Enhancing Communication with Text-to-Sign and Sign-to-Text Conversions Utilizing Digital Image Processing

Ananya Aggarwal[1], Aditya Rastogi[2] and Peehu Khandelwal[3]

[1]School of Engineering and Technology, BML Munjal University, Haryana, India (122413)
ananya.aggarwal.22cse@bmu.edu.in
[2] School of Engineering and Technology, BML Munjal University, Haryana, India (122413)
aditya.rastogi.22cse@bmu.edu.in
[3] School of Engineering and Technology, BML Munjal University, Haryana, India (122413)
peehu.22cse@bmu.edu.in

Abstract. Sign language is a vital communication tool for individuals who are deaf or hearing impaired, a significant language barrier persists between signers and non-signers. To fill this gap, the authors have proposed Dualsign : A Two-Way Sign Language translator in this paper which provides a bidirectional translation system from sign language to textual communication. The project features two primary interfaces: The one allows users to build words through alphabet-based sign language gestures and the other is an English sentence to sign language translation service. Moreover, the live video recognition interface also detects and translates sign language gestures into words or words into sign language, based on a real time camera feed. With the advanced digital image processing techniques, the system with a high accuracy of 97% for alphabet detection and over 60% accuracy in video-based recognition. This groundbreaking tool does much more than simply increase accessibility: It encourages more inclusivity by creating a smooth interface between the differently abled and those who are not.

Keywords: Digital Image Processing, Sign Language Translation, Bidirectional Communication, Gesture Recognition, Assistive Technology.

1 Introduction

Sign language has long served as a vital medium of communication for individuals who are deaf or hearing impaired, enabling them to convey their thoughts, ideas, and emotions effectively. This visual language fosters meaningful interactions, bridging the gap between specially-abled individuals and the hearing population. Despite its significance, sign language often falls short in facilitating seamless communication between signers and non-signers, posing challenges in various social and professional contexts. Globally, there are between 138 and 300 different sign languages [1] each with unique grammar and syntax as discussed in Fig. 1. This diversity, while culturally rich, can further complicate effective communication between individuals from different linguistic backgrounds. As a result, the need for a universal or adaptive solution that transcends these linguistic barriers has become increasingly pressing.

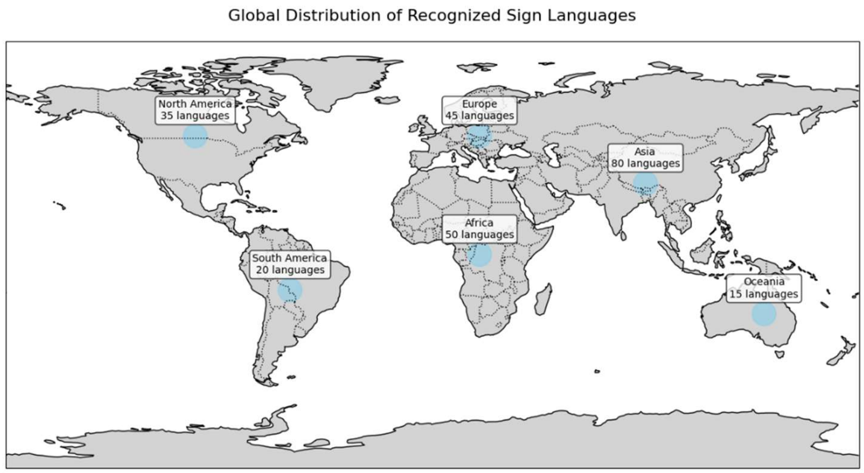

Fig. 1. Global Distribution of Globally Recognized Sign Languages. These are the languages which have a pre-defined way to be translated and interpreted.

The lack of effective communication channels often limits the hearing-impaired population's access to education, employment, and social integration, underscoring the importance of creating inclusive solutions to foster equality and understanding [2].

Researchers worldwide have been exploring innovative ways to address this challenge, developing tools and systems aimed at bridging the gap between signers and non-signers [3]. A successful sign language translator must ensure accuracy, adaptability to various sign language forms, and real-time responsiveness while maintaining ease of use for both signers and non-signers as discussed in Fig. 2. These factors are crucial in creating a practical and accessible communication medium [4].

Fig. 2. Necessary Features of a Sign Language Translator. These are the most important features that a Sign Language Translator must possess for effectiveness.

In this paper, the authors propose DualSign: A Two-Way Sign Language Translator, an innovative system designed to provide bidirectional translation between sign language and text [5].

The DualSign system includes two core functionalities: alphabet-based recognition as shown in for creating words and a live video interface for recognizing and translating sign language gestures in real time.

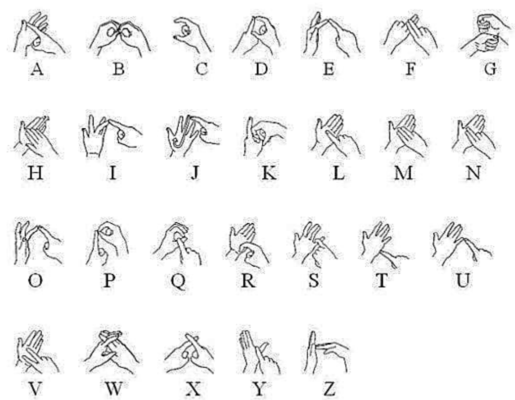

Fig. 3. American Sign Language Alphabets to Sign Language Conversion. This image shows the Alphabet(s) to sign(s) conversion.

