Jurnal ICT : Information and Communication Technologies, 17 (1) (2026) 27-31

Published by: Marq & Cha Institute

Jurnal ICT : Information and Communication Technologies

Journal homepage: www.ejournal.marqchainstitute.or.id/index.php/JICT

Real-Time Web-Based Indonesian Sign Language (BISINDO) Translator System Using CNN-LSTM Deep Learning and Text-to-Speech

Nabiel Muhammad Imjauzanansyah[1], Hevlie Winda Nazry S[2]
[1,2] Program Studi Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia

Article Info

Article history

Received : Apr 15, 2026
Revised : Apr 29, 2026
Accepted : Apr 30, 2026

Keywords:

BISINDO;
CNN-LSTM;
Deep Learning;
Real-Time Web;
Text-to-Speech.

Abstract

Indonesian Sign Language (BISINDO) is the primary communication medium for the deaf community, yet low public understanding often causes communication barriers. Previous sign language recognition studies mostly operated offline, lacked real-time web integration, and only produced text output. This study designs and develops a real-time web-based BISINDO translator system using a hybrid Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) method, integrated with a Text-to-Speech (TTS) feature. The dataset consisted of primary video data from 3 subjects, covering 11 category classes with 1,000 frames per class in grayscale format (100x89 pixels). The hybrid CNN-LSTM model was integrated into a React.js and Node.js web application (NzSignify). Testing results demonstrate that the model achieved 96% static accuracy based on Confusion Matrix evaluation. In real-time functional testing, an 80% Confidence Threshold effectively filtered incorrect gestures, enabling accurate translation of valid sign gestures into text and voice output.

Corresponding Author:

Nabiel Muhammad Imjauzanansyah,
Program Studi Sistem Informasi,
Universitas Muhammadiyah Sumatera Utara,
Jl. Kapt. Mukhtar Basri No. 3, Glugur Darat II, Kec. Medan Timur, Kota Medan, Sumatera Utara, 20238, Indonesia
Email: nabiel.zmuhammad11@gmail.com

1. Introduction

Indonesian Sign Language (BISINDO) is a visual communication system used by the deaf community (Tuli) in Indonesia, conveyed through combinations of hand movements, facial expressions, and body postures (World Health Organization, 2025). Despite its vital role, public understanding of BISINDO remains low, creating significant communication barriers in public services, education, and social interaction.

Deep learning approaches have demonstrated strong performance in sign language recognition from images and video. Convolutional Neural Networks (CNNs) are effective at extracting spatial features from hand gestures, while Long Short-Term Memory (LSTM) networks excel at modeling temporal relationships in sequential frame data. The combination of CNN-LSTM has been widely adopted in sign language research, achieving higher accuracy than single-model approaches (Aljabar, 2020; Kumari & Anand, 2024).

However, most existing systems operate offline, lack real-time web integration, and produce only text output without audio conversion (Altairika & Sari, 2023; Fadillah et al., 2022). This creates a gap in practical, accessible communication tools for the deaf community. Furthermore, previous attempts to adapt foreign sign language models (e.g., ASL) to BISINDO have yielded poor accuracy of approximately 30% due to gestural differences (Fadillah et al., 2022).

To address these limitations, this study presents NzSignify — a real-time web-based BISINDO translator system built on a hybrid CNN-LSTM architecture integrated with Text-to-Speech (TTS). The system uses React.js for the frontend and Node.js for the backend, making it accessible without special installation, requiring only a standard webcam (Axza et al., 2023; Ramadhani et al., 2025).

BISINDO developed naturally within the Indonesian Tuli community and differs structurally from the government-standardized SIBI system. Research shows BISINDO is more communicatively intuitive for deaf students (Isnaniah et al., 2023). Its temporal and visual-gestural characteristics make it well-suited for deep learning video processing approaches.

