International Journal of Science and Research Archive
eISSN: 2582-8185
Cross Ref DOI: 10.30574/ijsra
Journal homepage: https://ijsra.net/

(RESEARCH ARTICLE)

Check for updates

SignSpeak – Sign language translation system for hearing impaired

G. Kalaiselvi [1,*], N. Badri [2] and C. Karnan [2]

[1] *Assistant Professor, Department of Computer Science and Engineering, Anjalai Ammal Mahalingam Engineering College, Kovilvenni, Tamil Nadu, India.*
[2] *UG Student, Department of Computer Science and Engineering, Anjalai Ammal Mahalingam Engineering College, Kovilvenni, Tamil Nadu, India.*

International Journal of Science and Research Archive, 2025, 15(02), 921-930

Publication history: Received on 11 April 2025; revised on 21 May 2025; accepted on 23 May 2025

Article DOI: https://doi.org/10.30574/ijsra.2025.15.2.1523

Abstract

The deaf and hard-of-hearing community face significant challenges navigating through daily-life. Difficulties such as hearing impairments or speech disabilities often limit communication for the community which is a crucial aspect of human life. To bridge this gap, Sign Language, is a methodology that involves hand movements with face interaction that acts a medium to converse is utilized to convey the ideas that a hard-of-hearing wishes to share. However, Sign Language exists in different accents and does not follow a universal common language and is quite uncommon for an average group of people to have knowledge base regarding it. There are translators that have Sign Language understanding that interpret information to the hearing-impaired via hand signs. But they exist few in numbers along with their availability varying at times. In such situations, hearing-impaired community cannot actively participate in interactions. SignSpeak, a real-time translation system is developed that records audio input from the user and provides the transcribed sign gloss clips that interprets the user's vocabulary in sign language. Bidirectional Encoder Representation from Transformers (BERT) - a deep learning algorithm combines PyAudio to obtain audio input with OpenAI's Whisper model to generate text transcription and reorder them. It generates tokens which are then read by the FFmpeg module for sign gloss video retrieval and stitching and present a complete video output.

Keywords: Real-time Speech Translation; American Sign Language (ASL); Whisper ASR; BERT Transformer; Gloss Video Synthesis; Accessibility Technology; Multimodal Communication

1. Introduction

Communication is an integral part of human interaction. Yet millions of people across the world are faced with great difficulties in communication and reception because of hearing loss or speech disabilities. The World Health Organization (2024) has stated that more than 430 million people in the world suffer from disabling hearing loss. This state has profound impacts on several aspects of life, including learning, employment, social communication, and access to information. For instance, individuals who are predominantly users of sign language often encounter severe challenges in learning verbal material because they don't process or translate auditory input automatically into their visual mode of communication, especially in dynamic, real-time settings like classroom lectures, meetings, or TV information. The mental load of having to constantly decipher spoken language is heavy, and it causes fatigue and misunderstanding. And to that is added the complexity created by the differences between spoken language grammar and sign language grammar.

The need for accessible communication solutions for the hearing-impaired community has driven significant research and development efforts. Assistive technologies work towards bridging the communication gap and ensuring inclusiveness. The recent combination of natural language processing (NLP) and artificial intelligence (AI) has opened

* Corresponding author: G. Kalaiselvi

new opportunities for automatic speech-to-sign language translation. These new developments hold the promise of creating systems that are capable of understanding spoken language and producing resulting sign language visualizations in real-time, thus making communication smoother and more natural.

SignSpeak is an implementation of a speech-to-sign language translation system that integrates state-of-the-art technologies such as OpenAI's Whisper Automatic Speech Recognition (ASR) model and BERT for semantic reordering. In the present system, spoken language transcripts in American Sign Language (ASL) work by combining ASR, semantic processing, and video retrieval. A well-known dataset by the name of WLASL allows the system to search for relevant sign language video clips that show the gestures for the translated speech in real-time via video output synthesis.

At the heart of this approach is Whisper, which allows high-performance ASR and guarantees that speech is accurately transcribed in many environments. The translation is augmented further using BERT-based semantic reordering, taking into consideration the fact that sign languages follow different grammatical structure patterns as opposed to spoken ones. The system uses fallbacks with lemmatization and synonyms to cover for ambiguities or lack of clarity with respect to hard concepts for words or expressions found in the speech.