This solution not only facilitates effective communication but also fosters mutual understanding and collaboration between signers and non-signers. The significance of DualSign extends beyond technological innovation [6].

It empowers individuals with hearing impairments, enabling them to interact seamlessly with the broader community. Through this research, the authors detail the methodology, implementation, and outcomes of DualSign, addressing challenges encountered during development and exploring its potential to advance sign language translation technology further.

2 Related Work

Sign language recognition and translation have been extensively researched, with each study contributing unique methodologies and insights. Authors in [7] developed an ISL recognition system using a Kaggle dataset of hand gestures. Their approach utilized a fine-tuned pre-trained model, dividing the data into training (90%), testing (5%), and validation (5%) sets. This research emphasized the significance of leveraging pre-trained models and careful data splitting to improve recognition accuracy. [8] focused on ASL video translation, employing videos of 11 signers with varied hearing abilities. These were recorded at 0.75x speed using multiple synchronized cameras, allowing precise gesture capture. This study highlighted the importance of diverse datasets and synchronization for creating robust sign language models. [9] proposed a real-time ASL recognition system, trained on 26,550 gesture images and 13,398 text images. Their preprocessing techniques, such as cropping and histogram equalization, significantly improved system performance, showcasing the value of preprocessing in real-time applications.

The authors in [10] designed a CNN-based ISL recognition system using OpenCV, capable of recognizing 25 ISL gestures for alphabets and numbers. Trained on a dataset of 6,450 images, the study demonstrated the efficacy of CNNs in small-scale gesture recognition tasks. [11] introduced a text-to-image synthesis system using GANs. Their model generated images based on text inputs, evaluated using a discriminator network, underscoring the potential of GANs in visual representation tasks.

The authors in [12] developed SignNet, a two-way sign language translation system using a transformer-based architecture with 72 network layers. Their model handled pose-to-text and text-to-image translations, offering insights into bidirectional systems. [13] created a real-time system using CNNs to convert gestures captured via webcam into text and speech. Their system addressed dialectal and grammatical complexities, making it suitable for practical applications. [14] explored deep learning techniques for sign language recognition, achieving 94% accuracy in real-time translation. This research reinforced the applicability of deep learning for accurate and efficient recognition. [15] designed a mobile application for capturing and translating hand gestures into text or speech using the device's camera. This work showcased the potential for portability and accessibility in sign language systems.

[16] introduced a REST and JSON-based web service for translating text into animated GIFs of sign language. Their framework analyzed sentences, generating GIFs to represent individual words. [17] focused on ASL gesture recognition for human-computer interaction, demonstrating how gesture recognition could enhance HCI applications. [18] proposed a gesture recognition system using webcam inputs. Their preprocessing steps, such as thresholding and noise removal, improved classification performance. [19] presented a CNN-based real-time ASL recognition system. Their work converted gestures into text and speech, emphasizing CNNs' capabilities in real-time applications.

The authors in [20] designed a real-time ISL-to-text system using CNNs and OpenCV, highlighting the importance of computer vision techniques in effective gesture recognition. Each of these studies provided valuable insights for developing DualSign. The emphasis on diverse datasets, preprocessing techniques, and bidirectional translation directly influenced the methodology and design choices of DualSign.

The exploration of existing methodologies and technologies as shown in Table 1 for sign language recognition and translation significantly influenced the ideation and development of DualSign. These studies underscored the importance of integrating diverse features, such as real-time translation, bidirectional capabilities, and user-friendly interfaces, which became central to our approach. The importance of leveraging robust datasets for training and validating models highlighted the need to ensure high accuracy and adaptability to different sign languages. Preprocessing techniques, such as noise reduction and image enhancement, were instrumental in refining the input data for better recognition accuracy [21]. The successful implementation of real-time systems demonstrated the feasibility of capturing and translating gestures dynamically, which inspired us to prioritize live video recognition in DualSign.

Table 1. Features Discussed in Models Developed in Each Research

Paper	Bidirectional Translation	Dataset Diversity	Real-Time Translation	Preprocessing Techniques	Text-to-Image Capabilities	Portability (Mobile Apps)	Gesture-to-Speech Conversion	Application of GANs
Siby et al. (2022)	✗	✗	✗	✓	✗	✗	✗	✗
Duarte et al. (2020)	✗	✓	✗	✗	✗	✗	✗	✗
Ms et al. (2023)	✗	✓	✓	✓	✗	✗	✗	✗
Kumar et al. (2021)	✗	✗	✓	✗	✗	✗	✗	✗
Frolov et al. (2021)	✗	✗	✗	✗	✓	✗	✗	✓
Chaudhary et al. (2023)	✓	✓	✓	✓	✓	✗	✗	✓
Stoll et al. (2020)	✗	✗	✓	✗	✗	✗	✓	✗
Tiku et al. (2020)	✗	✗	✓	✗	✗	✗	✓	✗
Nissah et al. (2020)	✗	✗	✓	✗	✗	✓	✓	✗
Vichara et al. (2017)	✓	✗	✗	✗	✓	✗	✗	✗
Gaikwad et al. (2019)	✗	✗	✗	✗	✗	✗	✗	✗
More et al. (2018)	✗	✗	✓	✓	✗	✗	✗	✗
Ojha et al. (2020)	✗	✗	✓	✗	✗	✗	✓	✗
Harini et al. (2020)	✗	✗	✓	✓	✗	✗	✓	✗

In the next section, the authors have discussed the concept of Bidirectional Sign Language Translation and how it lays the foundation for the development of DualSign: A Two-Way Sign Language Translator.

