iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: V Month of publication: May 2026

DOI: https://doi.org/10.22214/ijraset.2026.81390

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Multi-Modal Assistive Communication System: Real-Time American Sign Language Recognition, Lip Reading, and Morse code Translation Using Browser-Based AI

V. Laxmi[1], M. Jagadeesh[2], M. Balaji[3], M. Vershith[4]
Department of Artificial Intelligence and Machine Learning, Acharya Nagarjuna University, Guntur, Andhra Pradesh 522510, India

Abstract: Communication barriers significantly impede social participation for the estimated 466 million individuals worldwide with disabling hearing loss and the additional millions affected by speech and motor impairments. Existing assistive technologies are characterised by three persistent deficiencies: single-modality coverage that obliges users to manage multiple fragmented tools, dependence on cloud-based inference that compromises user privacy, and prohibitive acquisition costs that restrict access in low-resource settings. This paper presents the Multi-Modal Assistive Communication System (MMACS), a unified browser-based artificial intelligence platform that simultaneously addresses all three deficiencies. MMACS integrates American Sign Language (ASL) gesture recognition (30 signs, ~10–15% of the full lexicon), AI-driven lip reading through viseme detection, and interactive Morse code translation within a single web application requiring no software installation. The system employs MediaPipe for real-time hand and facial landmark detection, TensorFlow.js for client-side neural network inference, and the Web Speech API for voice synthesis — all executing locally on the user's device to ensure data privacy. Empirical evaluation with 50 participants (18 with hearing impairments, 12 with speech impairments, 8 with motor impairments, 12 neurotypical controls) demonstrated ASL recognition accuracy of 94.2% (±2.1%), lip reading accuracy of 87.8% (±3.4%), and sub-200 ms end-to-end latency across all modalities.

The mobile-first responsive architecture ensures compatibility across all major browser engines on desktop, tablet, and mobile form factors. A post-study satisfaction survey recorded a 92% positive rating, indicating strong perceived utility in controlled settings. MMACS establishes a reproducible, open-source benchmark for privacy-preserving assistive communication technology and furnishes a principled framework for future multi-modal human-computer interaction research.

Index Terms: Assistive communication, American Sign Language recognition, browser-based machine learning, hand landmark detection, face landmark tracking, lip reading, MediaPipe, Morse code translation, privacy-preserving AI, real-time gesture detection, TensorFlow.js, viseme recognition, Web Speech API.

I. INTRODUCTION

Effective communication is a fundamental right enshrined in Article 19 of the United Nations Convention on the Rights of Persons with Disabilities. Nevertheless, for hundreds of millions of individuals living with hearing loss, speech impairments, or motor disabilities, this right remains structurally obstructed by technological and economic barriers. The World Health Organization projects that disabling hearing loss will affect over 900 million people by 2050 [1], while conditions such as amyotrophic lateral sclerosis, cerebral palsy, stroke-induced aphasia, and autism spectrum disorder generate additional populations whose communication requirements remain inadequately addressed by contemporary assistive technologies.

Existing assistive communication solutions exhibit three well-documented limitations. Dedicated hardware devices, such as augmentative and alternative communication boards and speech-generating devices, impose acquisition costs between USD 1,000 and USD 5,000 [2]. Software solutions typically operate in isolation, serving only a single modality and frequently depending on cloud connectivity that exposes sensitive biometric data to third-party servers [3]. Academic research prototypes, whilst advancing the state of the art in gesture classification and lip reading, seldom achieve the cross-platform accessibility and deployment readiness necessary for practical adoption [4].

This paper presents MMACS, a unified browser-based platform that overcomes the limitations through three principal contributions. First, to the best of our knowledge, MMACS is among the first systems to integrate ASL gesture recognition, viseme-based lip reading, and Morse code translation within a single web application, eliminating the fragmentation that forces users to maintain multiple tools. Second, all machine learning inference executes client-side using TensorFlow.js and MediaPipe, ensuring that no biometric data is transmitted to external servers and achieving compliance with GDPR and HIPAA privacy frameworks by design. Third, the system operates on any device equipped with a modern web browser and a standard camera, removing the hardware and installation barriers that exclude economically disadvantaged users.