This effort is aimed at augmenting the already flourishing AI-based sign language translation arena by providing a competent and scalable solution that accounts for real-time benefits. A broader goal is to aid in communication for hearing and speech impairments and elucidate how mixed AI models may assist in enhancing such real-time language translation systems.

2. Literature Review

Deep learning and AI have significantly advanced machine translation for sign languages. The systems rely heavily on Automatic Speech Recognition methods to convert spoken language into text. Amongst the notable ASR solutions is Whisper, which can recognize speech with impressive speed and accuracy even in the presence of noise and across languages [1, 13, 18]. The ability of Whisper to transcribe spontaneous speech with several accents allows it to feed higher-quality transcriptions into real-time translation systems, which is key for downstream tasks such as semantic reordering.

In addition to ASR, semantic reordering is crucial in ensuring that translations adhere to the grammatical structure of sign languages, such as American Sign Language (ASL). The deep-learning transformer BERT has proven advantageously effective in sentence reordering tasks due to its bidirectional encoding, which enables understanding and restructuring sentences grammatically [2, 11].

Additionally, another aspect of real-time sign language translation systems is video synthesis, where a pre-recorded gloss clip is used to synthesize corresponding sign language videos to the translated text. For FFmpeg, it is, such that it is an elegant stitch in using all the translation output to give that seamless effect [3]. In addition, advancements in sign language datasets, such as those at work on video-to-word recognition, will benefit the creation of gloss-based sign language systems [4, 22]. With better gesture recognition and improved accuracy, video synthesis can become even more precise [19, 20].

Sign language translation systems also need to effectively resolve linguistic ambiguities, and fallback mechanisms like lemmatization and synonym matching serve their purpose. These mechanisms help these systems interpret a variety of expressions, phrases, or alternative wordings in the input speech, thereby enhancing the system's robustness to different speaking styles while assisting the preservation of the meaning in the translation [10, 16].

Evaluation metrics form an important measure of the success of any ASR or translation system. JiWER (Just Another Word Error Rate), for example, is commonly employed in evaluating ASR systems with respect to transcriptional accuracy [5, 6]. In a similar vein, sign language translation systems may also be evaluated based on criteria like translation accuracy, video quality, and real-time processing speed.

Real-time sign language translation systems must also deal with multimodal inputs. Research on multimodal learning explores how combining both visual and auditory signals can improve performance, particularly when ASR and sign language recognition are synchronized [7]. By leveraging both the spoken language and sign gestures, these systems become more robust in challenging real-world scenarios, such as when background noise or ambiguous speech is present.

Recent developments in the field also call for high scalability of these systems: the models are trained using large-scale datasets to ensure their generalizability over a wide range of sign languages as well as contexts. This is particularly useful in open-domain sign language translation, where systems have to adapt to a variety of topics and conversational styles [8, 12, 21].

Real-time sign language translation proves to be a complicated task; thus, multilingual systems are adopting between different sign languages and between sign language and speech translation further increases accessibility. An example of this multilingual system is Sign.mt, which offers real-time translation between signing systems [9, 15, 17, 14]. Apart from these advances, the AI-driven models keep being improved to ensure that they can handle different gestures, cultural variations, and real-time interaction, thus improving the end-user experience of deaf and hard-of-hearing people.

To summarize, progress in ASR with Whisper, in semantic reordering with BERT, video synthesis with FFmpeg, and fallback in linguistic ambiguity can be taken up in strides, which, by the rapid pace at which they set, will certainly improve the effectiveness and usability of sign language translator systems. Integration of such technologies also does not enable these systems to become accurate in translation but also makes it possible for them to function in real-time, thus closing the communication gap for hearing-impaired individuals.

2.1. System Architecture

The envisioned speech-to-sign translation system is built as a pipeline of modules, with each module handling a distinct transformation task. The architecture emphasizes real-time processing, robustness, and accessibility to users by combining state-of-the-art deep models and conventional processing approaches. From audio input to sign language video output, the process transitions seamlessly, with semantic integrity and visual consistency ensured throughout.