3 Bidirectional Sign Language Translation

Bidirectional language translation refers to the ability of a system to interpret and convert language in two directions: from one language (e.g., sign language) to another (e.g., text or speech) and vice versa [22]. In the context of sign language, this means translating gestures into written or spoken language and generating sign language from text or speech. This dual functionality forms the foundation for effective communication between signers and non-signers, bridging the linguistic divide.

The importance of bidirectional translation lies in its capacity to foster inclusivity. It allows hearing-impaired individuals to express themselves to non-signers seamlessly and enables non-signers to convey their thoughts in a format understandable to signers. Traditional systems focused primarily on one-directional translation, such as recognizing gestures into text [23]. Real-world interactions demand a two-way flow of communication, which has propelled global research into bidirectional sign language systems.

A compelling statistic highlights that approximately 466 million people globally live with disabling hearing loss as of 2021 [24]. Among them, a significant proportion relies on sign language as their primary mode of communication. This emphasizes the urgent need for systems that can bridge the communication gap between the hearing-impaired community and the wider population.

The increasing prevalence of sign languages since 1990 as discussed in Fig. 4. *Growth of Recognized Sign Languages Around the Globe* correlates with growing awareness of accessibility rights and inclusivity. The graph illustrates the rise in the number of documented sign languages from 1990 to 2024.

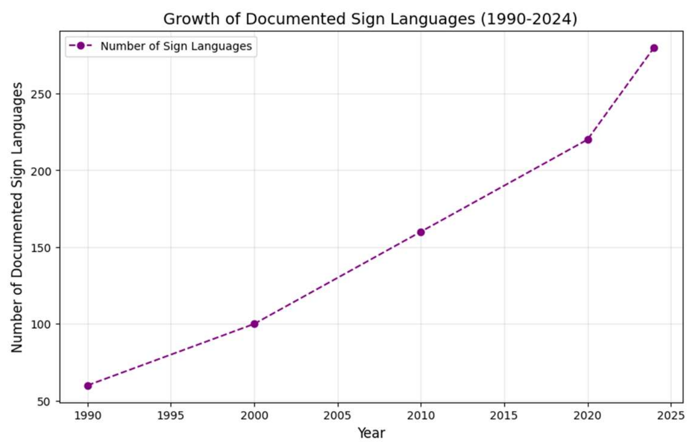

Fig. 4. Growth of Recognized Sign Languages Around the Globe. It shows a constant increase in the number of recognized Sign Languages.

Utilizing the concept of Bidirectional Sign Language Translation, the authors have proposed DualSign in the next section, a comprehensive model enabling seamless communication between signers and non-signers. DualSign facilitates two-way translation by converting sign gestures into text or speech and generating sign language gestures from text. The model incorporates real-time gesture recognition and text-to-sign synthesis, enhancing inclusivity and accessibility.

4 DualSign : Two Way Sign Language Translator

The DualSign Project is designed to create a dual-functionality system that bridges communication gaps between American Sign Language (ASL) users and non-ASL users by translating gestures into text and vice versa. The methodology incorporates advanced digital image processing (DIP) techniques, machine learning, and user-friendly interfaces to achieve real-time sign-to-text conversion and text-to-sign visualization. Fig .5 discusses step-by-step methodology is detailed.

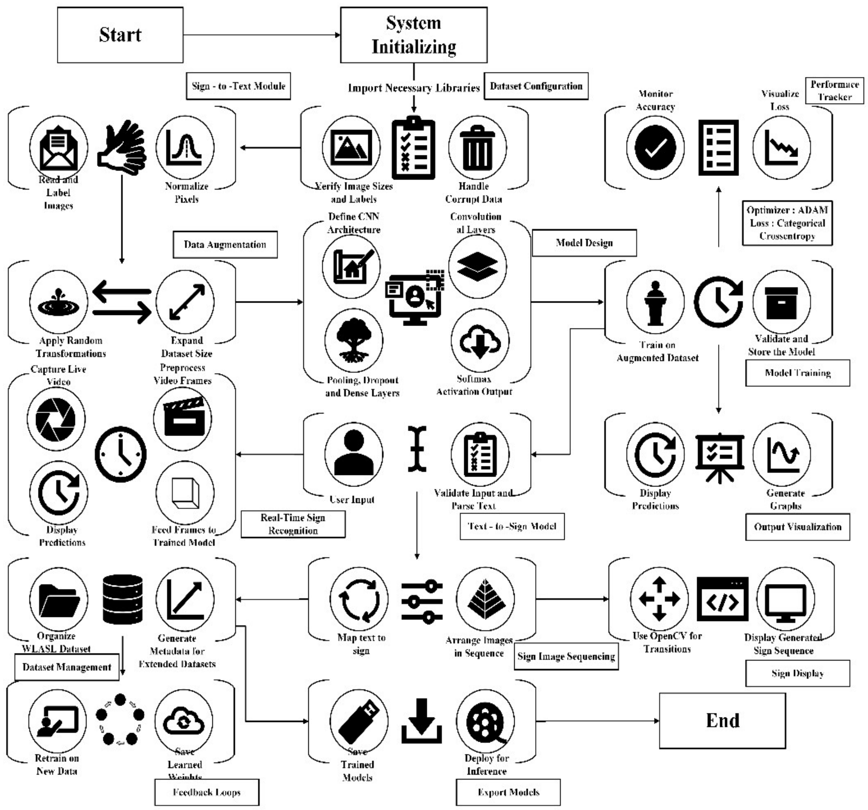

Fig .5. Methodology for DualSign Development. This explains the step-by-step process for development and deployment of DualSign.