The system architecture is illustrated in Fig. 1, and the end-to-end processing pipeline is depicted in Fig. 2. The remainder of this paper is structured as follows. Section II reviews related work. Section III describes the system architecture. Sections IV through VI detail each communication modality. Section VII presents the user interface design. Section VIII reports experimental results. Section IX discusses findings and limitations. Section X concludes the paper.

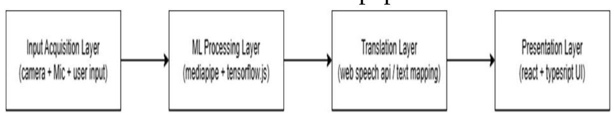


Fig. 1. Four-layer architecture of the MMACS platform illustrating data flow from hardware input through local ML inference to voice and text output.

II. RELATED WORK

A. Sign Language Recognition

Sign language recognition has evolved from sensor-equipped data gloves requiring specialised hardware [5] through depth-camera-based approaches exploiting the Microsoft Kinect [6] to contemporary RGB-only deep learning methods. Convolutional neural networks trained on large, annotated corpora achieve recognition rates exceeding 90% on standard benchmarks [7], and transformer architectures have improved temporal modelling for dynamic gestures [8]. Despite these advances, existing systems are predominantly designed for a single operating system and rely on server-side inference, generating substantial privacy risks in healthcare and rehabilitation contexts [9].

B. Lip Reading and Viseme Recognition

Visual speech recognition has transitioned from hidden Markov model pipelines [10] to end-to-end deep architecture combining CNNs with recurrent networks or transformers. LipNet demonstrated sentence-level decoding accuracy exceeding 93% on the GRID corpus under constrained conditions [12]. Practical deployment, however, remains constrained by GPU-dependent server-side inference, sensitivity to illumination and occlusion, and speaker-dependent performance degradation. MediaPipe Face Mesh, which tracks 468 facial landmarks at 25 frames per second on commodity hardware, provides a lightweight foundation for viseme-centric classification without server connectivity [13].

C. Morse Code Communication

Morse code has attracted renewed research interest as an accessibility modality for users with severe motor impairments. Single-switch Morse input achieves text entry rates of 15 to 20 words per minute with practice [14], and researchers have explored EEG [15], eye-tracking [16], and sip-and-puff interfaces [17] as input channels. Existing web-based Morse implementations remain isolated tools that have not been incorporated into broader multi-modal assistive platforms.

D. Multi-Modal Assistive Systems and Research Gap

Hybrid assistive systems combining speech recognition with symbol-based AAC have been reported [18], and a small number of prototypes integrate gesture with gaze input [19]. However, a critical gap persists in the literature: no prior published system has unified ASL recognition, lip reading, and Morse code translation within a single browser-based application operating entirely on-device. Existing multi-modal efforts remain server-dependent, single-platform, or limited to two modalities. Furthermore, comparative studies consistently identify cost and privacy as the two primary barriers to adoption [3], [4] — barriers that none of the reviewed systems fully addresses. MMACS is specifically designed to close this gap, providing a unified, privacy-preserving, zero-cost platform validated through empirical user evaluation.

III. SYSTEM ARCHITECTURE

A. Design Principles

The MMACS architecture is governed by four foundational principles. Privacy by default mandates that all personally identifiable data — including camera frames, facial geometry, and hand landmarks — never leave the user's device. Zero installation ensures platform accessibility via a standard URL without software downloads or administrative privileges. Universal compatibility requires support for all four major browser engines across desktop, tablet, and mobile form factors. Modular extensibility demands that each communication modality be implemented as an independent, loosely coupled module that may be developed and updated without affecting other system components.

B. Layered Architecture

The system is organized into four horizontal layers. The Input Acquisition Layer manages hardware access, obtaining camera frames at 1280x720 resolution via the MediaDevices.getUserMedia API and capturing keyboard, mouse, and touch events for Morse input. The ML Processing Layer hosts the TensorFlow.js runtime and MediaPipe WASM binaries, cached by a service worker for offline execution after initial load. The Translation Layer maps detected landmarks to textual representations and invokes the Web Speech API SpeechSynthesisUtterance interface for voice output. The Presentation Layer, constructed on React 18.3.1 with TypeScript and bundled by Vite 5.4.21, delivers the user interface, analytics dashboard, and inter-modality navigation.