The CNN-LSTM hybrid is the dominant architecture for video-based gesture recognition. CNNs process each frame independently to extract spatial features such as hand shape and orientation, while LSTM networks learn temporal dependencies across the frame sequence, enabling the system to understand the meaning of gestures in context rather than from isolated frames (Hochreiter & Schmidhuber, 1997; Alzubi et al., 2024). Recent studies confirm that CNN-LSTM remains effective and relevant for gesture recognition tasks, particularly where the balance between spatial accuracy and temporal context understanding is critical (Myagila & Nyambo, 2025).

Text-to-Speech (TTS) converts text into synthesized audio output. In sign language translation contexts, TTS serves as an inclusive output layer, allowing hearing individuals to understand sign gestures as spoken words in real time, significantly improving accessibility (Tan et al., 2023). Several previous studies have explored the development of BISINDO recognition systems using various deep learning approaches, yet important limitations remain. Altairika and Sari (2023) developed a real-time BISINDO detection system based on CNN and LSTM architectures. However, their system only produced text-label outputs and did not provide an interactive web-based application or audio feedback for users. In contrast, the present study integrates the CNN-LSTM model into an interactive web application equipped with Text-to-Speech (TTS) functionality, enabling more natural communication. Meanwhile, Deleviar et al. (2025) proposed a speech-to-video BISINDO translation website using the LSTM method. Although the study contributed to sign language accessibility, it focused on the reverse translation direction, namely converting speech into sign language videos. The current study instead emphasizes video isyarat-to-speech translation, which supports oral communication for deaf users in real-time interactions. Furthermore, Fadillah et al. (2022) developed a BISINDO translator using transfer learning from an ASL-based model, but the system achieved relatively low accuracy of only 30% because of the mismatch between ASL gestures and native BISINDO gestures. To address this limitation, the present study trains the CNN-LSTM architecture from scratch using a native BISINDO dataset, resulting in significantly higher recognition performance with an accuracy of 96%.

2. Research Methodolgy

3.1 Research Design

This study employs a quantitative experimental approach, measuring model performance through evaluation metrics including accuracy, precision, recall, F1-score, and loss, represented using a Confusion Matrix. The Prototype development model was chosen to enable iterative refinement of the system.

3.2 Dataset Collection and Preparation

The dataset consists of primary video data recorded using a standard webcam in an indoor environment with controlled lighting. Three subjects were used to capture gestural variation in hand shape, posture, and movement rhythm.

The dataset comprises 11 classes: 10 BISINDO sign words (Halo/Hello, Nama/Name, Saya/Me, Nabiel, Terima Kasih/Thank You, Maaf/Sorry, Software Engineering, Makan/Eat, Minum/Drink, Tidur/Sleep) and 1 idle class. Each class contains 1,000 frames recorded at 100x89 pixels in grayscale format, yielding 33 sequences of 30 frames per class. The full dataset totals 11,000 frames and 363 sequences. The dataset was split into 80% training, 10% validation, and 10% testing sets. Labels were applied using directory-based labeling and converted to One-Hot Encoding for deep learning processing.

3.3 CNN-LSTM Model Architecture

The hybrid CNN-LSTM model processes each video as a sequence of 30 frames. The architecture consists of the following layers:

(1) Input Layer: Receives sequential frames with shape (30, 89, 100, 1). (2) TimeDistributed Conv2D (32 & 64 filters, 3x3 kernel, ReLU): Extracts spatial features from each frame independently. (3) TimeDistributed MaxPooling2D (2x2): Reduces spatial dimensionality while retaining important features. (4) TimeDistributed Flatten: Converts 2D feature maps into 1D feature vectors for sequential processing. (5) LSTM (64 units, ReLU): Learns temporal dependencies across the 30-frame sequence. (6) Dense (64 units, ReLU) + Dropout (0.5): Performs classification while preventing overfitting. (7) Output Layer (Softmax, 11 neurons): Produces probability distribution across 11 gesture classes.

The model was trained for 40 epochs using the Categorical Crossentropy loss function with the EarlyStopping callback (restore_best_weights=True) to save the best-performing model weights. Training was conducted on Google Colaboratory with GPU acceleration.