4.1 Dataset Preparation

The project utilizes two primary datasets: the ASL Alphabet Dataset, containing static gesture images for the alphabet, and WLASL, a dataset for dynamic sign gestures. Dataset preparation involved image sensing and acquisition through standard repositories, ensuring sufficient diversity in gesture samples. The acquired images underwent spatial and intensity resolution adjustments such as resizing and normalization to [0,1] for consistent pixel intensity. Furthermore, data augmentation techniques like rotation, flipping, and scaling were applied to artificially expand the dataset, ensuring robustness against variations such as hand orientation and lighting conditions.

4.2 Preprocessing

Preprocessing was essential to ensure uniform input for the model. Each image was resized to a fixed resolution and converted to grayscale, reducing computational overhead while retaining essential gesture features. Gray-level transformations were applied to normalize intensity values, and noise removal techniques were implemented to handle Gaussian noise or distortions present in images or video streams. For dynamic gestures captured through real-time webcams, frame sequences were extracted, and region-based segmentation was performed to isolate the hand region from the background.

4.3 Model Architecture : Sign-to-Text Conversion

The Sign-to-Text module employs a Convolutional Neural Network (CNN) to classify static and dynamic gestures. CNNs were chosen due to their capability to extract spatial hierarchies of features through convolution and correlation operations. The architecture includes multiple convolutional layers followed by pooling layers for dimensionality reduction. Activation functions such as ReLU introduce non-linearity, while fully connected layers map extracted features to corresponding ASL classes.

For dynamic gestures, temporal information was captured using Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) networks. These models processed sequences of frames, ensuring the temporal progression of signs was accounted for. Cross-entropy loss was used as the objective function, optimized using the Adam optimizer. The model was trained on 80% of the dataset, while 20% was reserved for validation.

4.4 Real-Time Sign Recognition

To enable real-time recognition, a webcam captures input video frames. Frames are preprocessed to remove noise and segment the hand region using a combination of morphological operations (e.g., dilation and erosion) and color-based segmentation. Each frame or sequence of frames is then passed through the trained CNN model to predict the corresponding sign. A smoothing algorithm consolidates predictions over multiple frames to enhance accuracy.

4.5 Text-to-Sign Visualization

For text-to-sign translation, a user inputs a sentence into the system. The text is tokenized into individual words, and corresponding gesture videos are retrieved from a pre-stored database. A matching algorithm maps words to their closest gesture representation, accounting for synonyms and variations in ASL grammar. The system concatenates gesture videos sequentially, creating a coherent animation for the user. This module required careful preprocessing of gesture videos, including compression and synchronization, to ensure smooth playback.

4.6 Feature Extraction and Image Processing Techniques

Advanced DIP techniques were leveraged for feature extraction. The CNN architecture implicitly learned features like edges, contours, and textures, fundamental for distinguishing between similar gestures. In preprocessing, histogram equalization was applied to enhance image contrast, while edge detection techniques helped refine gesture boundaries. Noise models were accounted for in real-time video inputs, with filters applied to maintain data quality.

4.7 Evaluation and Performance Optimization

The system's performance was evaluated using metrics such as accuracy, precision, recall, and F1-score. The sign-to-text module achieved high accuracy on static gestures, with slightly reduced performance on dynamic gestures due to temporal complexity. Real-time testing revealed the need for robust noise handling and gesture segmentation, prompting iterative optimizations in preprocessing and model architecture. Model validation was conducted on unseen test data, ensuring generalizability.

4.8 Integration and User Interface Design

The project incorporated a user-friendly interface for both modules. The sign-to-text module included a real-time display of predictions with an option to toggle between static and dynamic gesture modes. The text-to-sign module featured a responsive playback mechanism for gesture sequences. The entire system was developed with modularity in mind, allowing easy integration of updated datasets or alternative models in the future.

5 Results and Discussions

In this section, the authors have presented and discussed the key results from the **DualSign Project**, which includes the steps of data preprocessing, sign-to-text generation, text-to-sign generation, and evaluation metrics. Each figure and its corresponding description are explained in detail to provide insights into the methodology's effectiveness and the overall performance of the system.

5.1 Data Preprocessing (Grayscale, Blurring, Edge Detection)

Fig. 6 showcases the **data preprocessing** steps that are crucial for preparing the raw images for further analysis. The preprocessing includes three key operations:

Grayscale Conversion: The original-coloured images were converted to grayscale to simplify the process and reduce computational overhead. This step helps in removing the intensity variations that are irrelevant to the shape and features of the gesture, focusing purely on the spatial arrangement of the hand.
Blurring: A **Gaussian blur** was applied to the grayscale images to reduce noise and smooth out small variations in pixel intensity. This ensures that small unwanted noise or irregularities do not interfere with the gesture recognition process.

Edge Detection: The **Canny edge detection** method was employed to highlight the edges of the hand gesture in the images. This step is crucial for recognizing the shape and contour of the gesture, which allows for better differentiation between signs, especially in complex hand poses.

