Improving Sign Language Recognition Performance Using Multimodal Data

Tomoe Nishimura
Computer Science
California State University, Channel Islands
Camarillo, California, USA

Bahareh Abbasi
Computer Science
California State University, Channel Islands
Camarillo, California, USA

***Abstract*—Sign language is a language primarily used by the hearing-impaired for communication and has more than 200 variations worldwide. Communication is nearly impossible between signers of different variations. Moreover for a person with normal hearing, learning sign language can be challenging because the syntax of sign language differs from that of natural language. Translation of signs by machine learning offers potential solutions to these challenges, facilitating communication for everyone. This study attempts to enhance the performance of the existing state-of-the-art sign language translation model, Gloss attention SLT network (GASLT), through the integration of a multimodal approach. By combining RGB video with 3D pose data extracted using Mediapipe in an innovative way, our multimodal method significantly enhances the GASLT's results. We conducted two experiments involving the fusion of video and pose data with the GASLT model. These experiments led to an 18.39% improvement in the model's BLEU score compared to the original model, showcasing the effectiveness of the multimodal approach in enhancing sign translation.**

***Index Terms*—sign language, computer vision, transformer, Mediapipe, multimodal.**

I. Introduction

Sign language, a non-vocal communication system primarily used by the hearing impaired, is employed by approximately 70 million people worldwide. There exist over 200 variations of sign language, each with its unique grammar and expressions [1]. Key elements in sign language include hand gestures, full-body postures, and facial expressions. While finger movements are often emphasized, non-verbal aspects like exaggerated expressions and facial cues play a crucial role in sign language, enhancing overall context [2].

Communication through sign language has two challenges: first, learning sign language is difficult for people with normal hearing because sign languages and natural languages are not syntactically identical [3]. Second, because there are many variations of sign language, most of which are distinct, communication between signers of different sign languages is nearly impossible. Thus, to overcome these difficulties, researchers are exploring several solutions using machine learning.

The task of translating a sign language video into natural language is called Sign Language Recognition (SLR); and various modalities have been tried, including sign language image/video input [3], depth camera input [4], and the use of pose estimation solutions [5].

Inspired by these previous studies, we hypothesized that combining data from multiple modalities could improve the recognition accuracy of sign language translation (SLT). Our approach aimed at enhancing the performance of the state-of-the-art SLT model, GASLT [6], by adopting a multimodal approach which involves the integration of video input (RGB images) and pose estimations by Mediapipe, the open source pose estimation library by Google [7].

GASLT, a gloss-free SLT model was originally trained solely on video input without utilizing glosses, which were dictionaries indicating word relations across languages. In this work, we introduced multimodality to enhance its capabilities and explore two different approaches to incorporate it: 1) *Direct Input Fusion Experiment:* aiming at assessing the impact of integrating diverse multimodal data types as direct input on the model's performance to recognize sign language; 2) *Internal Integration Experiment:* focusing on embedding multimodal data within the network's architecture, specifically at mid or final layers. Our results demonstrated a significant accuracy improvement of 18.39% compared to the original model (with only the image).

II. Related Work

Sign language recognition (SLR) is a machine learning task that translates visual elements of sign language into text [8]. This task can be accomplished through various methods such as employing wearable motion-capture devices with visual features [4] or utilizing a 3D depth camera for data input [4], [9]–[11]. Previously, gesture estimation and hand tracking in SLR tasks were challenging [10]. However, the advent of open-source pose estimation models such as OpenPose [12] and Mediapipe [7] permitted the use of regular camera-captured sign language images and videos as input.

SLR using video as input can be divided into two types: word-level sign language recognition (WSLR) and sign language translation (SLT). WSLR involves recognizing a single sign language word with an action, while SLT translates sequential sign language expressions in videos into natural language sentences [3]. The task of SLT is often approached as an automatic machine translation task between multiple languages, a typical task in Natural Language Processing (NLP). Therefore, evaluation measures such as BLEU [13],

2835-5776/24/$31.00 ©2024 IEEE
DOI 10.1109/IRI62200.2024.00047

which measures the n-gram matching rate between the generated translation and the reference translation, and Rouge [14], which considers the longest common subsequence and other factors in addition to the n-gram, are applied to SLR.

