Realtime Sign Language Recognition Using Computer Vision and AI

Gabriel Serrano
Department of Computer Science and Technology
Kean University
Union, NJ USA
sergabri@kean.edu

Daehan Kwak
Department of Computer Science and Technology
Kean University
Union, NJ USA
dkwak@kean.edu

***Abstract*—Promoting inclusive communication is essential and sign language plays a crucial role in achieving this goal. In this research, a system is developed capable of making it easier for those who primarily communicate through sign language to interact with those who are unable or may not be knowledgeable in the language. This has the potential to help bridge the communication gap between those who are fluent in sign language, and those who may be struggling to learn or are not knowledgeable. In its current state, our system is capable of recognizing two forms of sign language, namely: American Sign Language and British Sign Language. The system also is capable of performing facial expression analysis to account for non-verbal inflections expressed by the user. These tasks are accomplished by making use of computer vision provided by the OpenCV Python library. It also uses various machine learning models and the MediaPipe library. We explore two approaches for sign language recognition: contour-based recognition and landmark-based recognition. Additionally, facial landmarks for facial expression analysis are investigated which can be used to detect expressions and inflections from a user's face alone. The next steps of this research will consist of working with more complex words and phrases and investigating gesture recognition.**

Keywords—Sign Language, Facial Expression, Image Recognition, Computer Vision, Artificial Intelligence (AI), Machine Learning (ML).

I. INTRODUCTION

Based on the Survey of Income and Program Participation (SIPP), an estimated one-in-twenty Americans are currently deaf or hard of hearing [1]-[3]. This translates to roughly ten-million people who are hard of hearing and around one-million people who are deaf in America alone. While the population of individuals who are fully deaf or hard of hearing is already quite large, the World Health Organization [4] estimates that by 2050, nearly 2.5 billion people worldwide will experience some degree of hearing loss. Many individuals in this population primarily utilize sign language as their primary form of communication. However, while learning sign language may come easy to some, others struggle to learn the language, and this can often result in a gap in communication. These gaps affect not only individuals who are deaf or hard of hearing, but also friends and loved ones. To address this, we set out to leverage novel technologies such as Computer Vision (CV) [5], [6], Artificial Intelligence (AI), and Machine Learning (ML) [7] to forward and improve on the existing research performed on sign language recognition. By doing so, our goal is to develop new and improved methods that would enable the creation of software capable of bridging the communication gap between those who primarily communicate through sign language, and those who cannot learn this important set of languages.

Through the developed software, the user can engage in sign language recognition through images, videos, and real-time video footage. Currently, the software supports the non-motion characters of both the American Sign-Language (ASL), and British Sign Language (BSL) alphabets. It also includes facial expression analysis and can detect three expressions including Happy, Sad, and Neutral. However, support for more relevant expressions such as an eyebrow raise to express questioning will be implemented in a later version.

The rest of this paper is organized as follows. Section 2 discusses related works, describing existing approaches to sign language recognition. Section 3 elaborates our comparative analysis of the two approaches we explored for sign language recognition and explains why we chose the selected approach. Section 4 covers the implementation and importance of facial expression analysis in our research. Section 5 illustrates the results, accuracy, potential impacts, and current limitations of the research. Lastly, Section 6 concludes the paper and outlines plans for potential future work.

II. RELATED WORK

A. Sign Language Recognition Approaches

Over the years, numerous studies by various researchers have tackled the complex task of sign language recognition. Interestingly, a wide variety of approaches have been applied. Zafrulla [8] investigated using an Xbox Kinect for gamified sign language recognition. This unique approach utilizes colored gloves, the Kinect video feed, and embedded accelerators to track hand movements of the user. Through this approach, they achieved a range of 51.5% and 76.12% ASL sentence verification rates for adult users.

In another study conducted by Camgoz [9], researchers experimented with a transformer model that can learn continuous sign language recognition in an end-to-end manner. In this approach, researchers eliminated the need for

2769-5654/23/$31.00 ©2023 IEEE
DOI 10.1109/CSCI62032.2023.00198

timing information for motion-based gesture recognition, while also outperforming a wide variety of existing models. Furthermore, through the sign language recognition, they evaluated the recognition performance on the challenging PHOENIX14T dataset which consists of parallel sign language videos, annotations, and translations.