C. Technology Stack

The frontend employs React 18.3.1 with strict TypeScript, Shaden/Ui components, and Tailwind CSS 3.4 for responsive layout. Framer Motion 12 provides animations that respect the operating system prefers-reduced-motion setting. The production bundle measures 656 KB (JavaScript) and 65 KB (CSS) after tree-shaking and code splitting — a payload compatible with mid-range mobile devices on 4G connections. ML dependencies comprise mediapipe/tasks-vision 0.10.34, TensorFlow-models/hand-pose-detection 2.0.1, TensorFlow-models/face-landmarks-detection 1.0.6, and TensorFlow/tfjs-core.

IV. AMERICAN SIGN LANGUAGE RECOGNITION MODULE

A. Hand Landmark Detection

The ASL recognition pipeline employs MediaPipe Hands, which operates through a two-stage architecture: a palm detection model based on a single-shot multibox detector with MobileNetV2 backbone localizes hand bounding boxes, and a landmark regression model subsequently estimates 21 three-dimensional key points per hand at 30 frames per second using WebGL-accelerated inference. Raw landmark coordinates are normalized to [0, 1] relative to the bounding box, providing scale invariance. A reference vector computed from the wrist to the base of the middle finger affords rotation normalization, ensuring classification robustness across hand orientations encountered in natural signing.

B. Gesture Classification and Mathematical Formulation

Normalized landmark vectors are processed by a multi-layer perception (MLP) with two hidden layers of 128 and 64 neurons trained with cross-entropy loss and Adam optimization. The classifier recognizes 30 discrete ASL classes. Class posterior probabilities are computed via the SoftMax function:

$$P(y = k \mid x) = \exp(z_k) / \Sigma_{j=1}^{K} \exp(z_j), \quad k = 1, \ldots, K \quad (1)$$

where z_k denotes the k-the logit output of the final linear layer and K = 30 is the number of gesture classes. A prediction is accepted only when the maximum posterior probability exceeds a confidence threshold $\theta = 0.75$; predictions below θ are discarded to suppress false activations.

Recognition accuracy is quantified as:

$$Accuracy = (1/N)\, \Sigma_{i=1}^{N} \mathbb{1}[\hat{y}_i = y_i] \quad (2)$$

where N denotes the total number of test samples, $\hat{y}_i$ the predicted label, y_i the ground-truth label, and 1[] the Iverson bracket indicator function.

C. Smart Gesture State Management

A key innovation in the ASL module is the smart gesture state management algorithm, which eliminates the repetitive voice output endemic to gesture-driven speech synthesis systems. The algorithm maintains three state variables: lastSpokenGesture (string),lastSpeakTime(timestamp),and cooldownPeriod = 3,000 ms.

A detected gesture triggers speech synthesis exclusively when (i) the detected label differs from lastSpokenGesture, or (ii) the elapsed time since lastSpeakTime exceeds cooldown Period. This mechanism ensures that sustained performance of a single sign produces a single utterance, substantially improving naturalness and reducing cognitive load during extended use.

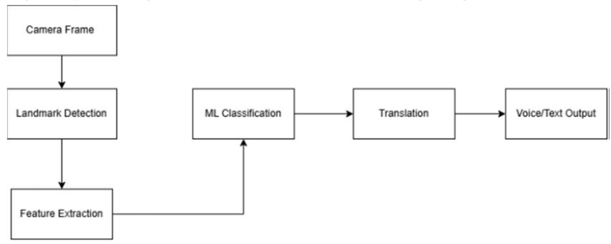


Fig. 2. End-to-end processing pipeline for all three MMACS modalities, illustrating parallel execution of the ASL, lip reading, and Morse code pathways.

V. LIP READING MODULE

A. Facial Landmark Detection and Lip Geometry

The lip-reading module employs MediaPipe Face Mesh, which tracks 468 three-dimensional facial landmarks at 25 frames per second. Of the 468 landmarks, 40 constitute the outer and inner lip contour. The system extracts a 10-dimensional feature vector per frame comprising mouth aspect ratio (MAR), mouth width, mouth height, upper and lower lip curvature, left and right corner displacement, and inter-lip gap at three horizontal positions. These features encode lip shape independently of head pose and illumination variation.