Numerous models were proposed for sign language translation/recognition, among which the transformer-based models stand out as a powerful and reliable approach. Transformer [15], introduced by Vaswani et al., enhanced the encoder-decoder [16] framework by integrating a self-attention mechanism within both encoders and decoders. Among the distinguished examples of this architecture were the Sign Language Recognition Transformer (SLRT) [17] and the Temporal Semantic Pyramid Network (TSPNet) [3].

SLT was seen as a particularly challenging task compared with WSLR. This was because it was difficult for machine learning to detect segments of a sign language expression and ensure consistency between sign language and natural language. For example, several prior studies employing Mediapipe for WSLR have recorded high recognition accuracies of about 99% [5], [18]. Nevertheless, no models have yet achieved the same degree of recognition accuracy in SLT.

In the SLT field, the Sign2Gloss2Text method [17] used a gloss annotation list detailing word-level correspondences between sign and natural language. While this approach enhanced sign language translation accuracy [17], compiling the gloss list required sign language domain knowledge and expertise, and was a daunting task [6]. Therefore, the Sign2Text model [19] was introduced as a solution to translate sign language videos directly into text without using a gloss list.

The Gloss attention SLT network (GASLT) [6], a Sign2Text model, employed gloss attention to replace the Sign2Gloss2Text model's gloss role, dynamically focusing on adjacent frames to discern semantic boundaries. The RWTH-PHOENIX-Weather-2014T dataset, which served as the input to this model, was first passed to the I3D model [20], a convolutional neural network developed for human activity recognition. The extracted visual features from the last layer of this network were then fed into the GASLT.

In this study, we employ the GASLT model which originally works with RGB video as input. We proposed to improve sign language recognition accuracy by combining the RGB videos and the 3D pose estimations via Mediapipe.

III. Experiments

In our experiments, we have used RWTH-PHOENIX-Weather-2014T dataset [21], a frequently used dataset for sign translation tasks [3], [6], [22]. This dataset, featuring news and weather broadcast signs from German public television stations, has been segmented into three subsets: train, dev, and test. These subsets contain 7,096, 519, and 642 videos, respectively. Each video has been recorded at 25 frames per second and has a resolution of 210×260 pixels.

To create multimodal data, we generated the 3D pose estimations of RGB video frames using Mediapipe [7], a pose estimation library. This library estimates the coordinates of each distinct landmark point, 478 locations on the face, 42 locations on the hands, and 33 locations on the body, all within the three-dimensional Cartesian coordinate system. In our experiments, we considered different combinations of these landmarks and trained the GASLT model [6] with the resulting multimodal data. We have conducted two sets of experiments (refer to Figure. 1):

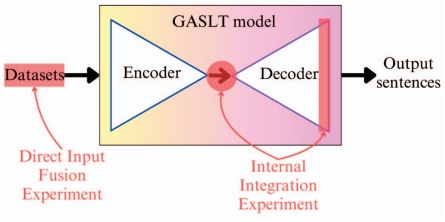

Fig. 1. The representation of the two conducted experiments: 1) the Direct Input Fusion Experiment that uses the multimodal data as the input, and 2) the Internal Integration Experiment that changes the network architecture either in the middle or at the end of the GASLT model.

1) Direct Input Fusion Experiment: In this approach, we tried different variations of combining the multimodal data and using it as an input for training the GASLT model.
2) Internal Integration Experiment: In this approach, instead of simply combining the landmark data with RGB images, we explored the idea of adding them to the middle or the end of the network.

All experimental outcomes were benchmarked against the baseline, the original GASLT model trained with only raw RGB videos.

The experiments were conducted using a system equipped with an Intel(R) Core(TM) i5-8600K CPU (3.60 GHz) and 32GB of RAM. The GPU initially utilized was a GEFORCE GTX 1070 Ti, but due to a malfunction, it was replaced by a GEFORCE RTX 4060 Ti during the experiment. Given that the GPU performance was not believed to impact the model's performance or accuracy [23], the experiment proceeded. Python 3.7.10, PyTorch 1.13.1, and Mediapipe 0.10.5 were utilized for model implementations.