The system is operated by the user using the Graphical User Interface (GUI) created with Tkinter. Here, processing is triggered by input of text or audio by the user. The interactive graphical interface shows the transcription of the speech and the resulting output video. Besides providing smooth playback of the synthesized sign language video, the GUI, integrated with OpenCV, provides controls to record, start processing, and play back the output. Real-time feedback is illustrated as maintaining transparency, with identified text and gloss mappings.

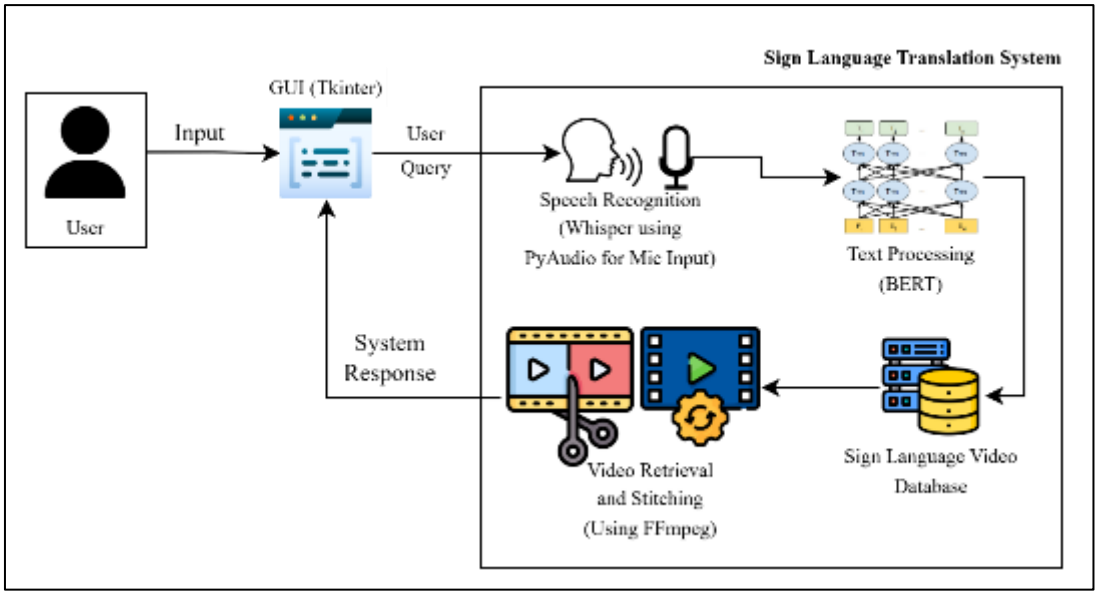

Figure 1 System Architecture

The Core System then invokes a multi-step processing pipeline on receiving the input. The input voice of the user is recorded in mono-channel and 16-bit audio with a sample rate of 16 kHz by using PyAudio for Speech Recognition. The audio, after it's saved as ".wav" file, gets processed by OpenAI's Whisper model, renowned for its multilingual capabilities and resilience to noise-prone environments. This particular module gives clean transcription of spoken content in English language.

The transcription is then passed to the BERT-based Semantic Reordering algorithm in the following step, Text Processing. The system relies on NLTK to perform part-of-speech tagging alongside BERT by Hugging Face's Transformers. The words are rearranged from typical Subject-Verb-Object (SVO) to the preferred Object-Subject-Verb (OSV) presentation in order to adhere to American Sign Language grammar to maintain semantic integrity.

The Video Processing component then accepts processed tokens. The WLASL dataset is searched to cross-reference each token to a Sign Language Video Database. The word is then matched to its corresponding gloss metadata, e.g., source, frame ranges, gloss label, and video ID, by means of a precompiled JSON dictionary. In order to maintain visual consistency, clips performed by Signer ID 11 only are selected. The following fallback mechanism is used when there is no token match available:

- Lemmatization and stemming to convert the token to its root form to allow broader searching
- WordNet-based synonym replacement to identify semantically equivalent glosses

The Video Stitching process begins after the following acquisition of all relevant clips. The clips are stitched in sequence following clipping to frame boundaries specified by FFmpeg and invoked by Python's subprocess module. Finally, the tool produces a smooth and cohesive video stream by ensuring constant frame rates, transitions, and uniform resolution.

Finally, the GUI displays to the user the System Response, an entirely synthesized ASL video. To give an end-to-end and readable sign language translation experience, the framework ensures that any module contributes to semantic accuracy, visual flow, and real-time response.