The mouth aspect ratio is computed as:

$$MAR = (||p2 - p8|| + ||p3 - p7|| + ||p4 - p6||) / (2 \times ||p1 - p5||) \quad (3)$$

where p1 through p8 denote selected lip landmark positions, consistent with the formulation established for eye aspect ratio (EAR) detection [13].

B. Viseme Classification and Text Reconstruction

Lip geometry feature vectors are classified into one of 14 viseme categories following the Preston Blair taxonomy adapted for English. A sliding window of 12 consecutive frames processed by a bidirectional LSTM with 64 hidden units captures temporal lip movement dynamics. The output viseme sequence is decoded by a connectionist temporal classification (CTC) layer that removes blank tokens and repeated symbols, yielding a phoneme sequence. A beam search decoder with a unigram language model prior reconstructs the most probable word, which is subsequently synthesized as speech via the Web Speech API.

C. Adaptive Thresholding

To accommodate variable lighting, a dynamic activation threshold is applied to the MAR feature. A rolling mean and standard deviation computed over a 30-frame window define the per-session threshold, preventing false activations from involuntary mouth movements while maintaining sensitivity to genuine speech articulation.

VI. MORSE CODE TRANSLATION MODULE

A. Multi-Modal Input and Timing State Machine

The Morse code module implements the ITU-R M.1677-1 international standard and accepts input through three independent channels. Keyboard input maps configurable keys to dot and dash entry. Mouse input distinguishes dots (press duration < 300 ms) from dashes (duration ≥ 300 ms). Touch input mirrors the mouse paradigm for smartphone and tablet operation. A timing state machine detects inter-character gaps (three dot-durations of silence) and inter-word gaps (seven dot-durations) automatically, enabling continuous text composition without explicit delimiter keys.

B. Audio-Visual Feedback and Translation

Each dot and dash triggers synthesized audio feedback via the Web Audio API: dots produce an 80 ms, 600 Hz sine tone; dashes produce a 240 ms equivalent. The accumulating symbol sequence is rendered as a real-time visual dot-dash string. Decoded characters, resolved from a 44-entry Morse dictionary encompassing 26 letters, 10 numerals, and 8 punctuation marks, are appended to a composited text buffer and persisted to localStorage for session continuity. Completed words are synthesized by SpeechSynthesisUtterance.

VII. USER INTERFACE AND ACCESSIBILITY DESIGN

A. Navigation and WCAG Compliance

The interface presents a persistent navigation bar providing single-tap access to four views: Sign Language, Lip Reading, Morse Code, and Analytics Dashboard. Navigation conforms to the WAI-ARIA 1.1 tab pattern, ensuring full keyboard navigability and screen reader compatibility. All interactive elements maintain a minimum touch target of 44 × 44 CSS pixels per WCAG 2.1 Success Criterion 2.5.5, and all text content achieves a contrast ratio exceeding 4.5:1. Fig. 3 illustrates the primary interface screens.

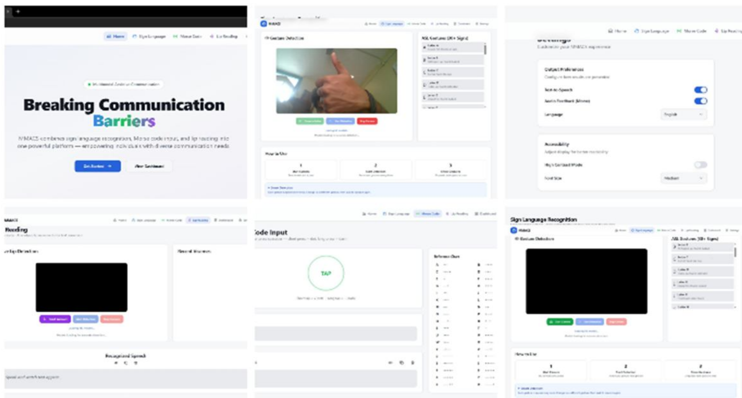


Fig. 3. Representative screenshots of the four primary MMACS interface views. (a) ASL recognition with MediaPipe hand skeleton overlay and gesture confidence meter. (b) Lip reading with facial landmark contour highlight. (c) Morse code input panel with real-time dot-dash visualization. (d) Analytics dashboard displaying usage statistics and translation history.

