutions for the hearing impaired.* South African Journal of Communication Disorders, 69(2). https://doi.org/10.4102/sajcd.v69i2.915

Muktha, D. (2024, November 23). Train, test, Validate: How data splits make or break machine learning models. Medium. https://medium.com/%40darshini.muktha/train-test-validate-how-data-splits-make-or-break-machine-learning-models-06001f69f1e9

Murillo, S. C. M., Villanueva, M. C. a. E., Tamayo, K. I. M., Apolinario, M. J. V., Lopez, M. J. D., & Edd. (2021, August 13). *Speak the Sign: a Real-Time Sign language to text converter application for basic Filipino words and phrases*. https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/92

Najib, F. M. (2024). *Sign language interpretation using machine learning and artificial intelligence.* Neural Computing and Applications. https://doi.org/10.1007/s00521-024-10395-9

Nathan, S. S., Hussain, A., & Hashim, N. L. (2018). *Usability evaluation of DEAF mobile application interface: A systematic review*. *Journal of Engineering and Applied Sciences, 13*(2), 291–297. https://www.researchgate.net/publication/323837037_Usability_evaluation_of_DEAF_mobile_application_interface_A_systematic_review

National Institute on Deafness and Other Communication Disorders. (2021, October 29). *American Sign Language*. National Institutes of Health. https://www.nidcd.nih.gov/health/american-sign-language

NCDHHS. (2013, September 1). What Is a Sign Language Interpreter. https://www.ncdhhs.gov/documents/files/what-sign-language-interpreter

NIDCD. (2019, November 12). *Assistive Devices for People With Hearing, Voice, Speech, or Language Disorders.* https://www.nidcd.nih.gov/health/assistive-devices-people-hearing-voice-speech-or-language-disorders

OHCHR. (n.d.). *Convention on the Rights of Persons with Disabilities*. https://www.ohchr.org/en/instruments-mechanisms/instruments/convention-rights-persons-disabilities

Pandya, M. (2023, September 25). *The impact of ASL interpreting on the workforce*. Unspoken ASL. https://www.unspokenasl.com/aslblogs/the-impact-of-asl-interpreting-on-the-workforce/

Pilare, L. K. M., Mahinay, J. C. A., Degamo, A. C. C., & Piner, B. N. C. (2024, August 31). *Filipino Sign Language Hand Gesture Recognition Using MediaPipe and Machine Learning*. https://stepacademic.net/ijcsr/article/view/618

Rajalakshmi, S., & Kumar, R. (2022). Gesture recognition using CNN and RNN. *International Journal for Research in Applied Science and Engineering Technology (IJRASET),* 10(9), 243–249. https://www.researchgate.net/publication/363669111_Gesture_Recognition_using_CNN_and_RNN

Rastgoo, R., Kiani, K., & Escalera, S. (2020b). Sign Language Recognition: A Deep Survey. *Expert Systems With Applications*, *164*, 113794. https://doi.org/10.1016/j.eswa.2020.113794

Rodríguez-Correa, P. A., Valencia-Arias, A., Patiño-Toro, O. N., Díaz, Y. O., & De La Puente, R. T. (2023). *Benefits and development of assistive technologies for Deaf people's communication: A systematic review. Frontiers in Education, 8*. https://doi.org/10.3389/feduc.2023.1121597

ScienceDirect. (2023). Gesture Recognition. https://www.sciencedirect.com/topics/computer-science/gesture-recognition

Sinha, N. K., Minj, J., & Patre, P. (2023). *A literature survey on computer vision towards data science and machine learning. International Journal of Current Science*. International Journal of Current Science. https://rjpn.org/ijcspub/papers/IJCSP23C1082.pdf

Tipan, L. K. A., Abalos, A. M., Bondoc, A. E., To, J. J., & Rivera, J. P. (2024). Filipino Sign Language Translation through Transfer Learning. *2024 8th International Conference on Natural Language Processing and Information Retrieval (NLPIR '24).*, 212–217. https://doi.org/10.1145/3711542.3711557

Tupal, I. J. L. (2023, April). *Recognizing Filipino sign language video sequences using deep learning techniques*. Animo Repository. https://animorepository.dlsu.edu.ph/etdm_ece/25/