3.4 System Architecture (NzSignify)

The NzSignify system adopts a three-tier architecture. The React.js frontend captures live webcam video, extracts frames, and displays real-time translation results. A Node.js backend serves as an API bridge between the frontend and the Python-based CNN-LSTM inference server. The Text-to-Speech module (Web Speech API) is invoked client-side upon receiving a valid prediction with confidence ≥ 80%.

3.5 Evaluation Methodology

Model performance is evaluated using: (1) Confusion Matrix to visualize classification results across all 11 classes; (2) Accuracy = (TP+TN)/(TP+TN+FP+FN) × 100%; (3) Precision = TP/(TP+FP); (4) Recall = TP/(TP+FN); (5) F1-Score = 2 × (Precision × Recall)/(Precision + Recall). Real-time performance is additionally evaluated using a Confidence Threshold mechanism: predictions below 80% confidence are withheld from output to prevent erroneous translations.

3. Results and Discussion

4.1 Model Training Performance

Training was conducted over 40 epochs. The model achieved a training accuracy of 96% and a validation accuracy of 93% at the final epoch. The slight gap between training and validation accuracy is normal for deep learning models and is effectively managed by the EarlyStopping callback, which restores the best weights corresponding to 96% static accuracy on the test set.

4.2 Classification Performance (Static Testing)

The model was evaluated on a test set comprising 73 samples. The overall classification accuracy reached 96%, as confirmed by the Confusion Matrix evaluation. The detailed per-class performance is presented in Table 2.

Table 2. Classification Report of CNN-LSTM Model

Gesture Class	Precision	Recall	F1-Score	Support
Diam (Idle)	1.00	1.00	1.00	7
Halo (Hello)	0.90	1.00	0.95	9

Maaf (Sorry)	1.00	0.89	0.94	9
Makan (Eat)	0.88	1.00	0.93	7
Minum (Drink)	1.00	0.80	0.89	5
Nabiel	0.86	1.00	0.92	6
Nama (Name)	1.00	0.80	0.89	5
Saya (Me)	1.00	1.00	1.00	6
Software Eng.	1.00	1.00	1.00	5
Terima Kasih	1.00	1.00	1.00	7
Tidur (Sleep)	1.00	1.00	1.00	7
Overall Accuracy			0.96	73

The model achieved perfect scores (Precision, Recall, F1 = 1.00) on five classes: Diam, Saya, Software Engineering, Terima Kasih, and Tidur. Minor misclassifications occurred in classes with spatial similarity, such as Maaf (Precision 1.00, Recall 0.89), where one sample was misclassified, and Nama (Recall 0.80) where one of five samples was mistakenly predicted as Nabiel — a plausible confusion given similar hand configurations. These results confirm the model performs robustly even when trained on a relatively small primary dataset.

4.3 Real-Time System Performance

Real-time functional testing was conducted directly via browser (localhost) using a live webcam feed. The Confidence Threshold mechanism at 80% proved highly effective: when users performed complete, clear gestures, the system predicted the correct class with confidence exceeding 90%, immediately triggering text display and TTS audio output. When the hand was in a transitional or ambiguous pose, the model distributed probability across multiple classes at low percentages (e.g., 37.9%), and the system correctly withheld any output.

Minor misclassifications observed in the real-time environment included 'Maaf' occasionally predicted as 'Halo', 'Minum' as 'Makan', and 'Nama' as 'Nabiel'. These errors correlate with similar spatial patterns between gesture pairs and are exacerbated by inconsistent lighting or incomplete gesture execution. Instructing users to perform gestures with clear articulation resolved most of these issues.

4.4 Text-to-Speech Integration

TTS was implemented using the browser-native Web Speech API (SpeechSynthesisUtterance), configured for Indonesian language (lang = 'id-ID') at a speech rate of 0.9. When a prediction exceeds the 80% threshold, the TTS module immediately synthesizes and vocalizes the translated word, achieving minimal perceptible latency. This enables hearing individuals to understand communicated sign gestures in real time without visual attention to the screen.

4. Conclusion

This study successfully designed and developed NzSignify — a real-time web-based BISINDO sign language translator system using a hybrid CNN-LSTM deep learning architecture integrated with Text-to-Speech. The key findings are as follows. The hybrid CNN-LSTM architecture is highly effective for BISINDO video processing. TimeDistributed CNN layers extract spatial hand features per frame, while LSTM captures temporal gesture patterns across 30-frame sequences. The model achieved 96% overall accuracy on the static test set, demonstrating robust performance across all 11 gesture classes despite training from scratch on a relatively modest primary dataset. The NzSignify web application built on React.js and Node.js delivers functional real-time translation accessible via any standard browser with a webcam, without specialized hardware or installation. The 80% Confidence Threshold mechanism effectively prevents erroneous output during transitional or ambiguous gestures, significantly improving real-time system reliability. Future work should focus

on: (1) expanding the gesture vocabulary from word-level to sentence-level BISINDO; (2) integrating MediaPipe hand landmark tracking for noise-robust feature extraction; and (3) applying model compression (quantization) techniques to improve FPS performance on low-specification devices.

References

World Health Organization. (2025). Deafness and hearing loss. WHO. https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

Aljabar, A. (2020). BISINDO (Bahasa Isyarat Indonesia) Sign Language Recognition Using CNN and LSTM. Journal of Information Systems, 5(5), 282-287.

Altairika, E., & Sari, W. P. (2023). Pengembangan Deteksi Realtime Bahasa Isyarat Indonesia Menggunakan CNN dan LSTM. Jurnal Teknologi Informatika Dan Komputer, 9(1), 1-13. https://doi.org/10.37012/jtik.v9i1.1272

Fadillah, R. Z., Irawan, A., & Susanty, M. (2022). Model Penerjemah Bahasa Isyarat Indonesia (BISINDO) Menggunakan Pendekatan Transfer Learning. 15(1), 1-9.

Isnaniah, S., Agustina, T., Islahuddin, & Annisa, F. (2023). Perbandingan Pemahaman Bahasa Isyarat Indonesia dan SIBI dalam Pembelajaran Siswa Tuli. Jurnal Pendidikan Luar Biasa.

Khan, S., Rahmani, H., Shah, S. A. A., & Bennamoun, M. (2021). A Survey of the Recent Architectures of Deep Convolutional Neural Networks. Artificial Intelligence Review, 53(8), 5455-5516. https://doi.org/10.1007/s10462-020-09825-6

Alzubi, J., Nayyar, A., & Kumar, A. (2024). Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications. Information, 15(9), 517. https://doi.org/10.3390/info15090517

Myagila, K., & Nyambo, D. G. (2025). Efficient spatio-temporal modeling for sign language recognition using CNN and RNN architectures. Frontiers in Artificial Intelligence. https://doi.org/10.3389/frai.2025.1630743

Tan, X., Wang, T., & Chen, J. (2023). NaturalSpeech: End-to-End Text to Speech Synthesis with Human-Level Quality. IEEE Transactions on Audio, Speech, and Language Processing.

Kumari, D., & Anand, R. S. (2024). Isolated Video-Based Sign Language Recognition Using a Hybrid CNN-LSTM Framework Based on Attention Mechanism.

Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach. Pearson.

Axza, F., Sofi, F., & Qoiriah, A. (2023). Analisis Perbandingan Framework Front-End Javascript React dan Vue Pada Pengembangan Website. 05, 157-164.

Ramadhani, A., Iriadi, N., & Hidayat, R. (2025). Implementasi Teknologi Rest API Dengan Node Js Untuk Aplikasi Rekomendasi Destinasi Wisata. 4(1), 22-29.

Deleviar, M. A., et al. (2025). Speech-to-Video BISINDO Website Using LSTM. Jurnal Teknologi Informatika.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735-1780.