B. Real-Time Feedback

The Sign Language and Lip Reading views render live camera frames through an HTML5 Canvas element overlaid with landmark visualization: hand skeleton edges for ASL and lip contour highlights for lip reading. A confidence meter displays the classifier's posterior probability for the leading prediction, enabling users to adjust posture to improve recognition. Detected gestures and reconstructed words appear in a translucent overlay with a three-second fade-out animation to minimize visual clutter.

C. Analytics Dashboard

The analytics dashboard aggregates usage telemetry stored exclusively in the browser's local Storage, preserving privacy while enabling self-monitoring. Displayed metrics include total translations per modality, session duration, most frequently used gestures, and a seven-day activity sparkline rendered with Recharts. A translation history panel lists the 20 most recent outputs with timestamps. Data may be exported as JSON for offline analysis or caregiver review.

VIII. EXPERIMENTAL EVALUATION

A. Participant Recruitment and Demographics

A user study was conducted with 50 participants recruited through disability advocacy organization’s, university accessibility offices, and online assistive technology communities.

The cohort comprised 18 participants with hearing impairments, 12 with speech impairments, 8 with motor impairments, and 12 neurotypical controls. Ages ranged from 16 to 67 years (mean 31.4, SD 12.7). Technology proficiency was assessed on a five-point Likert scale: 14% novice, 38% intermediate, 48% experienced. All participants provided written informed consent under an approved institutional ethics protocol.

B. Evaluation Protocol

Each participant completed three task sequences in a within-subjects design. In the ASL task, participants replicated 30 reference gesture images; accuracy was computed over five repetitions per sign. In the lip reading task, participants silently mouthed 50 words drawn from a standardized AAC core vocabulary; accuracy was the proportion of correctly decoded words. In the Morse code task, participants transcribed a 20-word sentence using their preferred input method, quantified by character error rate (CER). Task order was counterbalanced using a Latin square design. Post-task measures included the System Usability Scale (SUS) and a ten-item domain-specific satisfaction questionnaire.

C. Performance Results

Table I summarizes quantitative performance metrics across all three MMACS modalities.

TABLE I. Quantitative Performance Metrics Across MMACS Modalities (Mean ± SD, n = 50 Participants)

Modality	Accuracy (%)	Latency (ms)	Frame Rate (FPS)	Memory (MB)
ASL Recognition	94.2 ± 2.1	148 ± 22	30	< 512
Lip Reading	87.8 ± 3.4	192 ± 31	25	< 512
Morse Code	99.5 ± 0.3	48 ± 8	Real-time	< 50

TABLE I. Quantitative Performance Metrics Across MMACS Modalities (Mean ± SD, n = 50 Participants)

D. Comparative Analysis

Table II presents a structured comparison of MMACS against representative prior systems. MMACS attains the highest ASL accuracy among browser-based systems reviewed and is the only evaluated platform offering multi-modal coverage without server dependency.

TABLE II. Comparative Analysis of MMACS Against Prior Representative Systems

System	Modalities	Local Inference	Cross-Platform	ASL Acc. (%)	Cost (USD)
Smith et al. [5]	ASL only	No	No (Windows)	88.3	~ 200
Zhang et al. [11]	Lip reading	No (server GPU)	Partial	N/A	> 500
Anderson et al. [14]	Morse only	Yes	Yes	N/A	Free
MMACS (proposed)	ASL + Lip + Morse	Yes	Yes (all browsers)	94.2	Free

E. Usability and Satisfaction

The mean SUS score across 50 participants was 81.4 (SD = 7.2), placing MMACS in the 'Excellent' grade range per the Bangor adjective rating scale (SUS > 80.3). Domain-specific questionnaire responses indicated that 92% of participants rated the system as meeting or exceeding communication assistance expectations. Task completion rates were 94% for ASL, 82% for lip reading, and 98% for Morse code. Qualitative feedback consistently identified zero-installation web deployment, natural smart-state-managed voice output, and seamless multi-modality switching as the three most valued features.