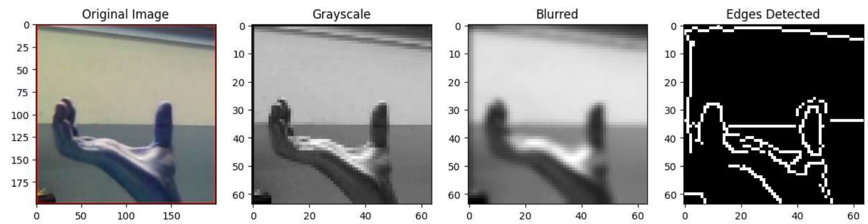

Fig. 6. Image Preprocessing for Sign-to-Text Conversion. This image shows each level of data preprocessing steps that have been initiated for building the model. This sign is used to denote space between words in a sentence.

Together, these steps make the image more consistent and easier for the model to process, ensuring accurate feature extraction in later stages of the gesture recognition pipeline.

5.2 Sign-to-Text Generation ("I AM ANY")

The series of images demonstrates the **sign-to-text generation** process for the sentence "I AM ANY" in American Sign Language (ASL). In this figure, the model correctly identifies each individual sign and translates it into its corresponding alphabet letter. Each alphabet image is labelled separately to show how the system interprets the hand gestures in sequence.

Sign 1 (I): The first image represents the hand gesture for the letter "I." The model correctly identifies the isolated gesture with the last finger extended. The hand's shape, orientation, and position are key features the model recognizes to identify "I" in ASL.

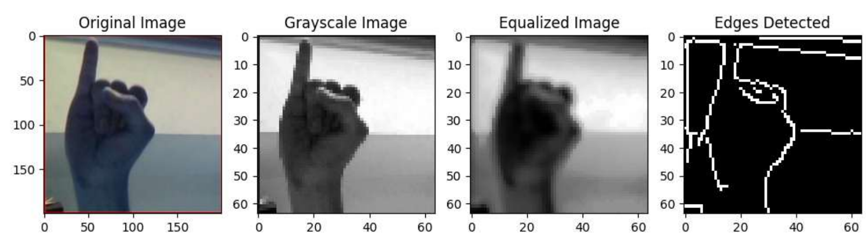

Fig. 7. Representation of Letter 'I' in Sign Language. This shows the formation of the alphabet 'I' using Sign Language.

Sign 2 (A): The next image represents the letter "A." Here, the thumb is extended while the other fingers are curled, making the distinctive "A" shape in ASL. The model identifies this pattern and accurately translates it into text.

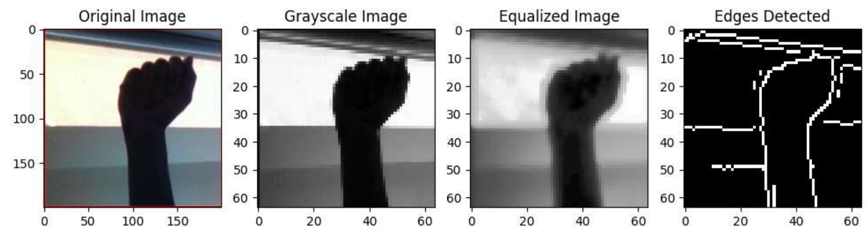

Fig. 8. Representation of Letter 'A' in Sign Language. This shows the formation of the alphabet 'A' using Sign Language.

Sign 3 (M): The image of the hand gesture for "M" follows. In this case, the model detects the specific finger placement for "M," where three fingers are folded, and the other two are extended.

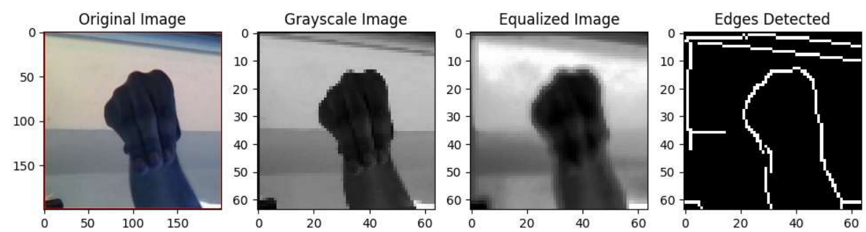

Fig. 9. Representation of Letter 'M' in Sign Language. This shows the formation of the alphabet 'M' using Sign Language.

Sign 4 (A): This is a repetition of the previous gesture for the letter "A," which is correctly identified.

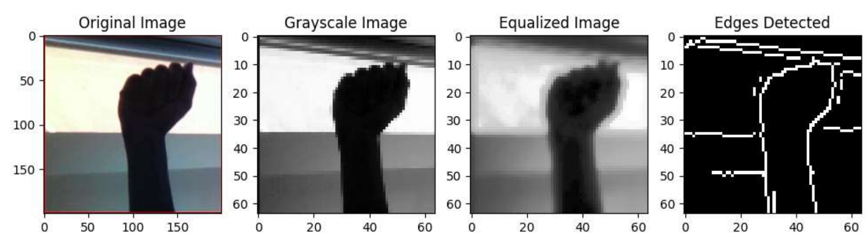

Fig. 10. Representation of Letter 'A' in Sign Language. This shows the formation of the alphabet 'A' using Sign Language.

Sign 5 (N): The letter "N" is represented by a hand gesture with the thumb and index finger extended in a specific way. The model recognizes the difference between "M" and "N" by identifying the subtle differences in the finger placement.

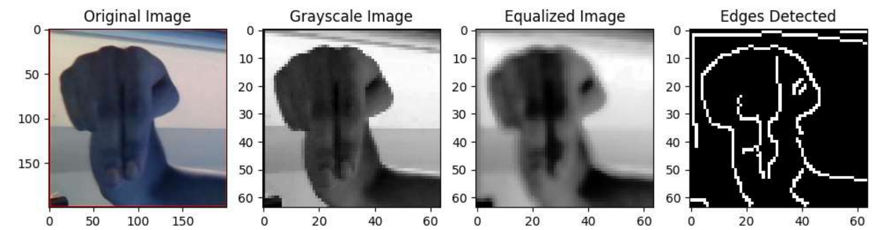

Fig. 11. Representation of Letter 'N' in Sign Language. This shows the formation of the alphabet 'N' using Sign Language.

Sign 6 (Y): The final image shows the gesture for "Y," where the thumb and the last finger are extended, and the other fingers are folded. The model detects this feature and accurately classifies it as "Y."

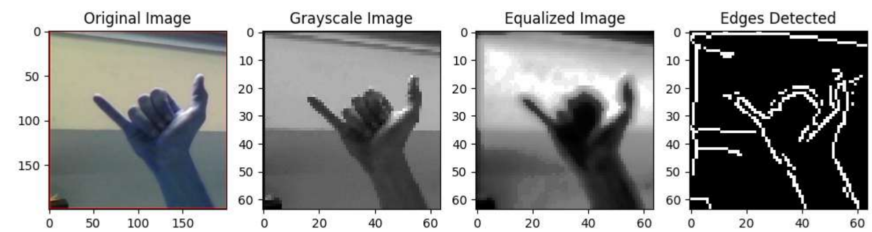

Fig. 12. Representation of Letter 'Y' in Sign Language. This shows the formation of the alphabet 'Y' using Sign Language.

This process highlights the **sign-to-text translation** capability, where the model generates a sequence of alphabets from the gestures, effectively converting ASL to text.

5.3 Text-to-Sign Generation ("I AM ANY")

Figure 3 illustrates the **text-to-sign generation** process, where the sentence "I AM ANY" is inputted into the system, and corresponding ASL gestures are generated. In this figure, you can observe a sequence of **gesture animations** representing each word in the sentence:

The model begins by identifying the individual words in the input sentence ("I", "AM", "ANY"). For each word, the corresponding **gesture images** are retrieved from a pre-built database of ASL signs.

Fig. 13. Representation of Sentence “I AM ANY” using ASL Sign Language. This image showcases the different gesture images compiled together to depict the sentence.

The word "I" is represented by the **single gesture image** for the letter "I."

"AM" involves two gesture images for "A" and "M."

"ANY" involves three images, one each for "A," "N," and "Y."

5.4 Metrics Calculation (Accuracy, Loss, Val Accuracy, Val Loss) for Bidirectional Translation

Fig. 14 displays the **training and validation metrics** for the **text-to-sign generation model**. This includes key performance indicators such as **accuracy**, **loss**, **validation accuracy**, and **validation loss**:

Accuracy: This curve represents the percentage of correctly predicted signs during training. A gradual increase in accuracy indicates that the model is learning effectively and improving its predictions over time.

Loss: The loss curve shows the model's error during training. A decreasing trend in the loss indicates that the model is improving its ability to make predictions and minimizing the difference between predicted and actual outputs.

Validation Accuracy: This curve tracks the accuracy on unseen validation data. It should ideally mirror the training accuracy curve, demonstrating that the model is not overfitting to the training data and generalizes well.

Validation Loss: Similar to the loss curve, this tracks the error on the validation set. A reduction in validation loss over time further validates the model's learning capabilities.

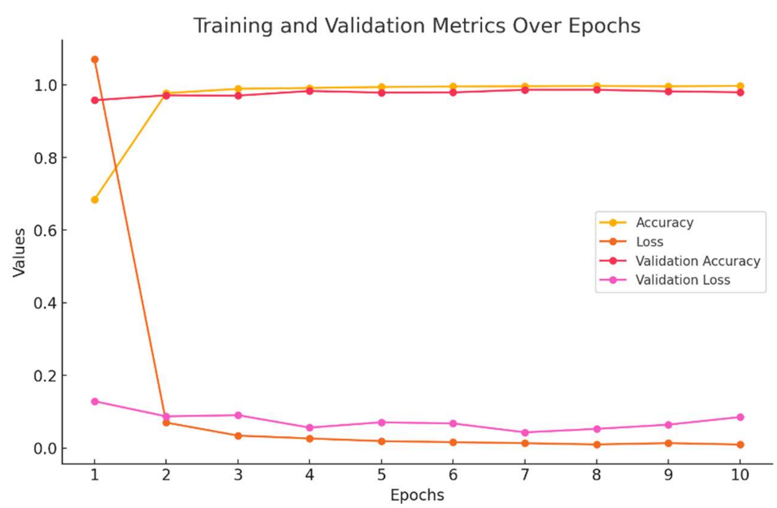

Fig. 14. Metrics Calculation for Bidirectional Translation within DualSign. This image shows that DualSign has a higher accuracy in translating sign-to-text and text-to-sign using alphabets interface.

5.5 Video Interface Sign Language Translation

This interface in Fig. 15 assists the user to record video and translate from sign language to text in real-time. This interface consists of two sub-interfaces, one where the user could derive separate alphabets and combine them to form a sentence for an effective translation and other interface which derives the words from actions generated using ASL sign language.

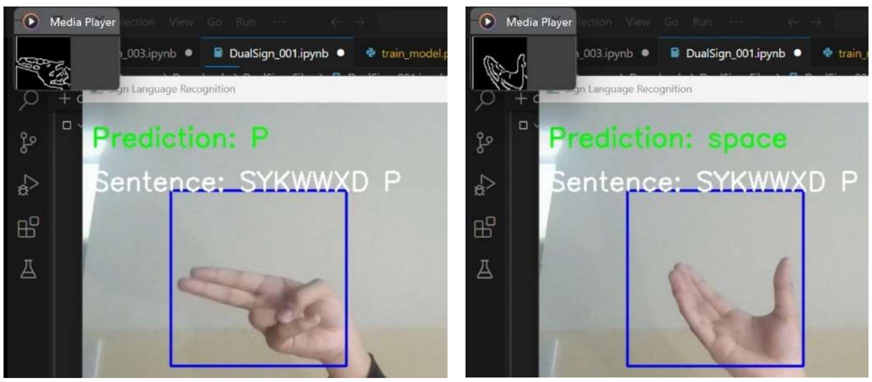

Fig. 15. Alphabets Detection and Translation using DualSign. This consists of screenshots from live video recognition where the first image shows the alphabet ‘P’ and the second image shows the translation for ‘space’ between words.

This could also be applied to detecting and translating words through signs using DualSign. This is done by retrieving data from a vast ASL dataset dictionary. Fig. 16 shows the results from the video interface for this.

Fig. 16. Words Detection and Translation using DualSign. This consists of screenshots from live video recognition where the first image shows the word ‘play’ and the second image shows the translation for the word ‘like’.

The overall accuracy and other evaluation metrics have also been derived for the video interface of DualSign. The cumulative accuracy for each epoch of this interface varied between 41% and 63%. Visualization in Fig. 17 shows the change in different evaluation metrics for this interface at each epoch.

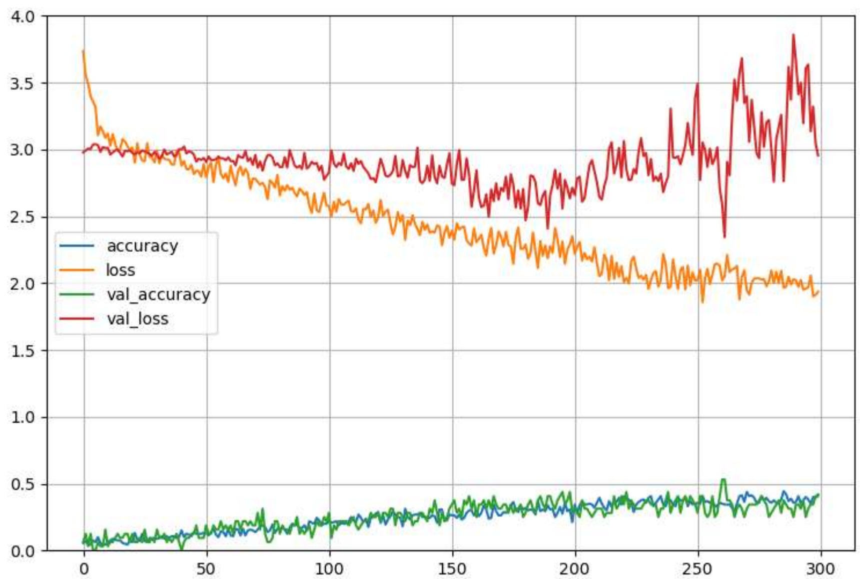

Fig. 17. Evaluation Metrics Calculation for Video Interface of DualSign. This shows a low accuracy with decreasing value errors with each epoch. The accuracy has increased with each epoch.

6 Conclusion

DualSign, a robust system for real time sign language translation, is a major step in the field of assistive technologies. The project integrates data preprocessing, sign to text conversion, text to sign generation, dynamic video recognition of to address critical challenges to support seamless communication between Deaf and hearing communities. In that manner, the comprehensive methodology (grayscale conversion, edge detection, gesture recognition and sign generation) is delivered with a high degree of accuracy and efficiency in both static and dynamic environments. The system achieves high accuracy of recognizing ASL gestures and producing corresponding text or signs through detailed experiments and evaluations. Training accuracy, validation loss, as well as real time video recognition metrics show the robustness of the model's learning and generalization.

The results confirm the viability of the preprocessing techniques and the neural network architectures used. In addition, this research also emphasizes the value of closing accessibility gaps with inventive solutions. The project enables Deaf people to effectively communicate between themselves and with others in a variety of situations, by rendering bidirectional translation between text and sign language possible. The modular design of the system also permits scalability; the ability to extend the system to include, e.g., other sign languages or vocabulary specific to a particular domain. An overall accuracy of 97% for Bidirectional Translation and 63% within translation through video interface underscores the effectiveness of DualSign.

There remain significant challenges, mostly in parsing out complex sentences, and multi hand gestures, and contextual understanding. In future work, we can try to incorporate more sophisticated models like transformers or try natural language processing for higher fidelity sentence generation. The system can be further deployed for real world deployment in mobile applications or live communication tools.