Ugale, M., Shinde, O. R. A., Desle, K., & Yadav, S. (2023). A Review on Sign Language Recognition Using CNN. *Advances in Computer Science Research*, 251–259. https://doi.org/10.2991/978-94-6463-136-4_23

Appendix A

MATRIX OF RELATED LITERATURE

Authors	Text & Speech Sign Language Interpreter	Gesture Recognition	Computer Vision	Deep Learning	User-friendly Desktop Software
Najib (2024)	Sign language translation is made possible with the help of artificial Intelligence (AI)				
Jucaban et al. (2022)		MediaPipe is a commonly used tool for gesture recognition			
Sinha et al. (2023)			Computer vision enabled real-time processing of images and videos.		
Rajalakshmi and Kumar (2022)				A combination of CNN and RNN layers made continuous sign language	

				recognition possible.	
Nathan, Hussain, and Hashim (2018)					A good application should be easy to use and good at doing its task in order to appeal to the user.
Negrido & Villalobos (2025)	Deep learning architectures enabled the development of SLT systems	MediaPipe is capable of extracting essential landmarks from the whole body	Computer Vision enables the computer to simulate the human’s ability to see.	CNN and RNN architectures can classify continuous sign language.	Desktop software is used to give the user control over the system through an interface.

Appendix B

MATRIX OF RELATED STUDIES

Authors	Text & Speech Sign Language Interpreter	Gesture Recognition	Computer Vision	Deep Learning	User-friendly Desktop Software
Alday and Torres (2024)	Demonstrated real-world application of SLT systems through mobile application				
Eunice et al. (2023)		Showcased a pose-based SLT system from extracted pose data using MediaPipe			
Alsharif et al.(2023)			Used OpenCV to detect and isolate user's hand for real-time input		
Bellil, Ghiri, and Boulesnane (2024)				Created an Algerian Sign Language Recognition model using 1D-CNN	

				deep learning architecture.	
Tupal (2023)					Created a E-learning desktop application for Filipino Sign Language (FSL)
Negrido and Villalobos (2025)	Create a deep-learning system that can translate ASL into text and audio for video conferencing.	Use MediaPipe for pose estimation in a pose-based approach to classifying ASL gestures.	OpenCV will allow the system to take real-time video inputs from a webcam.	Use 1D-CNN and LSTM layers to classify continuous ASL.	Create easy-to-use python-based desktop software for user interface.

Appendix B

MATRIX OF METHODS

Stages	Research Settings	Respondents/ Objects in Study	Data Gathering Procedures	Data Gathering Instruments	Statistical Analysis
Phase 1: Data Collection	This phase will be done in the researchers' household	Videos of signers signing ASL words and images of alphabetical and numerical ASL sign	Search for images/video for: - 50-100 ASL words - all ASL alphabets - all ASL numbers Organize the gathered dataset through labeling, augmentation, and splitting into train, test, and validation	Utilize open-source datasets for all ASL videos and images Python scripts for data augmentation	None
Phase 2: Image Processing and Model Development	This phase will be done in the researchers' household	SLT model Predicted Glosses	Extract pose data from corresponding video frames and images. Create a model that predicts the	Jupyter Notebook and Tensorflow framework	Training accuracy and loss Validation accuracy and loss F1-score

			corresponding gesture		
Phase 3: NLP Model Development	This phase will be done in the researchers' household	Constructed Sentences	Use a pre trained NLP model for constructing a contextually correct sentence from the predicted models	Jupyter Notebook and Tensorflow framework	none
Phase 4: Software Development	This phase will be done in the researchers' household	Desktop Software	Create a frontend user interface for system control	VS Code or any source code editing platforms	none
Phase 5: System Integration	This phase will be done in the researchers' household	Completed System	none	none	none
Phase 6: System Testing and Evaluation	This phase will be done in the researchers' household	ASL translation system performance in in video conferencing platforms	Measure computer resource usage, accuracy, and latency Conduct trials from signing different	Profiling tools, manual observations	Descriptive Statistics

			sentences constructed from the recognizable ASL words		