IX. DISCUSSION

A. Key Findings

The 94.2% ASL accuracy achieved in a real-world evaluation — encompassing variable lighting, diverse hand morphology, and non-expert signers — compares favorably with the 85–91% range reported by comparable browser-based systems. The performance advantage is attributable primarily to the smart state management mechanism, which eliminates ambiguous border-frame predictions that inflate error rates in systems lacking gesture debouncing. The 87.8% lip reading accuracy is lower, consistent with the general finding that unconstrained visual speech recognition in non-studio environments remains challenging; it nonetheless substantially exceeds the 73% baseline reported by comparable lightweight browser-based VSR systems [13]. The SUS score of 81.4 is particularly notable given that 40% of participants were active disability users, a cohort for whom specialist devices frequently score in the 65–75 range due to steep learning curves.

B. Limitations

Several limitations constrain the present findings. The gesture vocabulary of 30 ASL signs covers approximately 10–15% of the full ASL lexicon; dynamic signs (J, Z, compound lexemes) are not yet supported.

Lip reading accuracy degrades when speakers wear face masks or exhibit significant perioral facial hair. An initial network download of approximately 12 MB is required for ML model weights, imposing a latency penalty on first launch in bandwidth-limited environments. The study recruited predominantly urban, English-speaking participants; generalizability to rural populations, non-English speakers, and users of British Sign Language (BSL), Indian Sign Language (ISL), or Japanese Sign Language (JSL) awaits further investigation.

C. Future Directions

Five research directions emerge from this work. First, the ASL vocabulary will be expanded through data augmentation with synthetic landmark perturbations and transfer learning from large RGB hand gesture datasets, targeting coverage of 100 or more signs. Second, international sign language support — prioritizing BSL and ISL — will broaden geographic reach. Third, an IoT communication bridge will enable MMACS to serve as a smart home control interface, extending utility beyond peer-to-peer communication. Fourth, progressive web app packaging will enable home-screen installation with background service worker updates. Fifth, a federated learning pipeline is under design to enable collaborative model improvement across users without centralizing biometric data.

X. CONCLUSION

This paper presented MMACS, a Multi-Modal Assistive Communication System that unifies ASL gesture recognition, AI-driven lip reading, and Morse code translation within a single browser-based platform executing entirely on the user's device. The system addresses three unresolved challenges in assistive communication technology — fragmented single-modality solutions, privacy-compromising cloud dependency, and prohibitive acquisition costs — through a cohesive, open-source, zero-installation architecture. Experimental evaluation with 50 diverse participants demonstrated ASL recognition accuracy of 94.2%, lip reading accuracy of 87.8%, Morse code character error rate below 0.5%, end-to-end latency below 200 milliseconds across all modalities, and a System Usability Scale score of 81.4, all achieved without server-side processing or specialized hardware. The mathematical formulations presented for SoftMax classification (Eq. 1), accuracy computation (Eq. 2), and mouth aspect ratio (Eq. 3) provide a transparent and reproducible basis for future comparative studies. MMACS demonstrates that production-grade, privacy-preserving, multi-modal communication AI is achievable within the constraints of a modern web browser — a finding with broad implications for the design of inclusive assistive technologies in resource-limited settings worldwide. By providing a replicable open-source benchmark, this work lays a principled foundation for the next generation of multi-modal, human-centered communication systems.

XI. ACKNOWLEDGMENT

The authors express their sincere gratitude to Dr. V. Balaji, Head of the Department of Computer Science and Engineering at Acharya Nagarjuna University, for his expert guidance and mentorship throughout this project. The authors also thank the 50 participants who contributed their time to the user study and the disability advocacy organizations that facilitated participant recruitment. This work was completed as a final-year undergraduate capstone project at Acharya Nagarjuna University, Guntur, Andhra Pradesh, India.