While this approach has improved accuracy compared to other sign language recognition models, the primary downside to this approach is the significant number of resources the transformer model architecture requires. As such, this approach may become more suitable for everyday use in the near future when transformers are not as resource heavy as they currently are.

B. Facial Features in Sign Language Recognition

In various studies performed [10]-[14], researchers concluded that when working to implement forms of sign language recognition, it is important to be mindful of the significance of facial features as they play a vital role in the various forms of sign language. Facial expressions and features are used not only to indicate expression, but to indicate the grammar and inflections associated as well. Specifically, researchers discuss the importance of non-manual parameters which would include things like the user's head pose, facial expression, and lip patterns.

Overall, this study concluded that when combining the use of manual parameters such as hand signs, and non-manual parameters, there was an increase in recognition accuracy for both isolated and continuous sign language recognition scenarios. Based on these conclusions, we investigate facial expression analysis as part of our research, and plan to improve its function in later iterations.

III. COMAPRITIVE ANALYSIS OF TWO SIGN LANGUAGE RECOGNITION APPROACHES

The following section discusses the two separate approaches we investigated, as well as a comparative analysis between the two approaches with a detailed breakdown of the datasets used for each approach.

A. Contour-Based Sign-Language Recognition

The first of the two approaches we investigated is contour-based sign language recognition [15]-[17]. In this approach, the software uses the OpenCV [18] Python library to capture real-time camera footage from the user. A contour of the hand sign made by the user is generated by comparing two different "states" of the real-time camera feed within a designated frame. To do so, the software first starts off by capturing the background inside of the designated frame. This background serves as the initial "state" which will be utilized for generating a contour of the user's hand. Subsequently, the user creates a hand sign and positions their hand within the visualized bounds of the designated frame as shown in Fig. 1.

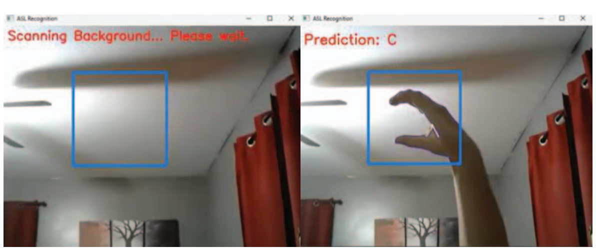

Fig. 1. Example of Contour-Based Sign Language Recognition Approach

Once this happens, a con tour is generated by identifying the pixels within the designated frame that deviated from their original values captured when scanning the background. Any differences found are depicted as white space, resulting in the contoured image as shown in Fig. 2.

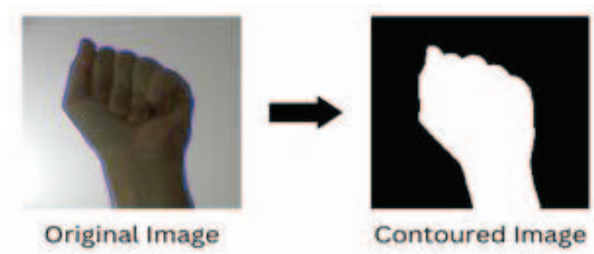

Fig. 2. Contoured Image Generation of the ASL Letter 'A'

This newly generated contoured image is then fed into a custom trained Convolutional Neural Network (CNN) [19] Model which returns a predicted value of the sign that has been made by the user. The model used has been custom trained on a dataset containing around 9,000 images at a resolution of 64x64 pixels. This dataset contains around 40 images per non-motion American Sign-Language (ASL) letter. The full dataset distribution is shown in Table 1.

TABLE I. CONTOUR-BASED APPROACH – ASL DATASET

Parameter	**Contour-Based Approach – ASL Dataset**	
	Values	***Comments***
Training Data	7,826 images	Resolution of 64x64 pixels, with around 40 images of each non-motions letter in the testing dataset, and approximately 325 images in the training dataset. (A-Z, excluding J and Z)
Testing Data	1,040 images	
Classes	24 classes	

Although this approach demonstrated high accuracy during testing, this approach was rather naïve and contained various pitfalls. Firstly, the predictions made using this approach can be easily influenced by changes in lighting conditions. In low-light environments, this approach struggles to recognize the user's hand, and will be unable to generate a properly contoured image. On top of that, minor environmental changes, such as someone passing by or a spinning fan blade, can negatively impact the accuracy of the predictions, as these extra movements introduce additional changes and generate obscure contours. Furthermore, there is also the risk of the contoured image of two signs being too close in similarities, such as the ASL letter 'A', and the ASL Letter 'S'. Since the contour of both signs strictly captures the outline, the resulting contour of each of these signs are not distinct enough from one another to ensure accurate predictions since the only difference between the two signs is the positioning of the thumb, which does not have much impact on the contour shape in this scenario. Finally, this approach is impractical for real-world applications as it cannot function without first scanning the user's background. This slows down the user for every hand sign that is made, and ultimately makes this approach unusable in the long run.