A. Data Preprocessing

Similar to the original GASLT model, we pre-processed each video frame and extracted visual features from the final layer (before the output layer) of the I3D model [20]. This results in 1024 features for each frame. During the I3D feature extraction, each frame within a video was assigned a "span", defined as a group of consecutive frames. This was important for discovering semantic connections between frames, because in sign language representations, a single word is often depicted across multiple frames. In our experimental setup, we set the span to 8. For each frame, 9 total sequential frames comprised of 4 preceding frames, the current frame, and 4 proceeding frames were input into the I3D model to generate the visual features. If four proceeding/preceding frames are not available, additional frames were sourced from the opposite

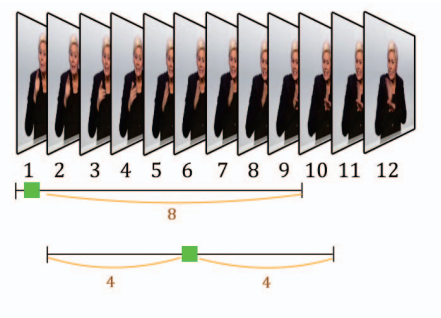

Fig. 2. Example of video prepartion with span of 8. The frames are selected from the RWTH-PHOENIX-Weather-2014 dataset [21]. The green point represents a current frame. with span of 8, a total of 9 frames, comprising the current frame as well as 8 frames both before and after it, is considered.

direction to maintain a constant total frame count. An example of this step is shown in Figure. 2.

B. Direct Input Fusion Experiment

In this experiment, we have explored 10 different scenarios for creating the multimodality and training our base model accordingly. Our baseline employed the raw RGB video frames, akin to the original GASLT model. To assess the effectiveness of facial expression for recognition, we considered the "hand + pose" and "full annotation (hand + pose + face)" scenarios. Both scenarios utilized annotated original RGB video frames by landmarks estimated via Mediapipe. In these scenarios, each landmark's information was depicted by green dots and lines on the raw RGB video frames. The "hand + pose" covered a total of 50 estimation points with 42 allocated to the hand and 8 to symmetrical body parts such as shoulders, elbows, wrists, and hips. In the "full annotation" scenario, in addition to the estimation points mentioned above, 17 facial landmarks from the mouth, eyebrows, and pupils were included. To prevent obscuring the original RGB frame content with green annotated dots and lines, we adhered to a total of 67 estimation points.

To evaluate the impact of color-coded versus single-color (only green) points annotations for body parts, we developed the "colored full annotation" scenario. This scenario mirrored the "full annotation" setup in terms of point selection, but introduced different colors for the dots and lines representing each body part. Our initial hypothesis was that color variations would enhance the model's interpretation of pose estimation, potentially leading to improved recognition accuracy compared to using a single color.

In the "colored skeleton" scenario, we retained the same annotations as in the "colored full annotation" scenario, but using a single-color background instead of raw RGB images. This choice was driven by the hypothesis that isolating the annotations on a single-color background would enhance the performance by reducing the noise from the raw RGB image. Examples of these scenarios are shown in Figure. 3. In all of these scenarios, the GASLT model was trained using visual features extracted from the multimodal annotated

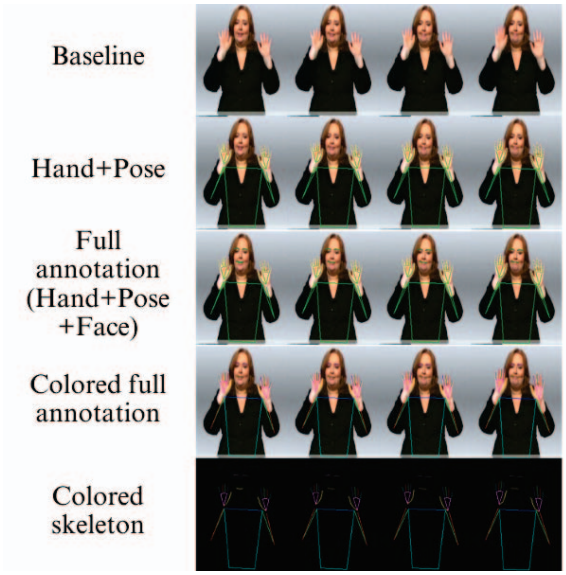

Fig. 3. The generated multimodal data in different scenarios. The video frames are selected from the RWTH-PHOENIX-Weather-2014 dataset [21]. See Table I for details on each scenario.