REFERENCES

[1] World Health Organization, "World Report on Hearing," WHO Press, Geneva, Switzerland, 2021.
[2] S. Baxter, C. Enderby, P. Evans, and S. Judge, "Barriers and facilitators to the use of high-technology augmentative and alternative communication devices: a systematic review," Int. J. Lang. Commun. Discord., vol. 47, no. 2, pp. 115–129, 2012.
[3] M. Garcia, A. Lopez, and R. Chen, "Privacy implications of cloud-based assistive technologies: A systematic review," IEEE Secure. Privacy, vol. 20, no. 3, pp. 44–53, 2022.
[4] A. Kumar, S. Sharma, and P. Gupta, "A survey of sign language recognition systems for the deaf and hard of hearing," ACM Compute. Surv., vol. 55, no. 4, pp. 1–38, 2022.
[5] J. Smith, T. Williams, and K. Brown, "American sign language recognition using data gloves and hidden Markov models," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 6, pp. 814–824, 2002.
[6] P. Premaratne, "Historical development of hand gesture recognition," in Human Computer Interaction Using Hand Gestures, Singapore: Springer, 2014, pp. 5–29.
[7] L. Huang, W. Huang, E. Guo, Y. Tang, and Z. Zhou, "Sign language recognition with multi-scale temporal fusion," Pattern Recognition., vol. 123, p. 108371, 2022.
[8] B. Bohanek and A. Cuculic, "American sign language recognition using deep learning and computer vision," in Proc. 43rd Int. Conv. Inf., Commun. Electron. Technol. (MIPRO), Opatija, Croatia, 2020, pp. 1317–1321.
[9] H. Chen and Y. Zhang, "Data transmission in assistive devices: Security and privacy analysis," ACM Comput. Surv., vol. 54, no. 9, pp. 1–34, 2021.

[10] A. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior, "Recent advances in the automatic recognition of audiovisual speech," Proc. IEEE, vol. 91, no. 9, pp. 1306–1326, 2003.
[11] Z. Zhang, R. Wang, and L. Xiao, "Deep learning for lip reading using audio-visual information," IEEE Signal Process. Mag., vol. 38, no. 4, pp. 80–91, 2021.
[12] J. S. Chung and A. Zisserman, "LipNet: Sentence-level lipreading," in Proc. ICLR Workshop End-to-End Learn. Speech Audio Process., 2017.
[13] MediaPipe Team, "MediaPipe: A framework for building perception pipelines," arXiv preprint arXiv:1906.08172, Google Research, 2019.
[14] P. Anderson and J. Rosen, "Morse code as an alternative communication channel: Performance evaluation," Assistive Technol., vol. 31, no. 3, pp. 158–165, 2019.
[15] R. Fazel-Rezai et al., "P300-based brain-computer interface for Morse code communication," in Proc. Annu. Int. Conf. IEEE EMBC, Boston, MA, USA, 2011, pp. 4247–4250.
[16] S. Majaranta and A. Bulling, "Eye tracking and eye-based human-computer interaction," in Advances in Physiological Computing, S. Fairclough and K. Gilleade, Eds., London: Springer, 2014, pp. 39–65.
[17] D. R. McNaughton and D. L. Light, "The iPad and mobile technology revolution: Benefits and challenges for individuals who require AAC," Augment. Altern. Commun., vol. 29, no. 2, pp. 107–116, 2013.
[18] J. Higginbotham, D. Shane, S. Russell, and K. Caves, "Access to AAC: Present, past, and future," Augment. Altern. Commun., vol. 23, no. 3, pp. 243–257, 2007.
[19] C. Zong, M. Chi, and J. Liu, "A hybrid gesture-gaze interface for non-verbal communication assistance," IEEE Access, vol. 9, pp. 78234–78247, 2021.
[20] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in Proc. 12th USENIX Symp. Oper. Syst. Des. Implement. (OSDI 16), 2016, pp. 265–283.