B. Landmark-Based Sign-Language Recognition

The second approach investigated was landmark-based sign language recognition. In this approach, the OpenCV and MediaPipe [20] Python libraries were used to detect the twenty-one hand landmarks from each of the user's hands. These landmarks include the wrist, tips, knuckles, and other joints of an individual's hand as shown in Fig. 3. Using these landmarks, the relative positions of each landmark to one another are determined, as well as the relative distance between each landmark.

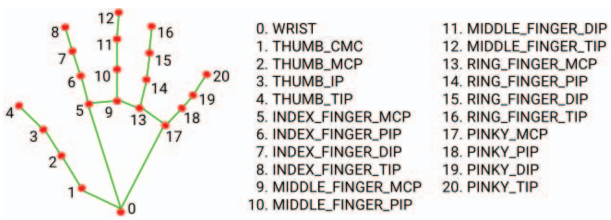

Fig. 3. MediaPipe Hand Landmarks Diagram from MediaPipe API [20]

Utilizing this approach, the system starts off by having the user select a sign language option, currently supporting American Sign Language (ASL) and British Sign Language (BSL). Next, the user's camera is turned on and begins collecting real-time video feed at the maximum supported frame rate. For most cameras, this maximum supported frame rate was 60 frames per second. Next, the list of hand landmarks for each hand in the frame is gathered using the MediaPipe library. With these landmarks, the system then determines whether each detected hand was the user's right or left hand by assessing the classification value of each detected hand. Following this, the list of hand landmarks was normalized to account for hands appearing closer or farther away from the screen. This normalization also accounted for the variance in hand sizes that may be detected. Once we generated the normalized list of hand landmarks, we then drew the hand landmarks onto the real-time video feed, and onto a smaller frame on a black background. The purpose of illustrating the hand landmarks is to provide the user with a clear view of what hand landmarks and signs are being detected. Lastly, the normalized hand landmark list is fed into a custom trained Recurrent Neural Network (RNN) [21]-[23] model, which predicts the value of the sign that has been made by the user. An example of this approach is shown in Fig. 4.

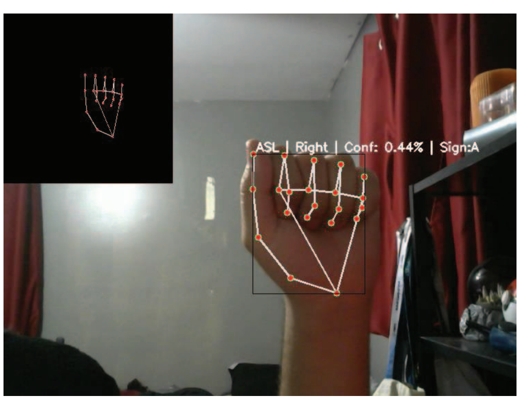

Fig. 4. Example of Landmark-Based Sign Language Recognition

There are currently two models present in our research for this approach, with each model representing a form of sign language, such as ASL. The datasets for each form of sign language vary based on the unique qualities of the languages. However, both datasets were created to include the non-motion letters of their respective languages. The dataset distribution for both models is shown in Tables 2 and 3.

TABLE II. LANDMARK-BASED APPROACH – ASL DATASET

Parameter	Landmark-Based Approach – ASL Dataset	
	Values	***Comments***
Training Data	960 rows of data (80%)	Each row of data consists of the normalized values of the 21 hand landmarks, and their relative distances between each other. There are approximately 50 rows of data per non-motion letter.
Testing Data	240 rows of data (20%)	
Classes	24 classes	

TABLE III. LANDMARK-BASED APPROACH – BSL DATASET

Parameter	Landmark-Based Approach – BSL Dataset	
	Values	***Comments***
Training Data	1,040 rows of data (80%)	Each row of data consists of the normalized values of the 21 hand landmarks, and their relative distances between each other. There are approximately 50 rows of data per non-motion letter.
Testing Data	260 rows of data (20%)	
Classes	26 classes	

As mentioned previously, the models used for this approach are custom trained RNN models. We selected this model type due to its suitability for easier implementation of gesture recognition, owing to the sequential nature of RNNs. Further research into gesture recognition will be conducted as part of our future work.

C. Contour-Based vs Landmark-Based Recognition

Between the two approaches explored in our research, it was concluded that the landmark-based approach is overall more efficient and accurate than its counterpart. The landmark-based approach effectively solves most of the issues associated with the contour-based approach. Firstly, the landmark-based approach is capable of detecting hands in most lighting conditions due to the highly sophisticated hand detection model the MediaPipe library offers. The landmark-based approach also does not require the user to scan their background before use since this approach is only focused on detecting the hands and extracting landmarks, rather than detecting a change in the user's background to generate a contour. The landmark-based approach also accounts for similar signs such as ASL 'A' and ASL 'S' since we are utilizing very specific landmarks and relative positions/distances of the landmarks rather than strictly using the outline of the user's hand. Finally, the landmark-based approach is much quicker and more efficient since we are primarily handling numeric data when making inferences to our machine learning models, in contrast to utilizing contoured images. As such, the landmark-based approach has been and will be used for the remainder of this research.

IV. Facial Expression Analysis

This section discusses the overall implementation and importance of the facial expression analysis portion of the research.

A. The Importance of Facial Expression Analysis

While people who communicate audibly are able to express linguistic information and emotions vocally, this is not the case for those who may be deaf or hard of hearing. In place of vocalized expressions, facial expressions allow individuals to express the same information without speech [24]. An example of this would be how you can indicate a question by raising an eyebrow in most forms of sign language.

B. Implementation of Facial Expression Analysis

To implement facial expression analysis, we made further use of the MediaPipe library similar to the landmark-based approach utilized for sign language recognition based on the user's hands. However, in contrast to the 21 hand landmarks used for sign language recognition based on the user's hands, there are 468 facial landmarks that are detected by the MediaPipe library. Due to the vast number of landmarks, the system is capable of picking up small details and changes in the user's expression. As such, we implemented facial expression analysis by collecting the normalized values of the relative distances between each of the 468 facial landmarks for each expression. These normalized values were then used to train an RNN model which returns the predicted value of the user's expression when inferenced. Once the prediction has been made, a face mesh is drawn onto the user's video feed and onto a smaller frame which isolates the detected facial landmarks. An example of facial expression analysis is shown in Fig. 5 where the user is demonstrating a neutral expression.

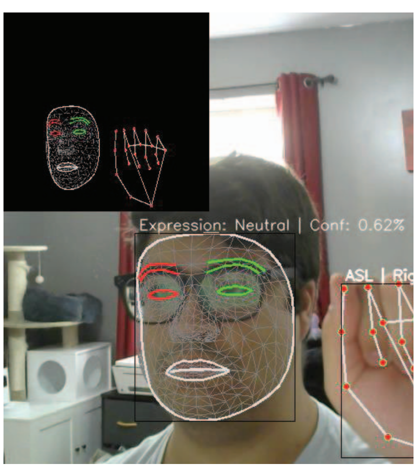

Fig. 5. Example of Facial Expresison Analysis for a Neutral Expression

Currently, the resulting software supports recognition of three expressions, Happy, Sad, and Neutral. The datasets collected for each expression consist of 50 rows of data per expression, with each row of data containing the list of normalized facial landmarks and their relative distances and positioning from each other. The full dataset breakdown is shown in Table 4. Other more relevant expressions, such as raising an eyebrow with a tilted head to indicate the presence of a question, are part of our future development efforts.

TABLE IV. Facial Expression Analysis Dataset

Parameter	**Facial Expression Analysis Dataset**	
	Values	***Comments***
Training Data	113 rows of data (75%)	Supports three expressions: Happy, Neutral, and Sad, with approximately 50 rows of data per expression
Testing Data	37 rows of data (25%)	
Classes	3 classes	

V. Results and Discussion

A. Classification Accuracy of Contour-Based Recognition

The classification accuracy of predicting the ASL alphabet while using the contour-based recognition approach

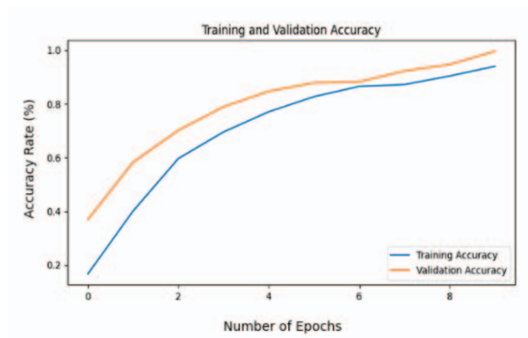

Fig. 6. Contour-Based Recognition (ASL Alphabet) Training and Validation Accuracy

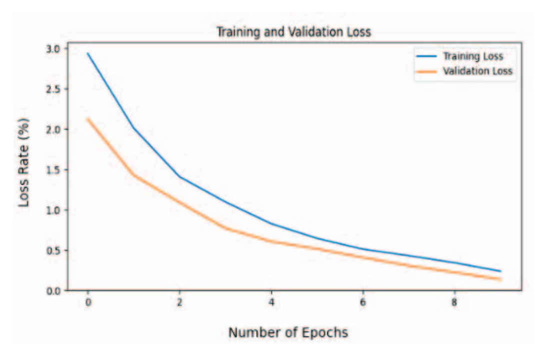

Fig. 7. Contour-Based Recognition (ASL Alphabet) Training and Validation Loss

was approximately 80%. While this level of accuracy is good, there were too many downsides associated with this approach that could negatively affect the contours used to make these predictions such as a changing background or poor lighting conditions. The results for this model's training are based on the first ten epochs. The training and validation accuracy of this approach is shown in Fig. 6 and Fig. 7.

B. Classification Accuracy of Landmark-Based Recognition

In its current state, the classification accuracy of the ASL alphabet dataset while using the landmark-based recognition approach is roughly 87%. This level of accuracy is sufficient for our purposes, especially considering this approach does not run the risk of irregular inputs, as seen in the contour-based approach. The training for this model employed an early-stopper and ran for roughly 406 epochs with a maximum of 1,000 epochs. The training and validation accuracy and loss are shown in Fig. 8 and Fig. 9.

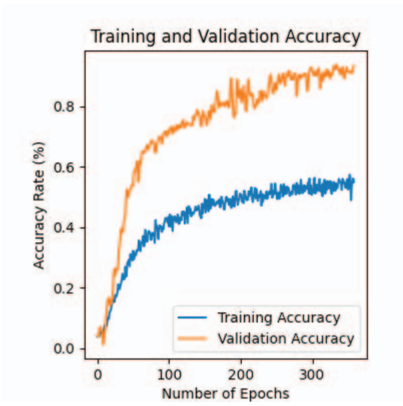

Fig. 8. Landmark-Based Recognition (ASL Alphabet) Training and Validation Accuracy

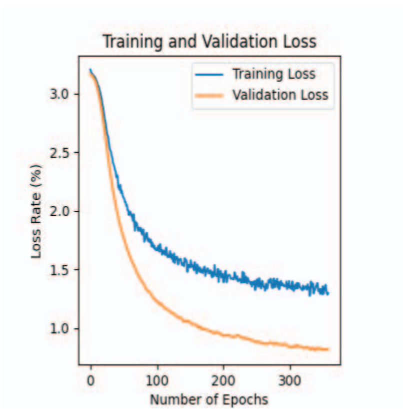

Fig. 9. Landmark-Based Recognition (ASL Alphabet) Training and

C. Classification Accuracy of Facial Expression Analysis

The facial expression analysis model can currently predict the user's expression with approximately 92% accuracy. However, the accuracy is most likely attributed to the limited number of exprssions it currently supports. When the model is inevitabley trained to support more expressions, we anticipate the classification accuracy to decrease by around 5-15%. The training/validation accuracy and loss for facial expression analysis model are shown in Fig. 10 and Fig. 11.

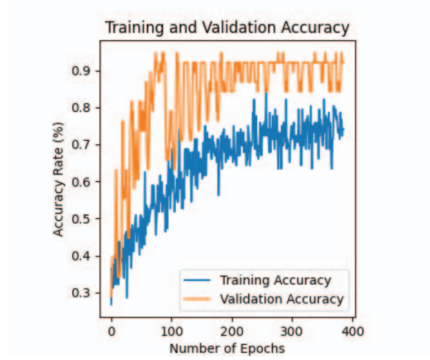

Fig. 10. Facial Expression Analysis Training and Validation Accuracy

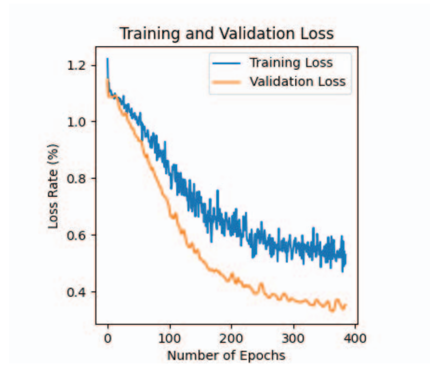

Fig. 11. Facial Expression Analysis Training and Validation Loss

D. Current Limitations

While the results of this research have provided a good baseline for future iterations of sign language recognition software, several limitations remain. The primary limitation pertains to the quality and quantity of publicly available datasets. While there are several publicly available datasets for the ASL alphabet, there are few that include a wide variety of words and phrases in the language. Furthermore, the majority of sign language datasets from different regions in the world are of lower quality, and there are not many datasets for other forms.

The other limitation present in our current resulting software is a lack of gesture recognition. In its current state, the resulting software only detects stationary letters, and does not support words or phrases. This is a heavy limitation as gesture recognition will play a vital role in developing a fully functional sign language recognition program.

E. Current Program Flow for Sign Language Recognition

The current program flow for landmark-based sign language recognition is as follows: First, the user launches the software which will turn on their camera, and the user selects a language option from the dropdown menu. Next, the user will choose which form of Sign Language Recognition they want to use. Specifically, they will choose between Realtime,

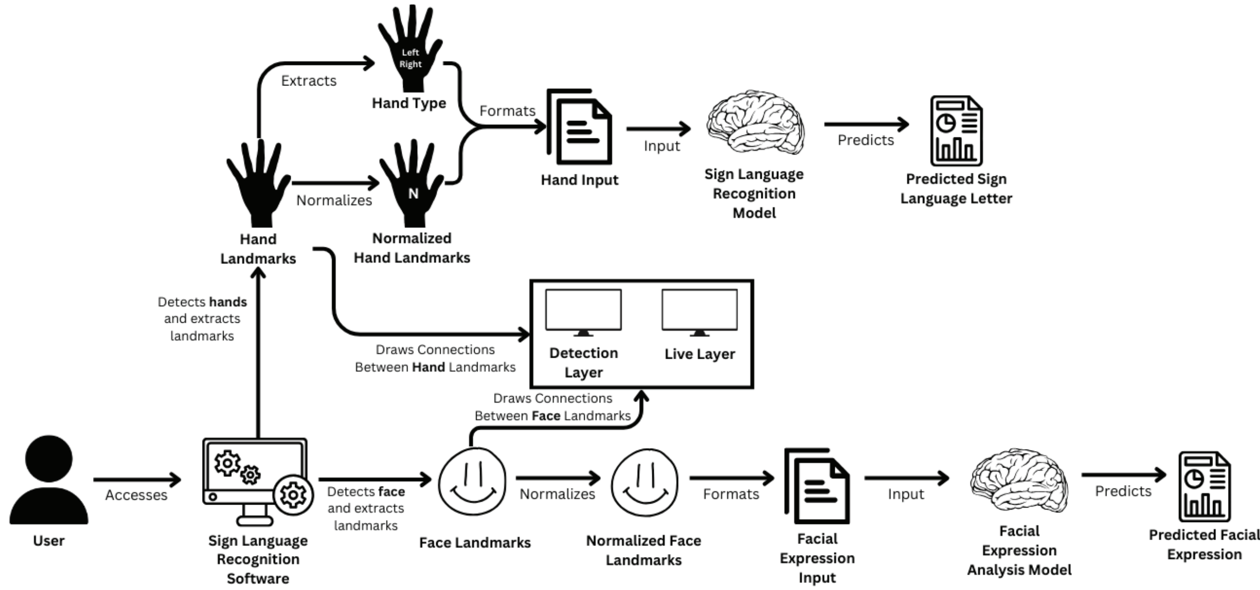

Fig. 12. Facial Expression Analysis Training and Validation Loss

Photo, and Video based recognition. Regardless of the user's choice, the software's recognition process works by concurrently detecting any hands and faces from the user's camera feed or input image/video. The software will only detect the two closet hands and singular closest face based on its current configuration in order to direct focus to the user. Once the hands have been detected, the software illustrates the connections between the hand landmarks onto the live layer and detection layer feeds using the OpenCV library. After that, the software extracts the hand type for each detected hand using the MediaPipe library classification values. After that, the software extracts the landmarks for each detected hand and normalizes the list of landmarks for each. These bits of information are combined and fed into a sign language recognition model based on the previously selected language option. The model then returns the predicted sign language letter value for the selected language.

Afterwards, the software extracts the facial landmarks from the detected face and normalizes the values. Next, using the MediaPipe library, a face mesh is generated and drawn onto the live layer and detection layer. The normalized values of the facial landmarks are then formatted and input into the facial expression analysis model, which returns the predicted facial expression value based on the user's input. The program flow has been visualized in Fig. 12.

VI. Conclusion

This paper summarizes our sign language system and illustrates the comparative analysis of the two investigated approaches for sign language recognition, as well as facial expression analysis. While there is a lot more work to be done for sign language recognition to be advanced enough to be used in the real world, we believe that this research will set the foundation for future iterations of our study.

In the future, we plan to investigate gesture recognition in order to allow motion-based gestures to be recognized. By doing so, we will be able to add support for proper words and phrases in sign language rather than solely the alphabet. Additionally, we intend to investigate the grammatical structure of sign language and develop methods for translating sentences from sign language into other languages and vice versa, while ensuring proper sentence structure. This effort aims to provide accurate translations as well as setting the foundation for a potential sign language learning tool. Furthermore, we aim to extend support for additional forms of sign language, such as Chinese Sign-Language (CSL) and many others, although we anticipate challenges in obtaining the large datasets required for training. Ultimately, we hope that this research makes a meaningful impact to help many individuals in the future and bridge the communication gap. In future iterations of our system, we aim to explore more advanced techniques for emotion recognition [25] to enhance sign language communication

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No. DUE-2247157 and DUE-2129795.

References

[1] R. E. Mitchell, "How Many Deaf People Are There in the United States? Estimates From the Survey of Income and Program Participation," in *Journal of Deaf Studies and Deaf Education*, vol. 11, no. 1, pp. 112–119, Oct. 2005. doi: 10.1093/deafed/enj004.

[2] J. D. Schein and M. T. Delk, "The Deaf Population of the United States.," eric.ed.gov, 1974. https://eric.ed.gov/?id=ed101517 (accessed Oct. 8, 2023).

[3] B. B. Blanchfield, J. J. Feldman, J. L. Dunbar, and E. N. Gardner, "The Severely to Profoundly Hearing-Impaired Population in the United States: Prevalence Estimates and Demographics," *Journal of the American Academy of Audiology*, vol. 12, no. 04, pp. 183–189, Apr. 2001. doi: 10.1055/s-0042-1745596.

[4] World Health Organization, “Deafness and Hearing Loss,” *WHO*, Feb. 27, 2023. https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss (accessed Oct. 8, 2023).

[5] T. Dacayan, D. Kwak and X. Zhang, “Computer-Vision Based Attention Monitoring for Online Meetings,” *2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI)*, Chengdu, China, 2022, pp. 533-538, doi: 10.1109/PRAI55851.2022.9904097.

[6] F. Ullah, H. Anwar, I. Shahzadi, A. Ur Rehman, S. Mehmood, S. Niaz, K. Mahmood Awan, A. Khan, and D. Kwak “Barrier Access Control Using Sensors Platform and Vehicle License Plate Characters Recognition,” *Sensors*, vol. 19, no. 13, p. 3015, Jul. 2019. doi: 10.3390/s19133015.

[7] H. Anwar, F. Ullah, A. Iqbal, A. Ul Hasnain, A. Ur Rehman, P. Bell, and D. Kwak, “Invariant Image-Based Currency Denomination Recognition Using Local Entropy and Range Filters,” *Entropy*, vol. 21, no. 11, p. 1085, Nov. 2019. doi: 10.3390/e21111085.

[8] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, and P. Presti, “American sign language recognition with the kinect,” *Proceedings of the 13th international conference on multimodal interfaces* (ICMI ’11), 2011. doi: 10.1145/2070481.2070532.

[9] N. C. Camgöz, O. Koller, S. Hadfield, and R. Bowden, “Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation,” *arXiv*, Mar. 2020. doi: 10.48550/arxiv.2003.13830.

[10] U. von Agris, M. Knorr, and K.-F. Kraiss, “The significance of facial features for automatic sign language recognition,” *IEEE International Conference on Automatic Face & Gesture Recognition*, Sep. 2008. doi: 10.1109/afgr.2008.4813472.

[11] E. A. Elliott and A. M. Jacobs, “Facial Expressions, Emotions, and Sign Languages,” *Frontiers in Psychology*, vol. 4, 2013. doi: 10.3389/fpsyg.2013.00115.

[12] J. Zheng, Y. Chen, C. Wu, X. Shi, and S. M. Kamal, “Enhancing Neural Sign Language Translation by highlighting the facial expression information,” *Neurocomputing*, vol. 464, pp. 462–472, Nov. 2021. doi: 10.1016/j.neucom.2021.08.079.

[13] N. E. Goldstein and R. S. Feldman, “Knowledge of American sign language and the ability of hearing individuals to decode facial expressions of emotion,” *Journal of Nonverbal Behavior*, vol. 20, no. 2, pp. 111–122, Jun. 1996. doi: 10.1007/bf02253072.

[14] C. Viegas, M. İnan, L. Quandt, and M. Alikhani, “Including Facial Expressions in Contextual Embeddings for Sign Language Generation,” *arXiv*, Feb. 2022. doi: 10.48550/arXiv.2202.05383.

[15] M. Hatano, S. Sako, and T. Kitamura, “Contour-based Hand Pose Recognition for Sign Language Recognition,” *Proceedings of SLPAT 2015: 6th Workshop on Speech and Language Processing for Assistive Technologies*, Sep. 2015. doi: 10.18653/v1/w15-5104.

[16] R. R. Itkarkar, Ashis Kumar Nandi, and B. K. Mane, “Contour-Based Real-Time Hand Gesture Recognition for Indian Sign Language,” *Advances in intelligent systems and computing*, pp. 683–691, Jan. 2017. doi: 10.1007/978-981-10-3874-7_65.

[17] K. Amrutha and P. Prabu, "ML Based Sign Language Recognition System," *2021 International Conference on Innovative Trends in Information Technology* (ICITIIT), Kottayam, India, pp. 1-6, 2021. doi: 10.1109/icitiit51526.2021.9399594.

[18] OpenCV, “OpenCV library,” Opencv.org, 2019. Accessed: Oct. 08, 2023. https://opencv.org/.

[19] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 6999-7019, Dec. 2022. doi: 10.1109/tnnls.2021.3084827.

[20] “MediaPipe Framework in Python,” Google Developers, https://developers.google.com/mediapipe/framework/getting_started/python_framework (accessed Oct. 8, 2023).

[21] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, “Recent Advances in Recurrent Neural Network,” *arXiv* preprint, 2018. doi: 10.48550/arXiv.1801.01078.

[22] Y. Yu, X. Si, C. Hu, and J. Zhang, “A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures,” *Neural Computation*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019. doi: 10.1162/neco_a_01199.

[23] A. Cossu, A. Carta, V. Lomonaco, and D. Bacciu, “Continual learning for recurrent neural networks: An empirical evaluation,” *Neural Networks*, vol. 143, pp. 607–627, Nov. 2021. doi: 10.1016/j.neunet.2021.07.021.

[24] V. Kimmelman, A. Imashev, M. Mukushev, and A. Sandygulova, “Eyebrow position in grammatical and emotional expressions in Kazakh-Russian Sign Language: A quantitative study,” *PLOS ONE*, vol. 15, no. 6, p. e0233731, Jun. 2020. doi: 10.1371/journal.pone.0233731.

[25] J. Singh, F. Ali, B. Shah, K. S. Bhangu and D. Kwak, “Emotion Quantification Using Variational Quantum State Fidelity Estimation,” *IEEE Access*, vol. 10, pp. 115108-115119, 2022, doi: 10.1109/access.2022.3216890.