frames generated in these scenarios. In each case, the videos were processed according to the explained pre-processing step. This tensor object representing the video was paired with a sentence, the natural language translation of the video, and the file, compressed as a list, became the input to the actual GASLT model. Our training setup was the same as the one used for the original GASLT model in [6].

For our next scenarios, namely "baseline + colored full annotation", "baseline + colored skeleton", and "dual baseline", we combined the raw RGB frames features with the annotated RGB frames with landmarks data. In each scenario, the features were created by combining I3D model output tensors from the raw RGB frames with the ones created from selected annotated frames resulting in a total number of 2048 features per frame. The final scenario in this category, "dual baseline" was simply the combination of extracted I3D features from raw RGB frames and their replicated as the input to the model.

The final three scenarios employed pose estimation coordinates to create the input features, we call them: "small coordinates", "large coordinates", and "baseline + small coordinates" respectively. In the "small coordinates" scenario, we used 2D coordinates of 17 landmark points associated to the mouth, eyebrows, and eyes in addition to the 33 pose and 42 hands landmarks (total of 92 coordinates). For the "large coordinates" scenario, we used the same setup including the 2D coordinates of all face landmarks resulting in a total of 553 coordinates. Given the total of 478 face landmarks, these two scenarios were designed to measure the impact of the face landmarks on the model performance.

In these scenarios, pose estimation data underwent a cleansing process due to multiple missing values, which occurred

TABLE I
DIRECT INPUT FUSION EXPERIMENT T-TEST SUMMARY (COMPARED AGAINST THE BASELINE). P-VALUE THRESHOLD IS < 0.05.

Scenario	Feature Description	BLEU		Rouge	
		t	p	t	p
Hand + Pose	Green landmarks annotations for hands, and pose drawn over the raw image	-7.61185	<.00001	-3.09654	0.003113
Full annotation	Green landmarks annotations for hands, pose, and face drawn over the raw image	-7.95699	<.00001	-5.82165	<.00001
Colored full annotation	Color coded landmarks annotations for hands, pose, and face drawn over the raw image	-8.40591	<.00001	-5.55343	0.000014
Colored skeleton	Color coded landmarks annotations for hands, pose, and face drawn over the single-color background	-12.75315	<.00001	-14.30877	<.00001
Baseline + Colored skeleton	Combined I3D features "baseline" and "Colored skeleton"	4.10205	0.000335	4.45639	0.000152
Baseline + Colored full annotation	Combined I3D feature of "baseline" and "Colored full annotation"	10.17582	<.00001	8.0528	<.00001
Dual baseline	Combined two baseline I3D features	1.04057	0.155925	-0.7222	0.239726
Baseline + Small coordinates	Combined I3D features from "baseline" with "small coordinates"	12.92535	<.00001	18.05847	<.00001
Large coordinates	2D landmark coordinates, including hands, body, and all face points	-4.99934	0.000046	-5.43067	0.000018
Small coordinates	2D landmark coordinates, including hands, body, and 17 points of face	8.35702	<.00001	8.81824	<.00001

when Mediapipe failed to estimate during large movements by signers. Missing frame values were estimated by averaging adjacent frames. However, videos that lacked any estimable frames, totaling 41 out of 8257, were excluded, leaving 8216 videos for feature extraction. Following the cleansing process, datasets were standardized using the formula: (data - dataset mean) / dataset standard deviation, to bridge the numerical gap with the I3D features, ensuring effective recognition by the model upon merging. This data cleaning process was applied exclusively to features that included coordinates. In contrast, features that overlay an annotation on the image, which were based on Mediapipe's estimation and drawn on the original video frames, remained functional even in the presence of missing data and hence were exempt from data cleansing.