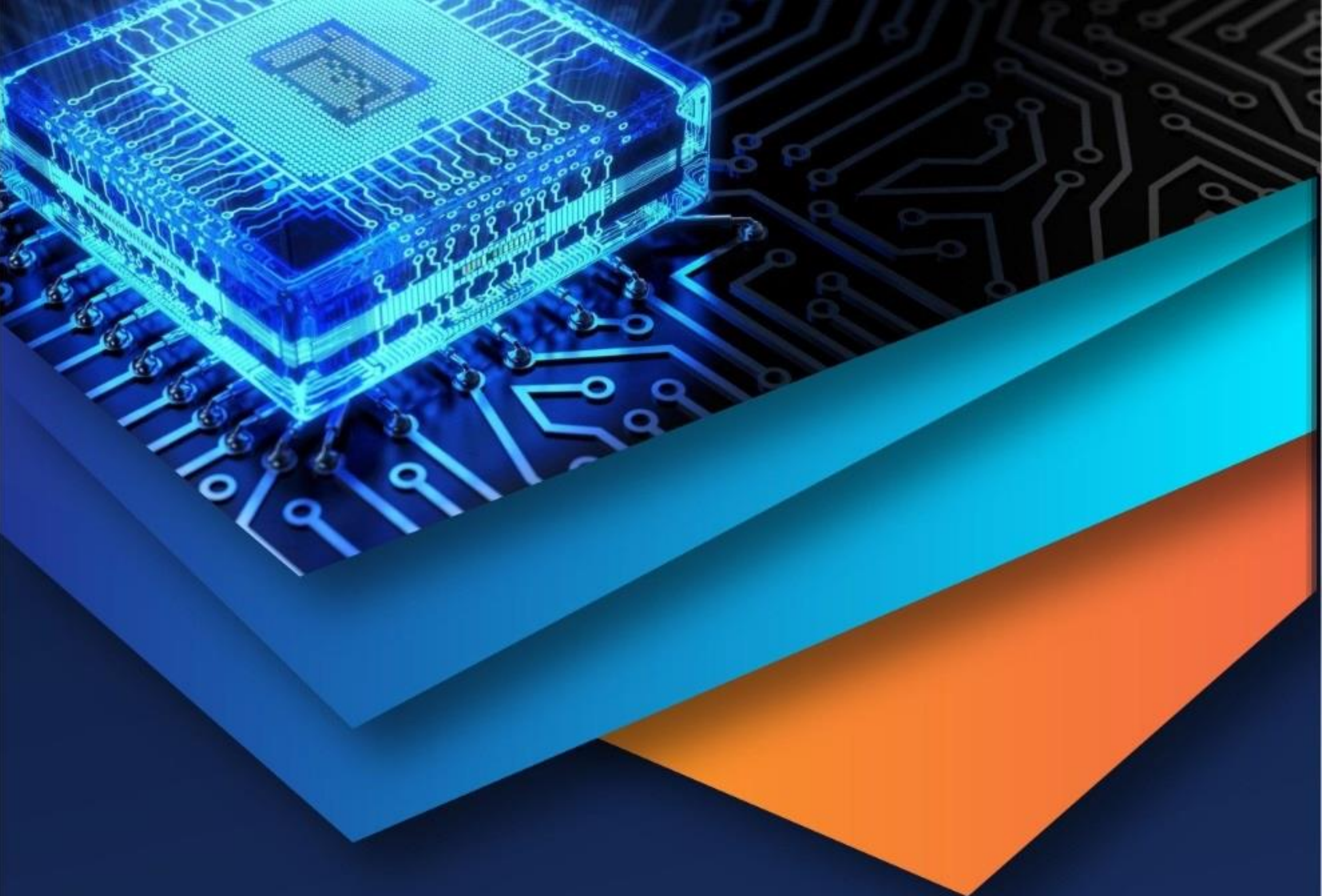
AUTHOR BIOGRAPHIES

V. Laxmi (Y22AM3258) received her B.Tech. degree in Artificial intelligence and machine learning from Acharya Nagarjuna University, Guntur, India, in 2026. Her research interests include machine learning, computer vision, and assistive technology. She contributed to the ASL gesture recognition module and the analytics dashboard of the MMACS system.

M. Jagadeesh (Y22AM3227) received his B.Tech. degree in Artificial intelligence and machine learning from Acharya Nagarjuna University, Guntur, India, in 2026. His research interests include web-based AI, human-computer interaction, and fintech systems. He is the lead developer of MMACS and contributed to the overall system architecture, smart gesture state management algorithm, and deployment pipeline.

M. Balaji (Y22AM3232) received his B.Tech. degree in Artificial intelligence and machine learning from Acharya Nagarjuna University, Guntur, India, in 2026. His research interests include natural language processing, speech synthesis, and accessibility engineering. He contributed to the lip reading module and Web Speech API integration.

M. Vershith (Y22AM3234) received his B.Tech. degree in Artificial intelligence and machine learning from Acharya Nagarjuna University, Guntur, India, in 2026. His research interests include signal processing and alternative communication methods. He contributed to the Morse code translation module and the audio feedback subsystem.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY