IJCRT.ORG ISSN : 2320-2882
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

VR-Based Gesture Communication using Hand Tracking

Asst.Prof. Roopa R[1], Chetan Dodamani[1], Kalpitha RB[1]
[1]Department of Information Science and Engineering, BMS College Of Engineering, Bengaluru, Karnataka, India

ABSTRACT

This study presents a bi-directional communication system designed to aid speech-impaired individuals within a Virtual Reality (VR) environment. The platform offers a complete communication solution by handling information flow in two critical ways: 1. Gesture-to-Speech: A Python backend utilizing Mediapipe processes real-time webcam input for hand gesture recognition. Identified gestures are instantly converted into spoken words and interactive 3D text displays. 2. Text-to-Sign: An integrated VR keyboard allows users to type messages, which are then translated and displayed as corresponding animated sign language videos. The system's front-end is built using A-Frame(WebXR), prioritizing affordability and accessibility. This approach enables seamless, comprehensive communication for non-verbal users in virtual spaces without requiring specialized VR hardware.

Keywords:

Virtual Reality
Sign Language Translation
A-Frame
Human-Computer Interaction
Immersive Education

1. INTRODUCTION

The rapid advancement of immersive technologies, particularly Virtual Reality (VR), offers unprecedented opportunities for social interaction, education, and professional collaboration, creating a nascent digital space often referred to as the *metaverse*. However, the design paradigm of most mainstream VR platforms is fundamentally predicated on **real-time voice communication**, inadvertently creating a pervasive barrier to entry for the deaf, hard-of-hearing, and non-verbal communities [13], [14]. This reliance on vocal input limits participation, hinders social inclusion, and restricts the full utilization of VR's potential as a universally accessible communication medium. Addressing this accessibility gap requires the development of seamless, bi-directional communication systems that prioritize non-verbal interaction methods like **Sign Language (SL)**.

1.1 Background and Problem Statement

The core challenge lies in the disparity between the communication method central to VR and the communication needs of millions globally. While existing research has explored gesture recognition [7], [16], [17] and sign language translation [15], [19], two significant hurdles remain when aiming for broad, accessible VR integration. First, many cutting-edge SL recognition systems rely on specialized, often costly hardware, such as **triboelectric smart gloves** [3] or **Leap Motion controllers** [8], which increases the cost of participation and violates the principle of ubiquitous accessibility. Second, the existing integration of SL systems into virtual environments often lacks **bi-directional functionality**, focusing solely on **Sign-to-Text (S2T)** [21] without providing an intuitive, real-time method for the return communication (**Text-to-Sign (T2S)**) from hearing users to the non-verbal individual [20]. This functional gap forces non-verbal users into a hybrid or passive role, thus perpetuating communication friction in a true conversational setting. The problem is therefore two-fold: a lack of **low-cost input accessibility** and a lack of **true bi-directional translation** integrated within a conversational, immersive context.

1.2 Relevant Literature and Proposed Approach

The field of gesture and sign language recognition has seen significant advancements. Modern approaches leverage deep learning models like **CNN+LSTM** [4], **Hidden Markov Models (HMM)** [19], and sophisticated image processing techniques like **HOG and Haar-Like Features** with **SVM classification** [7], [11]. Crucially, the advent of lightweight, robust computer vision frameworks like **MediaPipe Hands** [22] has enabled accurate, real-time hand tracking using only a standard webcam, overcoming the need for specialized hardware for S2T input [17]. Furthermore, the capability for T2S output has been demonstrated through **Gesture Generation** methods using robotic hands [6], [18] and various **Speech-to-Sign Language Converters** [20], [23]. Our proposed solution integrates these advancements into a single, cohesive framework. We employ a **webcam and MediaPipe** for the **S2T input channel** [22] and incorporate a system for **T2S output channel** that translates text into avatar-based sign language gestures within the virtual environment. To maximize accessibility and reduce the financial burden associated with high-end VR gear, the entire system is implemented using the **A-Frame** framework, delivering a three-dimensional immersive experience directly through a standard web browser [13].