3. Results and Discussion

3.1. User Interface Navigation

SignSpeak, is designed as a communication tool to bridge the gap between speakers and sign language users by translating spoken language into sign language videos. It features a user-friendly interface that integrates speech recognition capabilities.

Figure 2 SignSpeak GUI

Figure 2 represents the main homepage of SignSpeak. Here, there are two methods of providing input, first being by recording audio and the second by entering text manually into the text box to process text. For audio input, the “Record Audio” option upon clicking will record the speech till the speech is completed. The “Process Text” option exists to feed the speech text directly to the core system. The “Key Words” section lists the keywords identified in the text and “Sign Video Output” displays the stitched video output for the sentence.

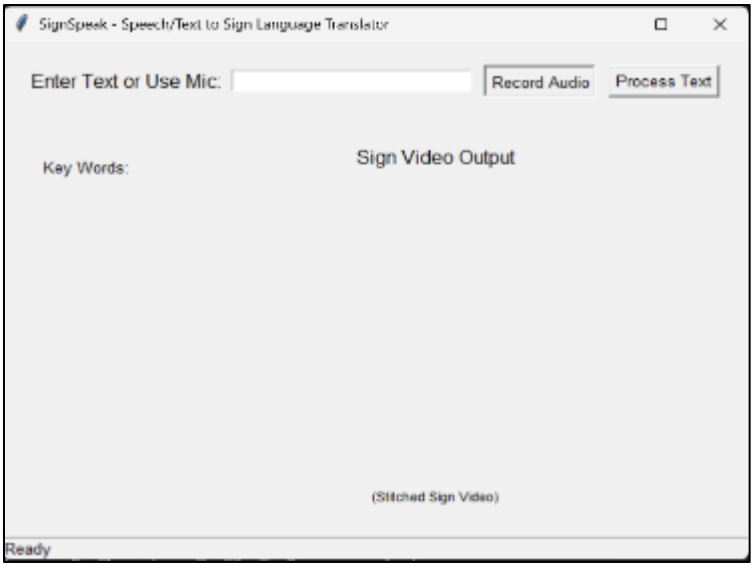

Figure 3 SignSpeak Recording Audio

Figure 4 SignSpeak playing the stitched video for the keywords identified

Users can interact with the application by speaking their queries, which the system then processes to generate sign language output as shown in **Figure 3** and **4**. Leveraging Whisper for speech-to-text conversion and BERT for semantic reordering, the system analyzes spoken input to determine the appropriate sequence of sign language videos. Specifically, the speech recognition functionality enables voice interaction, enhancing accessibility for users who may find typing inconvenient. SignSpeak then stitches the corresponding sign language video clips using FFmpeg to ensure that information is effectively communicated in a visual format, promoting inclusivity.

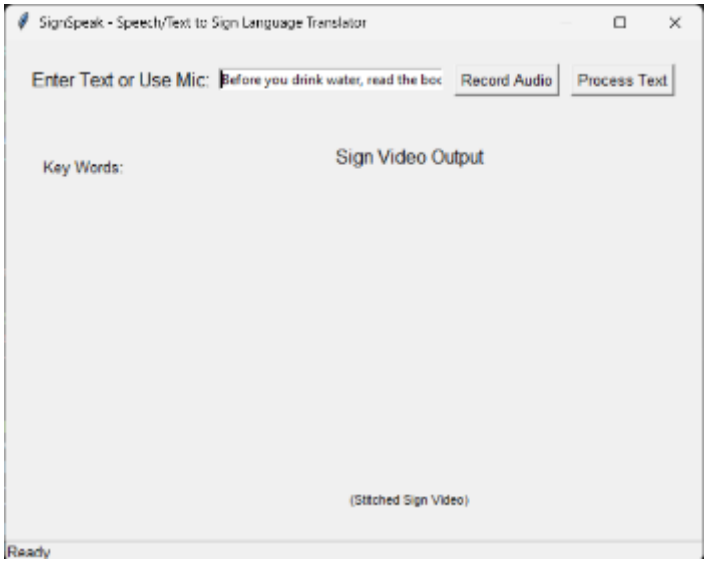

Figure 5 Entering text manually for process

In addition, text can be directly provided as input to the application for any manual edits that need to be made by the user as shown in **Figure 5**. This allows the user to make changes to the speech as per their requirement. This approach also assists the User if in case they do not have access to a microphone making it serve for dynamic purposes

Figure 6 Sign Language Video Output for direct text processing

The sign language video output for direct text processing is represented in **Figure 6**. The identified keywords are listed on the left followed by the stitched video comprising of the key words sign glosses being shown on the right.

3.2. Performance Metrics

3.2.1. ASR WER (Automatic Speech Recognition Word Error Rate):

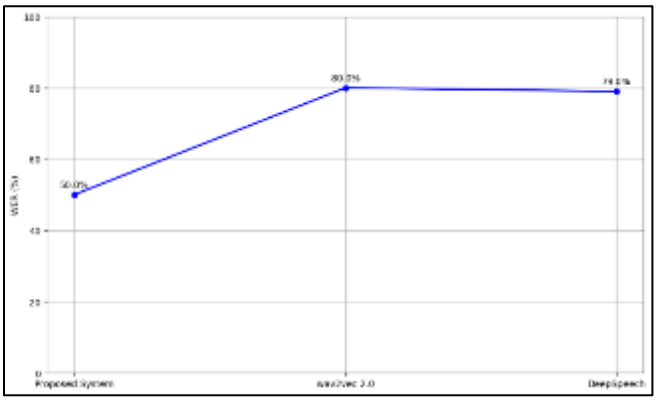

Figure 7 ASR WER Comparison

Word Error Rate is one of the standard metrics for the assessment of automated speech recognition (ASR) systems. It is the ratio of mis-predicted to predicted words compared to the original transcript. WER is the total number of substitutions, deletions, and insertions needed to align the predicted transcript with the ground truth, and is divided by the number of word tokens in the ground truth. Lower WER denotes improved ASR. It also shows how well the system can transcribe spoken input, specifically when the input is in the presence of different conditions of noise.

3.2.2. Gloss Coverage

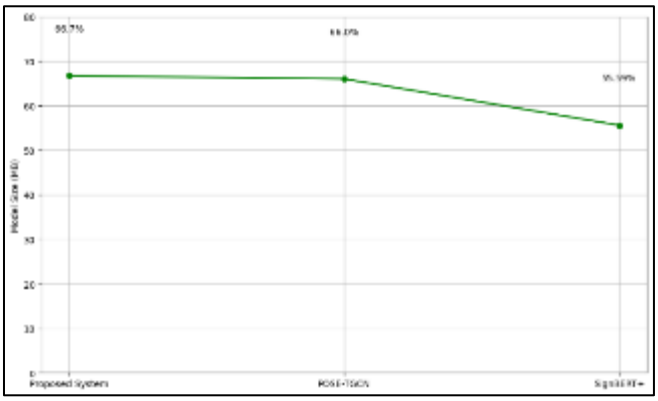

Figure 8 Gloss Coverage

Gloss coverage is the rate of spoken material which can be effectively associated with the equivalent glosses reduced sign language descriptions that are employed to produce sign outputs. It describes how effectively the system is able to translate spoken word into identified sign language units from an established gloss dictionary. High gloss coverage denotes that the system is able to translate a greater number of vocabularies. Restricted gloss coverage results in incomplete or incorrect sign interpretation, which makes gloss coverage an important metric in the evaluation of completeness and semantic accuracy during translation.

3.2.3. Video Synthesis Time:

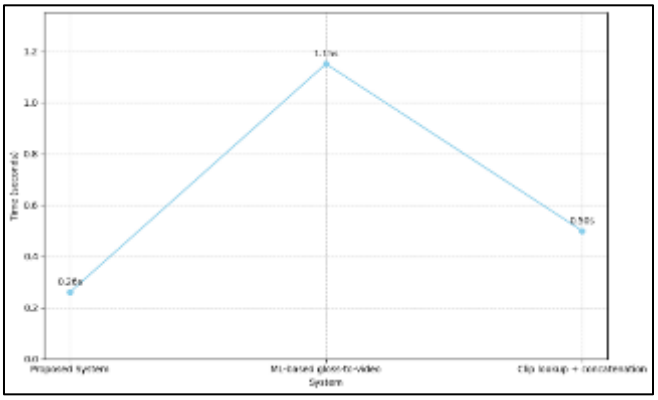

Figure 9 Video Synthesis time per sentence comparison

Video synthesis time is the average amount of time the system takes to generate and render the equivalent sign language video to an input sentence. This is usually expressed in units of seconds per sentence and indicates the system's response and real-time viability. Lower synthesis times provide directly proportionally better user experiences and are particularly beneficial for interactive or assistive applications. It is an important metric to use in assessing the system's real-time viability.

3.2.4. Comparison of Existing System with Proposed System

Table 1 Performance Metrics comparison

System	ASR WER	Gloss Coverage	Video Synthesis Time
Proposed System	50.0%	66.7%	0.26s
Existing System	80.0%	66.0%	1.15s
	79.0%	55.59%	0.50s

Existing System:

- ASR WER: wav2vec 2.0 Base = 80%, DeepSpeech Base = 79%.
- Gloss Coverage: Pose-TGCN = 66%, SignBERT+ = 55.59%
- Video Synthesis Time: ML-based gloss-to-video = 1.15s, Clip lookup + concatenation = 0.50s.

The proposed system for sign language translation exhibits a competitive advantage over other existing systems across major performance metrics. The ASR performance is presented in **Figure 7**, with the proposed system using the Whisper Base model delivering a 50.0% WER. Although wav2vec 2.0 Base records a WER of 80.0% on pre-trained data, the proposed system enjoys better computational and deployment efficiencies, and it performs significantly better than the DeepSpeech Base (79.0% WER). The gloss coverage is detailed in **Figure 8**, with the proposed system achieving 66.7% using an augmented model of BERT and signer-specific training. This is better than SignBERT+ (55.59%) and tied for Pose-TGCN (66%). Video synthesis time, an important metric for real-time usage, is shown in **Figure 9**. It has an average synthesis time of about 0.26 seconds per sentence, significantly better than the 1.15s for the ML-based gloss-to-video model, and 0.50s for the Clip lookup + concatenation approach. This comparison is summarized for the metrics together in **Table 1**, further demonstrating the better achievement for the proposed system.

4. Conclusion

The increasing global population of the Deaf and Hard of Hearing, along with the scarcity of certified sign language interpreters, raises the demand for effective and convenient Speech-to-Sign Language Translation Systems. The solution has come in the form of a real-time modular system built on Whisper for speech-to-text recognition, BERT for reordering the sentences syntactically, and FFmpeg for video synthesis, using a gloss-based dictionary to accurately generate the ASL video. This design ensures smooth output through semantic parsing, lemmatization, and synonym matching, even

under conditions of shallow gloss coverage. Visual consistency was maintained by using a single signer (ID 11), improving fluidity and usability, especially for non-technical users accessing the system via a simplified GUI.

In the future, future improvements are set to enhance performance as well as scalability. Upgrades that are being proposed consist of dynamic gloss expansion with the use of contextual models such as GPT, incorporation of non-manual components of the sign (e.g., facial expressions), and replacement of the existing GUI using stronger frameworks such as PyQt5. In addition, fallback into 3D avatars such as SignAvatar or DeepMotion can overcome limitations on gloss coverage. Although the existing system shows considerable improvement compared to current alternatives, these improvements will further facilitate the creation of an improved, flexible, and inclusive real-time Speech-to-Sign Language translator.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Ghale, A., Janaki, K., & Verma, D. C. (2022). Instant transcription and translation tool using OpenAI's Whisper ASR model. International Journal of Science and Research (IJSR), 11(12). Retrieved from [https://www.ijsr.net/archive/v11i12/SR221203164929.pdf?utm]

[2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)10 (pp. 4171-4186).

[3] Rajasri, K., Gayathri, D., Ilanthirayan, B., & Sundra, A. (2018). Real time panoramic video in OpenCV using image stitching techniques. International Journal of Computer Sciences and Engineering, 6(3), 184–188.

[4] Li, D., Opazo, C. R., Yu, X., & Li, H. (2019). Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison. ArXiv. https://arxiv.org/abs/1910.11006

[5] Sainburg, M. S. (2020). JiWER: The 'Just Another Word Error Rate' package. [Online]. Available: https://github.com/jitsi/jiwer

[6] Kuhn, K., Kersken, V., Reuter, B., Egger, N., & Zimmermann, G. (2023). Measuring the accuracy of automatic speech recognition solutions. ACM Transactions on Accessible Computing, 16(4), Article 25. https://doi.org/10.1145/3636513

[7] Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in neural information7 processing systems, 33, 12449-12460.8

[8] Shi, B., Brentari, D., Shakhnarovich, G., & Livescu, K. (2024). Open-domain sign language translation learned from online video. 2022. arXiv preprint arXiv:2205.12870.

[9] Moryossef, A. (2023). sign. mt: Real-time multilingual sign language translation application. arXiv preprint arXiv:2310.05064.

[10] Najib, F. M. (2024). Sign language interpretation using machine learning and artificial intelligence. *Neural Computing and Applications*, 1-17.

[11] Shahin, N., & Ismail, L. (2024) [3]. From rule-based models to deep learning transformers architectures for natural language processing and sign language translation systems: survey, taxonomy and performance evaluation. *Artificial Intelligence Review*, *57*(10), 271.[1]

[12] Strobel, G., Schoormann, T., Banh, L., & Möller, F. (2023). Artificial intelligence for sign language translation–A design science research study.[2] *Communications of the association for information systems*,[3] *53*(1), 42-64.

[13] Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023, July). Robust speech recognition via large-scale weak supervision. In *International conference on machine learning* (pp. 28492-28518).[4] PMLR.

[14] Khaskheli, A. H., Mirani, S. H., & Arain, A. An Android App for Deaf & Dumb People (Sign language Recognition–Smart Talk App). (February 2022).

[15] Papastratis, I., Chatzikonstantinou, C., Konstantinidis, D., Dimitropoulos, K., & Daras, P. (2021). Artificial intelligence technologies for sign language.5 Sensors, 21(17), 5843.6

[16] Yi, J., Tao, J., Bai, Y., Tian, Z., & Fan, C. (2020). Adversarial Transfer Learning for Punctuation Restoration. arXiv. https://arxiv.org/abs/2004.002481.

[17] Mishra, S. K., Sinha, S., Sinha, S., & Bilgaiyan, S. (2019). Recognition of hand gestures and conversion of voice for betterment of deaf and mute people.11 In Advances in Computing and Data Sciences: Third International Conference, ICACDS 2019, Ghaziabad, India, April 12–13, 2019, Revised Selected Papers, Part II12 3 (pp. 46-57). Springer Singapore.

[18] Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., ... & Ng, A. Y. (2014). Deepspeech: Scaling up end-to-end speech recognition. arXiv preprint arXiv:1412.5567.13

[19] Mk NB (2018) Conversion of sign language into text. Int J Appl Eng Res 13(9):7154–7161

[20] Manikandan K, Patidar A, Walia P, Roy AB (2018) Hand gesture detection and conversion to speech and text. arXiv preprint arXiv:1811.11997.

[21] Shovkoplias G, Tkachenko M, Asadulaev A, Alekseeva O, Dobrenko N, Kazantsev D, Vatian A, Shalyto A, Gusarova N. Support for communication with deaf and dumb patients via few shot machine learning. In International conferences ICT, society, and human beings (pp. 216-220).

[22] Tolentino LKS, Juan RS, Thio-ac AC, Pamahoy MAB, Forteza JRR, Garcia XJO (2019) Static sign language recognition using deep learning. Int J Mach Learn Comput 9(6):821–827

[23] World Health Organization. (2024, February 26). Deafness and hearing loss (Fact sheet). Retrieved May 4, 2025, from https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

[24] Baevski, A., Zhou, H., Mohamed, A., & Auli, M. (2020). Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. ArXiv. https://arxiv.org/abs/2006.11477

[25] Kumalija, E., & Nakamoto, Y. (2022). Performance evaluation of automatic speech recognition systems on integrated noise-network distorted speech. Frontiers in Signal Processing, 2, 999457.

[26] Li, D., Opazo, C. R., Yu, X., & Li, H. (2019). Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison. *ArXiv*. https://arxiv.org/abs/1910.11006

[27] Zhao, W., Hu, H., Zhou, W., Shi, J., & Li, H. (2023). BEST: BERT Pre-Training for Sign Language Recognition with Coupling Tokenization. *ArXiv*. https://arxiv.org/abs/2302.05075