C. Internal Integration Experiment

In this experiment, we tried 5 different experimental configurations. In each configuration, we used the raw RGB video frames pre-processed by the I3D model as the main input to the model. Contrary to the approach in the Direct Input Fusion Experiment, where landmark points were directly integrated with the raw image, this experiment introduced a different strategy. Here, the integration of landmark data within the model occurred either mid-way between the encoder and decoder or at the terminal stage of the decoder.

For adding the landmarks to the middle of the model, we explored 3 configurations. In the first configuration, the "coordinate-direct", we used the "small coordinate" feature and directly combined them with the encoder output. In the next configuration, the "colored full annotation-direct", the I3D extracted feature from the "Colored full annotation" setup was added to the encoder output. In the last configuration named the "coordinate-new encoder", we added a second encoder, identical to the initial GASLT encoder, to process the landmark data. The outputs from both encoders were then merged and passed to the decoder. In other words, this configuration included 2 encoders and 1 decoder.

In order to add the landmark data to the end of the model, we tried 2 different configurations. These configurations only affected the model's decoder. In the final stages of the GASLT model, a normalization layer followed by a linear layer was used for output generation. The "coordinate-before norm" configuration merged the landmarks data with the decoder feature before the normalization layer. The "coordinate-after norm" configuration, integrated landmarks with the decoder features after the normalization layer.

D. Metrics

To evaluate the performance of our model in each scenario, we employed BLEU-4 [13], which measures 4-gram matching rates, and Rouge-L [14], which considers the longest common subsequence, due to their popularity in the field. To assess whether adding multimodality improves results compared to the baseline model, we conducted a t-test (with 95% confidence intervals). In the GASLT model, each scenario and feature sets or experiment configurations underwent 10 iterations of training and testing, with parameters set to GASLT's initial values. The scheduler managed the learning rate and epochs, automatically adjusting the learning rate at each training step and halting training when no accuracy improvement was detected. The model state yielding the best results was saved for testing purposes. The test results from this best training state were adopted as the final results.

The average, maximum, and t-test values of the experiments are reported in Table I. The t-test, with a p-value set at 0.05, assessed the impact of multimodal feature modification on the results, with the hypothesis that the method outperforms the baseline if the t-value is a non-negative number and the p-value is less than or equal to 0.05. The baseline performance

TABLE II
T-TEST RESULTS IN INTERNAL INTEGRATION EXPERIMENT COMPARED WITH THE BASELINE. P-VALUE THRESHOLD IS < 0.05.

	BLEU		Rouge	
	t	p	t	p
Middle: coordinate-direct	-15.95274	<.00001	-14.92574	<.00001
Middle: coordinate-new encoder	-17.92434	<.00001	-19.56225	<.00001
Middle: colored full annotation-direct	-7.75038	<.00001	-11.6803	<.00001
End: coordinate-before norm	-7.19151	<.00001	-8.17744	<.00001
End: coordinate-after norm	-7.56444	<.00001	-5.15086	<.00001

reported in this paper is resulted from our replication of the GASLT setup as described in [6].

IV. RESULT AND DISCUSSION

A. Experiment Results

1) Direct Input Fusion Experiment: Three scenarios, "baseline + colored full annotation", "baseline + small coordinates", and "small coordinates", recorded higher results than the baseline. The recognition accuracy for "hand + pose", "full annotation", "colored full annotation", and "colored skeleton" were lower than the baseline. Features used in the "baseline + colored skeleton" and "baseline + colored full annotation" scenarios scored higher recognition accuracy than the baseline. On the other hand, the p-value of the "dual baseline" was above 0.05, which means this feature extension strategy adversely impacted the recognition accuracy. The multimodal features introduced in the "baseline + small coordinates" and "small coordinates" scenarios, improved the baseline results, while the "large coordinates" scenario did not positively affect the performance. In particular, the multimodal feature used in the "baseline + small coordinates" scenario produced the best results and outperformed the rest of the scenarios including the baseline. This feature results in 18.39% higher for BLEU and 7.85% higher for Rouge scores compared to baseline. The results of the t-test comparing each scenario with the baseline can be found in Table I, and refer to Table III for the maximum score for each feature.

2) Internal Integration Experiment: The results of this experiment showed that the performance of the models trained via this approach was worse than the baseline approach in all the tried configurations. The highest scores among all configurations were 14.44 (End: coordinate/before norm) for BLEU and 38.22 (End: coordinate/after norm) for Rouge. Both results were below the baseline BLEU of 15.03 and Rouge of 39.13. The results of the t-test comparing each configuration with the baseline can be found in Table II, and refer to Table III for the maximum score for each configuration.

TABLE III
MAXIMUM TEST RESULT SCORES

	BLEU	Rouge
Baseline	15.03	39.13
Ex1: Hand + Pose	14.56	38.62
Ex1: Full annotation	14.35	38.42
Ex1: Colored full annotation	14.39	38.39
Ex1: Colored skeleton	13.27	36.22
Ex1: Baseline + Colored skeleton	15.62	40.08
Ex1: Baseline + Colored full annotation	17.24	41.07
Ex1: Dual baseline	15.36	39.06
Ex1: Baseline + Small coordinates	**17.58**	**41.76**
Ex1: Large coordinates	14.84	38.60
Ex1: Small coordinates	16.69	40.84
Ex2: Middle/ coordinate/direct	13.53	36.78
Ex2: Middle/ coordinate/new encoder	12.68	34.59
Ex2: Middle/colored full annotation/direct	14.14	37.30
Ex2: End/ coordinate/before norm	14.44	37.81
Ex2: End/ coordinate/after norm	14.40	38.22

B. Discussion

To summarize the results of the Direct Input Fusion Experiment, the approach of extending the number of features using 2D coordinate data recorded better results and outperformed the rest of the scenarios. All three initial scenarios, "hand + pose", "full annotation", and "colored full annotation", demonstrated that simply overlaying skeletons annotations on the raw image frames cannot enhance the recognition accuracy. A possible justification can be the additional annotations on the image may simply be treated as noise on the frames and impact the training adversely. In addition, the "colored skeleton" scenario resulted in a significantly lower score than the "colored full annotation", and also performed the worst of all features in different scenarios of the Direct Input Fusion Experiment. From these results, we can conclude that removing visual features does not have a positive impact on the model.

Next, for the "baseline + colored skeleton" and the "baseline + colored full annotation" scenarios, the number of features was increased by combining two video frame features, which contributed significantly to the score improvement. Since the score did not improve for the "dual baseline" scenario, it was also found that extending the raw frames with annotated frames (including multimodal data) was actually effective. This approach involved combining multiple frame variations, both with and without annotation, into a single frame of data. As a result, the frames became multimodal data, and the richness and modality of the data were enhanced due to the merging of multiple data sources.

The "large coordinates" scenario, which included all facial landmark coordinates, scored below the baseline, while the "small coordinates", focused on specific facial landmarks,

scored above the baseline. This indicates that coordinate data is beneficial for training the model, however, including too many facial landmarks may impede recognition performance.

Unlike the Direct Input Fusion Experiment, which produced promising outcomes, the results of the Internal Integration Experiment indicate that none of the configurations scored above the baseline. The Transformer model consists of an encoder and a decoder, optimized under the assumption that the output of the encoder serves as the input for the decoder. Incorporating data in the middle or at the end of the model could be regarded as noise, potentially disrupting this optimized process.

One interesting observation was adding the landmarks data at the end of the decoder scored better than adding it in the middle of the encoder and decoder. When landmark data were added in the middle of the model, both the encoder and the decoder were affected since the encoder's output was returned to the upper layers separately from the decoder's output, and was used directly in the loss calculation.

In summary, these results show that the multimodal data augmentation approach using Mediapipe can significantly improve recognition accuracy.

V. Conclusion and Future Work

Increasing the accuracy of machine learning-based sign language recognition holds significant importance in facilitating communication with the hearing-impaired community. In this work, we propose a novel multimodal approach to improve the performance of the state-of-the-art sign language recognition model, GASLT [6].We used the RWTH-PHOENIX-Weather-2014T dataset [21] and conducted two sets of experiments to introduce the multimodal approach to the model. In the Direct Input Fusion Experiment, multimodal features were combined and utilized as the input for model training. In the Internal Integration Experiment, multiple modalities of data were integrated into the middle and the end of the model. Our best experiment result was a BLEU score of 17.58, 18.39% higher than the baseline (trained with only raw video frames). Our results show that combining RGB video features with Mediapipe's body landmark coordinates outperforms the rest of the scenarios including the baseline.

In conclusion, experiments indicated that incorporating multiple modalities of input data at the start of the model effectively enhances sign language recognition accuracy. By enriching the input with the output from the pose estimation solution, the model can capture more nuanced sign language behavior, which may have contributed to improved recognition performance. The application of this method to other SLT models can be a subject for future research.

References

[1] W. F. of the Deaf, "Our work," 2021, accessed on November 13, 2023. [Online]. Available: https://wfdeaf.org/our-work/

[2] M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 1, pp. 131–153, Jan 2019. [Online]. Available: https://doi.org/10.1007/s13042-017-0705-5

[3] D. Li, C. Xu, X. Yu, K. Zhang, B. Swift, H. Suominen, and H. Li, "Tspnet: Hierarchical feature learning via temporal semantic pyramid for sign language translation," 2020.

[4] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, and P. Presti, "American sign language recognition with the kinect," in *Proceedings of the 13th International Conference on Multimodal Interfaces*, ser. ICMI '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 279–286. [Online]. Available: https://doi-org.ezproxy.csuci.edu/10.1145/2070481.2070532

[5] G. H. Samaan, A. R. Wadie, A. K. Attia, A. M. Asaad, A. E. Kamel, S. O. Slim, M. S. Abdallah, and Y.-I. Cho, "Mediapipe's landmarks with rnn for dynamic sign language recognition," *Electronics*, vol. 11, no. 19, 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/19/3228

[6] A. Yin, T. Zhong, L. Tang, W. Jin, T. Jin, and Z. Zhao, "Gloss attention for gloss-free sign language translation," 2023.

[7] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann, "Mediapipe: A framework for building perception pipelines," 2019.

[8] R. Rastgoo, K. Kiani, and S. Escalera, "Sign language recognition: A deep survey," *Expert Systems with Applications*, vol. 164, p. 113794, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095741742030614X

[9] A. Kuznetsova, L. Leal-Taixe, and B. Rosenhahn, "Real-time sign language recognition using a consumer depth camera," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, June 2013.

[10] J. Huang, W. Zhou, H. Li, and W. Li, "Sign language recognition using 3d convolutional neural networks," in *2015 IEEE International Conference on Multimedia and Expo (ICME)*, 2015, pp. 1–6.

[11] T. Liu, W. Zhou, and H. Li, "Sign language recognition with long short-term memory," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 2871–2875.

[12] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," 2019.

[13] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ser. ACL '02. USA: Association for Computational Linguistics, 2002, p. 311–318. [Online]. Available: https://doi-org.ezproxy.csuci.edu/10.3115/1073083.1073135

[14] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://aclanthology.org/W04-1013

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023.

[16] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," 2014.

[17] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Sign language transformers: Joint end-to-end sign language recognition and translation," 2020.

[18] D. Gandhi, K. Shah, and M. Chandane, "Dynamic sign language recognition and emotion detection using mediapipe and deep learning," in *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2022, pp. 1–7.

[19] Z. Liang, H. Li, and J. Chai, "Sign language translation: A survey of approaches and techniques," *Electronics*, vol. 12, no. 12, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/12/2678

[20] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," 2018.

[21] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, Dec. 2015.

[22] L. T. Woods and Z. A. Rana, "Modelling sign language with encoder-only transformers and human pose estimation keypoint data," *Mathematics*, vol. 11, no. 9, 2023. [Online]. Available: https://www.mdpi.com/2227-7390/11/9/2129

[23] D. Gyawali, "Comparative analysis of cpu and gpu profiling for deep learning models," 2023.