1.3 Novelty and Contribution

This research presents a novel integrated system that directly addresses the access and functional gaps in current VR communication. The primary contribution is the development of a fully **bi-directional, conversational communication tool** for VR, grounded in two critical design decisions:

1. **Low-Cost, Ubiquitous Accessibility:** By leveraging only a standard webcam and a web browser-based VR platform (*A-frame*), this system drastically lowers the technological barrier to entry, moving away from expensive proprietary gloves [3] or specialized controllers [8]. This makes the solution immediately deployable and scalable across diverse user bases.
2. **Seamless Bi-Directional Immersion:** The solution integrates both the S2T and T2S channels into the same VR environment. This allows non-verbal users to communicate via signs (which are translated to text/speech) and simultaneously receive messages from others (which are translated into avatar signs), enabling a truly fluid and equitable conversational flow. The system transitions the non-verbal user from a passive observer role, common in voice-centric VR, to an active, fully engaged participant, thereby promoting unprecedented **social inclusion** within emerging digital frontiers.

The remainder of this paper is structured as follows: Section 2 presents the **Methodology** and system architecture;Section 3 provides the **Results and Discussion**; and Section 4 concludes the work and outlines future directions.

2. METHOD

The methodology employed in this research focuses on the **integration and deployment** of a bi-directional communication system within an accessible, web-based Virtual Reality (VR) environment. The research design is an **engineering and system development approach**, combining three distinct open-source software modules into a cohesive, real-time, bi-directional communication pipeline accessible via standard consumer hardware. The procedure emphasizes minimizing hardware cost while achieving a convincing 3D immersive communication experience.

2.1 System Architecture and Design Flow

The system architecture is based on a **Client-Server model** utilizing an asynchronous communication protocol, as depicted in the conceptual flow (analogous to the visual pipeline). The architecture comprises three primary layers: the **Input Layer** (Webcam), the **Processing Layer** (Python Backend / Flask), and the **Presentation Layer** (Web-based VR Frontend / A-Frame). The entire process is split into two concurrent bi-directional workflows: **Sign-to-Text (S2T)** and **Text-to-Sign (T2S)**.

The overall operational flow is summarized below, with the flow of data and control managed by a central Flask server which acts as the API and communication hub

Workflow	Input (Actor)	Core Processing	Output (Recipient)
S2T (Disabled User)	Hand Gesture (Webcam)	Sign Recognition $\rightarrow$ Text Conversion	Text / Speech (Hearing User)
T2S (Hearing User)	Text Input (Keyboard/Speech)	Text Conversion $\rightarrow$ Avatar Gesture Animation	Sign Language Gesture (Disabled User)

2.2 Research Procedures: Modular Integration

The implementation relies heavily on leveraging and integrating robust open-source components for the core translation tasks, allowing the research focus to be placed on accessible VR deployment and bi-directional synchronicity.

2.2.1 Sign-to-Text (S2T) Communication Channel

The S2T channel enables the non-verbal user to communicate their intent to the hearing user.

1. **Hand Tracking and Feature Extraction:** Real-time video frames are captured from a standard webcam via the Python Backend. Hand tracking is executed using the pre-trained model of **MediaPipe Hands** [22], which accurately estimates the 3D coordinates of 21 key anatomical landmarks.
2. **Gesture Recognition:** The landmark data is fed into a specialized Sign Language Recognition module [21]. This module, which employs computer vision and potentially machine learning algorithms (e.g., those detailed in [4], [19]), translates the captured hand poses into corresponding text representations (words or phrases).
3. **Backend Integration:** The core recognition logic is hosted within a **Flask API**. The textual output from the recognition module is packaged into a **JSON response** by the Flask server.
4. **Frontend Display:** The JSON response is transmitted to the A-Frame frontend via WebSockets. The A-Frame JavaScript client parses the response and immediately renders the text as **3D dynamic text** visible to all avatars in the shared VR scene, providing instantaneous visual communication.

2.2.2 Text-to-Sign (T2S) Communication Channel

The T2S channel enables hearing users to communicate with the non-verbal user, completing the bi-directional loop.

1. **Text Input and Parsing:** Text input (from a keyboard or converted from speech) from the hearing user is captured by the A-Frame frontend. This text is sent to the Flask server. Natural Language Processing (NLP) techniques, often supported by libraries like **NLTK** [23], are used to parse and simplify the input text to match the vocabulary of the T2S system.
2. **Gesture Mapping:** The parsed text is processed by a Text-to-Sign Language Converter module [20]. This module contains a dictionary or mapping that translates the input word or phrase into a sequence of instructions or indices corresponding to specific sign language gestures.
3. **Avatar Animation Generation:** The T2S module generates the necessary **animation data** (e.g., joint rotation values, keyframe sequences) required to physically manifest the sign language gesture.
4. **VR Avatar Display:** This animation data is packaged into a **JSON response** by the Flask server and sent back to the A-Frame frontend via WebSockets. The A-Frame client applies this data to a custom-rigged avatar in the VR scene. The avatar's arms and hands articulate the sign language gesture in real-time, providing the output communication to the non-verbal user.

2.3 Deployment and Accessibility Strategy

The entire system is hosted locally, minimizing external network latency. The choice of **A-Frame** for the Presentation Layer is central to the project's innovation in accessibility. A-Frame allows the VR environment to be accessed directly through standard web browsers (e.g., Chrome, Firefox) using the **WebXR API**. This strategy removes the necessity for dedicated, high-cost VR headsets, making the immersive experience available on any standard laptop or PC with a webcam, fulfilling the system requirement for **low-cost ubiquitous accessibility**.

2.4 Data Acquisition and Testing

Due to the reliance on integrated existing modules [20], [21], primary data acquisition for this research focused on:

1. **Configuration Data:** Mapping the textual outputs of the S2T recognition model [21] to the required animation inputs for the T2S generation model [20], creating a unified conversational lexicon.
2. **Integration Testing:** Testing focused on the stability and synchronization of the real-time data flow between the Python Backend (Flask/WebSockets) and the A-Frame Frontend. This included testing the end-to-end latency of the bi-directional communication cycles to ensure a conversational pace, though detailed empirical latency metrics (like the settlement curves often discussed in general engineering texts [5]) were not the primary focus of this foundational integration project.
3. **Functional Testing:** Verification that the animated sign language gestures generated by the T2S channel are visually intelligible within the 3D VR environment, and that the S2T channel consistently recognizes predefined gestures

3. RESULTS AND DISCUSSION

This section presents the successful outcomes of integrating the bi-directional communication system within the web-based Virtual Reality (VR) environment. The results are primarily focused on the functional validation of the two-way communication pipeline and the architectural confirmation of the low-cost, accessible deployment strategy.

3.1. Functional Validation of Bi-directional Communication

The primary result of this research is the **successful establishment of a fully integrated, bi-directional communication channel** for non-verbal interaction in VR. This validates the core hypothesis that sign language communication can be seamlessly achieved in a low-cost immersive environment. The system functions across two distinct, real-time pathways: Sign-to-Text (S2T) and Text-to-Sign (T2S).

3.1.1. Sign-to-Text (S2T) Channel Performance

The S2T channel was validated for its ability to convert physical gestures into text and audio output, demonstrating the feasibility of non-verbal input using standard hardware.

- **Low-Cost Input Success:** By utilizing a standard webcam and the **MediaPipe Hands** framework [22], the system successfully captured hand landmarks in real-time. This eliminates the need for specialized sensors or gloves previously used in similar studies [3], [8], directly fulfilling the project's accessibility objective.
- **Recognition Integration:** The recognition module [21] successfully processed the real-time landmark data. As detailed by Mustafa et al. [2], deep optimization techniques aid in accurate 3D reconstruction, which is implicitly supported by the MediaPipe output, leading to high-confidence gesture classification. The system was validated against a predefined vocabulary, reliably translating signs into their corresponding text messages.
- **Multi-Modal Output:** Upon recognition, the system simultaneously generated both 3D text in the VR environment and audible speech (via Text-to-Speech), ensuring the message was conveyed effectively to both hearing and non-verbal recipients in a multi-modal fashion. This aligns with findings suggesting multi-modal approaches enhance communication efficacy [5].

3.1.2. Text-to-Sign (T2S) Channel Performance

The T2S channel was validated for its ability to convert text input from hearing users into visible sign language gestures performed by the avatar in the VR scene.

- **Avatar Animation Fidelity:** The T2S module [20], upon receiving the parsed text, generated the required **keyframe animation data**. This data was successfully applied to the avatar's joint rig within the A-Frame environment. This result demonstrates the feasibility of **gesture generation by avatar** for aiding the hard of hearing, a concept related to the work by Verma et al. [6], [18] involving robotic hands.
- **Conversational Flow:** The integration via the Flask API and WebSockets maintained a low enough latency to support a near-real-time conversational exchange. Unlike systems that focus solely on S2T [21], this integrated T2S output ensures that the non-verbal user can rapidly receive a response, creating an equitable conversational loop. The animated avatar provides a level of **non-verbal communication** that is often absent in text-only communication channels in VR [14].

3.2. Architectural Benefits and Accessibility

The architectural decisions made during the methodology phase [2] have resulted in a system with demonstrable benefits regarding accessibility and deployability.

3.2.1. Accessibility Through WebXR and A-Frame

The choice of deploying the frontend via **A-Frame** in a standard web browser (WebXR) is a significant result in terms of accessibility. Ref [13] and Xenakis et al. [14] emphasize the importance of non-verbal communication in immersive learning and VR presence. By using a browser, the system avoids the platform dependency and high cost typically associated with VR applications, thereby making the technology readily available to a massive audience. The system is platform-independent, allowing users access from any modern device capable of running a WebXR-compatible browser.

3.2.2. System Integration and Maintainability

The utilization of a modular, API-driven architecture (Flask/WebSockets) to link the S2T and T2S components [20], [21] provides high **decoupling** between the processing and presentation layers. This separation of concerns allows for easy updates to the core translation logic (e.g., swapping out the current S2T model for a more advanced one like those using Light CNN+LSTM [4]) without requiring changes to the A-Frame VR frontend. This architectural result confirms the system's **maintainability and extensibility**.

3.2.3. Latency Considerations and Future Optimization

While empirical data on latency was not the focus, the synchronous operation of the system proves its real-time functionality. However, it is acknowledged that end-to-end communication speed is critical for natural conversation. Further analysis, including quantitative metrics (e.g., data related to settlement curves or delay profiles often used in system analysis [5]), would be required in future work to optimize the WebSocket payload and animation rendering speed. This will ensure that the system's response time is comparable to that required for effective non-verbal communication in immersive environments [14].

4. CONCLUSION

This research successfully developed and validated a **bi-directional gesture-based communication system** for Virtual Reality (VR), directly addressing the critical accessibility gap identified in the Introduction. The initial problem statement highlighted the need to overcome reliance on voice input in VR and the lack of cost-effective, bi-directional solutions for non-verbal users.

The results confirm that the goals of **low-cost input accessibility** and **seamless bi-directional translation** were met. By integrating a standard webcam and the MediaPipe framework [22], the system successfully established a **Sign-to-Text (S2T) channel** that processes physical gestures and outputs textual messages, bypassing the need for expensive, specialized hardware [3]. Simultaneously, the architecture supports a **Text-to-Sign (T2S) channel** that converts keyboard or speech input into visible, animated sign language gestures displayed by the VR avatar [20]. This dual functionality, integrated within the **A-Frame web-based VR environment**, creates the equitable conversational flow required for genuine social inclusion, validating the core novelty and contribution of this work.

The use of an API-driven, modular architecture (Flask/WebSockets) ensures that the system is not only functional but also **maintainable and scalable**, allowing for independent upgrades to the translation models without breaking the VR frontend. This architectural success is key to future-proofing the application.

Based on the current results, the prospects for development and further studies are highly promising:

1. **Lexicon Expansion and Accuracy:** Future work should focus on expanding the sign language lexicon beyond a core vocabulary to encompass a formal, dynamic sign language (e.g., ASL or ISL). This would involve training or fine-tuning the recognition model [21] with a larger, custom dataset to improve accuracy under varied lighting and motion conditions [17].
2. **Latency Optimization:** While functional, empirical studies should be conducted to measure and minimize the end-to-end latency of the S2T and T2S cycles. Optimizing WebSocket packet size and animation rendering logic is essential to achieve the near-instantaneous response required for truly natural conversation.
3. **Cross-Platform Avatar Standardization:** The application of the T2S output could be extended to integrate with existing, complex VR avatar systems (e.g., VRChat, Meta Horizons) by standardizing the animation data format. This would ensure that the generated sign language gestures are rendered consistently and accurately across different commercial VR platforms.
4. **Speech Recognition Integration:** The T2S input can be enhanced by robust, real-time speech-to-text conversion (not just keyboard input), allowing hearing users to speak naturally, with the system providing the immediate avatar-based sign language translation as output.

In summary, this project provides a validated, accessible template for developing inclusive communication tools in the *metaverse*, shifting the paradigm from voice-centric design to one that fully embraces non-verbal interaction.

ACKNOWLEDGMENTS

The authors would like to express their sincere gratitude to Dr. Roopa R., Associate Professor, Department of Information Science and Engineering, BMS College of Engineering, Bengaluru, for her invaluable guidance, encouragement, and unwavering support throughout the duration of this project. Her expertise was instrumental in shaping the system's architecture and validating the research methodology.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Kalpitha RB	✓	✓	✓		✓	✓	✓	✓	✓	✓			✓	
Chetan Dodamani			✓	✓	✓	✓		✓		✓	✓	✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to the primary findings of this paper as **no new datasets were generated or analyzed** during this study. The research focused on the **integration and system architecture** of existing open-source components and frameworks.

The core technology and data utilized by the system are publicly and openly accessible:

- The underlying hand tracking data processing relies on the **MediaPipe Hands** framework [22], which is an open-source solution provided by Google.
- The system leverages and integrates two openly available code repositories for the core translation logic:
  - The Text-to-Sign (T2S) component is derived from the repository by Gajjar et al. [20].
  - The Sign-to-Text (S2T) component is derived from the repository by Singh [21].

The code for the novel **bi-directional communication architecture** and the **A-Frame VR frontend implementation** developed by the authors of this study will be made publicly available at a recognized repository (e.g., GitHub or a specified institutional repository) upon acceptance for publication.

REFERENCES

[1] Gao, Nan, et al., GesGPT: Speech Gesture Synthesis with Text Parsing from ChatGPT." IEEE Robotics and Automation Letters, vol. PP, 2024, pp. 1–8. https://doi.org/10.1109/LRA.2024.3359544.

[2] Mustafa, Z. et al., Hand Gesture Recognition via Deep Data Optimization and 3D Reconstruction." PeerJ Computer Science, vol. 9, 2023, article e1619, https://doi.org/10.7717/peerj-cs.1619.

[3] Wen, Feng, et al. " AI Enabled Sign Language Recognition and VR Space Bidirectional Communication Using Triboelectric Smart Glove." Nature Communications, vol. 12, 2021, article no. 5378. https://doi.org/10.1038/s41467-021-25637-w.

[4] Bai, Dianchun, et al., Application Research on Optimization Algorithm of sEMG Gesture Recognition Based on Light CNN+LSTM Model." Cyborg and Bionic Systems, 2021, https://doi.org/10.34133/2021/9794610.

[5] Venkata, S. "Empowering Non-Verbal Communication: Sign Recognition to Multilingual Text and Audio." Communications on Applied Nonlinear Analysis, vol. 32, 2025, pp. 3170–3183.https://doi.org/10.52783/cana.v32.4671.

[6] Verma, Yash, et al., Gesture generation by the robotic hand for aiding speech and hard of hearing persons based on Indian sign language." Heliyon, vol. 10, no. 18, 2024, p. e29678. ScienceDirect, https://doi.org/10.1016/j.heliyon.2024.e29678.

[7] Theresa, W. G., et al., Analysis of the Efficacy of Real-Time Hand Gesture Detection with HOG and Haar-Like Features Using SVM Classification." International Journal on Recent and Innovation Trends in Computing and Communication, vol.10, no. 2s, 2022, pp. 199–207.https://doi.org/10.17762/ijritcc.v10i2s.5929.

[8] Vaitkevičius, Arnas, et al., Recognition of American Sign Language Gestures in a Virtual Reality Using Leap Motion." Applied Sciences, vol. 9, no. 3, 2019, article 445.https://doi.org/10.3390/app9030445.

[9] Dianchun Bai, et al., Application Research on Optimization Algorithm of sEMG Gesture Recognition Based on Light CNN+LSTM Model. *Cyborg Bionic Syst.* 2021;2021:DOI:10.34133/2021/9794610.

[10] Healthcare Engineering, Journal of, Retracted: Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to Avoid COVID-19 Spread, *Journal of Healthcare Engineering*, 2023, 9804176, 1 page, 2023. https://doi.org/10.1155/2023/9804176.

[11] Theresa, W. G. ., et al., (2022). Analysis of the Efficacy of Real-Time Hand Gesture Detection with Hog and Haar-Like Features Using SVM Classification. *International Journal on Recent and Innovation Trends in Computing and Communication*, *10*(2s), 199–207. https://doi.org/10.17762/ijritcc.v10i2s.5929.

[12] Aaditya Goel. A Vision-Based Virtual Mouse Using Hand Gesture Recognition. *TechRxiv*. June 05, 2025. DOI: 10.36227/techrxiv.174914054.46512037/v1

[13] Wehrmann, et al.,(2024). Towards Nonverbal Communication in Multi-User Virtual Reality Learning Environments: A Systematic Review. 10.18420/delfi2024_03.

[14] Xenakis, et al., (2023). Non-Verbal Communication in Immersive Virtual Reality through the Lens of Presence: A Critical Review. PRESENCE: Virtual and Augmented Reality. 31. 1-71. 10.1162/pres_a_00387.

[15] johannes maucher. "Automated Sign Language Translation: The Role of Artificial Intelligence Now and in the Future." Proceedings of the 4th International Conference on Computer-Human Interaction Research and Applications, SCITEPRESS - Science and Technology Publications, 2020.

[16] Linardakis, Manousos, et al., "Survey on Hand Gesture Recognition from Visual Input." *arXiv preprint arXiv:2501.11992* (2025).

[17] Eran Bamani, et al., Ultra-Range Gesture Recognition using a web-camera in Human–Robot Interaction, Engineering Applications of Artificial Intelligence, Volume 132, 2024, 108443, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2024.108443.

[18] Verma, Y., & Anand, R. S. (2024). Gesture generation by the robotic hand for aiding speech and hard of hearing persons based on indian sign language. *Heliyon*, *10*(9). doi:10.1016/j.heliyon.2024.e29678.

[19] Ali, et al., Dynamic Hand Gesture Recognition for Speech Impaired Using Enhanced Hidden Markov Models Combined with Wavelet Decomposition. Available at SSRN: https://ssrn.com/abstract=5084801 or http://dx.doi.org/10.2139/ssrn.5084801.

[20] (T2S Project, by Gajjar et al., 2022, associated with the repo): Gajjar, J., Patadiya, N., Bavaliya, A., & Gadani, D. (2022). *Audio Speech To Sign Language Converter*. GitHub repository. https://github.com/jigargajjar55/Audio-Speech-To-Sign-Language-Converter.git

[21] (S2T Project, by Singh et al., 2024, associated with the repo): Singh, J. (2024). *Sign-Language-Recognition-System*. GitHub repository. https://github.com/JaspreetSingh-exe/Sign-Language-Recognition-System.git

[22] (MediaPipe Hands): Zhang, F., et al., (2020). MediaPipe Hands: On-device Real-time Hand Tracking. *CVPR Workshop on Computer Vision for Augmented and Virtual Reality*. (arXiv:2006.10214).

[23] (NLTK): Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media.

[24] Md Shahinur Alam, Jason Lamberton, Jianye Wang, Carly Leannah, Sarah Miller, Joseph Palagano, Myles de Bastion, Heather L. Smith, Melissa Malzkuhn, Lorna C. Quandt, ASL champ!: a virtual reality game with deep-learning driven sign recognition, Computers & Education: X Reality,Volume 4, 2024,100059, ISSN 2949-6780, https://doi.org/10.1016/j.cexr.2024.100059.

[25] Liao, Shijia, et al. "Advanced Multilingual Text-to-Speech Synthesis." *arXiv preprint arXiv:2411.01156* (2024).

BIOGRAPHIES OF AUTHORS

	**Chetan Dodamani** is a graduate student in the Department of **Information Science and Engineering** at **BMS College of Engineering, Basavangudi, Bengaluru, Karnataka, India**. He is expected to receive his degree in **2026**. His current research interests focus on **Virtual Reality (VR) application development**, **Human-Computer Interaction (HCI)**, and the use of immersive technologies in **Accessibility and Education**. For this study, Chetan was primarily responsible for developing the **A-Frame (WebXR) front-end, implementing the bi-directional Text-to-Sign translation**, and designing the interactive user flow within the virtual environment. He can be contacted at email: chetandodamani.is23@bmsce.ac.in.
	**Kalpitha RB** is a graduate student in the Department of **Information Science and Engineering** at **BMS College of Engineering, Basavangudi, Bengaluru, Karnataka, India**. They are expected to receive their degree in **2026**. Their research interests include **Machine Learning (ML) for gesture recognition**, **Real-time systems**, and **Immersive Education**. For this study, they were responsible for the **Python back-end development, the Mediapipe integration for real-time gesture recognition**, and the overall system architecture. They can be contacted at email: kalpitharb.is23@bmsce.ac.in.