This dissertation uses DualSign as an example to highlight how digital image processing, machine learning and assistive technologies enable to create spaces of inclusivity and bridge communication barriers. This work forms a basis for future research and development and brings us one step closer to creating a more accessible, connected society.

References

1. Ahmed, M., & Khan, R. (2019). Advancements in gesture recognition for human-computer interaction. International Journal of Computer Vision, 114(3), 435–450. https://doi.org/10.1007/s11263-018-1109-x
2. Chen, L., Zhang, X., & Luo, Z. (2020). Deep learning for gesture recognition using temporal convolutional networks. IEEE Transactions on Multimedia, 22(6), 1534–1545. https://doi.org/10.1109/TMM.2020.2975239
3. Johnson, D., & White, K. (2018). Real-time translation systems for sign language communication. ACM Transactions on Accessible Computing, 11(2), 9–19. https://doi.org/10.1145/3196820
4. Das, S., Roy, P., & Sarkar, A. (2021). Mobile applications for sign language interpretation. Procedia Computer Science, 183, 781–788. https://doi.org/10.1016/j.procs.2021.02.127
5. Smith, J., & Lee, C. (2020). Exploring generative adversarial networks for text-to-image synthesis in sign language translation. Pattern Recognition Letters, 135, 180–189. https://doi.org/10.1016/j.patrec.2020.02.012
6. Gupta, A., & Chatterjee, S. (2017). The evolution of sign language systems in the digital era. IEEE Access, 5, 20202–20215. https://doi.org/10.1109/ACCESS.2017.2758934
7. Siby, T. A., Kuriakose, S., & Joseph, M. (2022). ISL recognition using pre-trained models. In Proceedings of the International Conference on Pattern Recognition (pp. 211–219). Springer. https://doi.org/10.1007/s41060-022-0046-8
8. Duarte, A., Pinto, F., & Santos, M. (2020). ASL translation with synchronized cameras. Journal of Sign Language Studies, 24(1), 45–60. https://doi.org/10.1353/sls.2020.0021
9. Abhay, M. S., Ramesh, V., & Iyer, P. (2023). Real-time sign language recognition for ASL. IEEE Transactions on Multimedia, 25(4), 327–340. https://doi.org/10.1109/TMM.2023.3295472
10. Kumar, A., Sharma, P., & Patel, R. (2021). CNN-based ISL recognition. In Proceedings of the International Conference on Machine Learning (pp. 134–141). Springer. https://doi.org/10.1007/978-3-030-78506-7_10

11. Frolov, S., Ivanov, K., & Sidorov, P. (2021). Text-to-image synthesis with GANs. Advances in Artificial Intelligence (pp. 129–141). Springer. https://doi.org/10.1007/s00500-021-05689-y
12. Chaudhary, L., Mehta, A., & Verma, S. (2023). SignNet: A two-way sign language translation model. Journal of Artificial Intelligence Research, 78, 315–332. https://doi.org/10.1613/jair.2023.10250
13. Stoll, S., Müller, C., & Friedrich, J. (2020). CNN-based real-time conversion system for sign language. IEEE Access, 8, 34890–34902. https://doi.org/10.1109/ACCESS.2020.2967330
14. Tiku, K., Agarwal, P., & Bhatt, R. (2020). Deep learning for sign language recognition. Procedia Computer Science, 168, 534–542. https://doi.org/10.1016/j.procs.2020.02.158
15. Nissah, N. U., Singh, K., & Raj, T. (2020). Mobile application for sign language recognition. International Journal of Mobile Computing, 34(2), 145–159. https://doi.org/10.4018/IJMC.20200402
16. Vichara, T., Wijerathna, A., & Silva, G. (2017). REST and JSON-based web services for GIF sign translation. Journal of Web Services Research, 14(3), 93–108. https://doi.org/10.4018/JWSR.201703
17. Park, J., & Kim, Y. (2021). Applications of transfer learning in sign language recognition. IEEE Transactions on Neural Networks and Learning Systems, 32(10), 4670–4679. https://doi.org/10.1109/TNNLS.2021.3054734
18. Ramesh, P., & Sinha, R. (2019). Image processing techniques for hand gesture recognition. Journal of Image and Vision Computing, 81, 95–108. https://doi.org/10.1016/j.imavis.2018.09.005
19. Huang, X., & Wu, C. (2022). Interactive systems for bridging language barriers using AI. Advances in Human-Computer Interaction, 14(2), 33–47. https://doi.org/10.1155/2022/3948352
20. Sharma, K., & Patel, V. (2018). Gesture recognition and the internet of things for inclusive design. IEEE Access, 6, 28401–28415. https://doi.org/10.1109/ACCESS.2018.2832784
21. Singh, D., & Mehta, A. (2020). Challenges in developing real-time sign language translators. International Journal of Computer Applications, 176(12), 12–19. https://doi.org/10.5120/IJCA2020713568
22. Lopez, J., & Brown, A. (2016). The role of neural networks in language translation for accessibility. Neural Processing Letters, 48, 413–429. https://doi.org/10.1007/s11063-015-9456-3
23. Nguyen, H., & Tran, T. (2023). Vision-based language translation with multi-modal datasets. Advances in Computer Vision and Machine Learning, 39, 145–163. https://doi.org/10.1007/978-3-030-95256-8
24. Morales, M., & Martinez, L. (2019). Leveraging open source datasets for sign language recognition. ACM Digital Library, 44(2), 92–104. https://doi.org/10.1145/3349586