Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Deep Learning-Based Recognition of Indonesian Sign Language (BISINDO) Alphabetic Gestures Using Skeletal Feature Extraction and LSTM

Teuku M Arief Afwan*[1], Rahmat Gernowo[2], Helmie Arif Wibawa[3]

[1]Master Program of Information System, Diponegoro University, Semarang, Indonesia
[2]Doctoral Program of Information System, Diponegoro University, Semarang, Indonesia
[3]Department of Computer Science / Informatics, Diponegoro University, Semarang, Indonesia

Email: [1]tmariefafwan@gmail.com

Received : Sep 25, 2025; Revised : Nov 11, 2025; Accepted : Nov 12, 2025; Published : Apr 15, 2026

Abstract

Communication is a fundamental aspect of human life, and for the deaf community, sign language serves as the primary medium of interaction. In Indonesia, the Indonesian Sign Language (BISINDO) is widely used, however, research on automatic BISINDO recognition remains limited due to the scarcity of representative datasets. This study presents the development of a BISINDO recognition system based on deep learning by integrating the Long Short-Term Memory (LSTM) architecture with the MediaPipe Holistic framework. To address data limitations, a custom dataset comprising 866 BISINDO alphabetic gesture videos was collected, involving recordings from both expert and non-expert signers to capture stylistic variations. Extracted skeletal landmark features were processed through a three-layer LSTM network followed by dense layers for sequential modeling and classification. Experimental results show that the proposed model achieved a validation accuracy of approximately 93%, outperforming static image–based methods and demonstrating the effectiveness of skeletal features in representing dynamic gestures. The model also exhibited real-time applicability with promising performance, although challenges such as misclassification of visually similar gestures and dataset imbalance remain. This study contributes to the underexplored field of BISINDO recognition by providing a baseline system and dataset, and further advances the domains of computer vision and human–computer interaction within informatics through an inclusive, data-driven framework for Indonesian Sign Language recognition and future AI-assisted accessibility technologies.

Keywords : *Deep Learning, Hand gesture, LSTM, Mediapipe, Sign Language*

1. INTRODUCTION

Around the world, sign language is an essential visual communication system for deaf and hard-of-hearing individuals [1]-[2]. It serves as a fundamental means of interaction for deaf communities worldwide, encompassing over 300 distinct sign languages [3], each uniquely shaped by its own cultural and linguistic context. In sign language, each movement can represent a letter, word, or number, which combine to form complete sentences [4][5]. Due to the historical evolution of sign languages, regional variations have emerged, each with a distinct grammar and vocabulary [6]-[7].

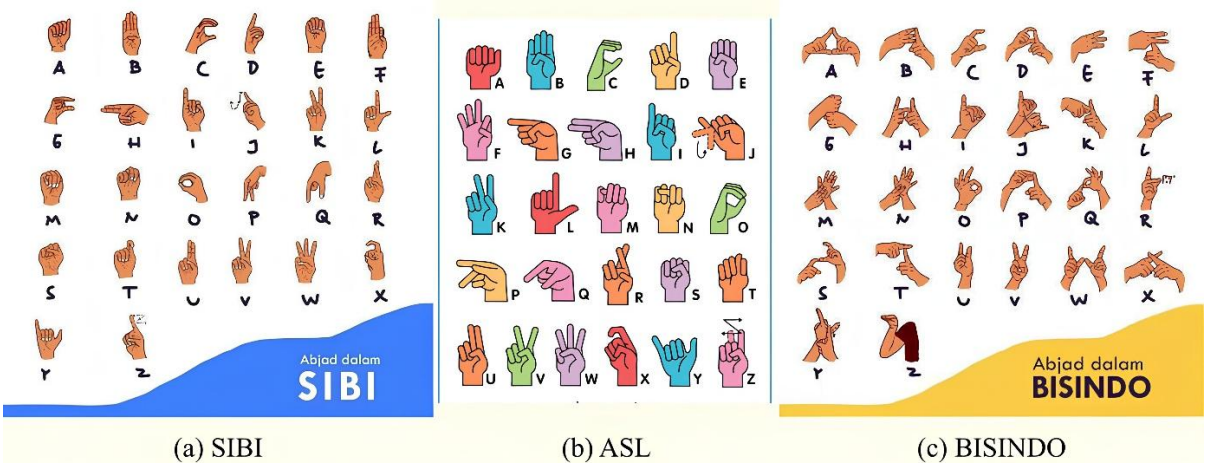
Despite its global importance [8], sign language remains challenging to understand and process due to its visual complexity and non-verbal structure, which continue to cause significant communication barriers for deaf individuals [9]-[10]. Consequently, the technological infrastructure supporting automatic sign language recognition (SLR) and translation is still underdeveloped [11]-[12]. Most research efforts have focused on widely used international sign languages such as American Sign Language (ASL) and Indian Sign Language (ISL), where large annotated datasets are publicly available. However, this global concentration has left many regional sign languages underrepresented. Without

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

sufficient data, deep learning systems struggle to capture variations in signer style, fluency, and gesture dynamics [13]-[14], limiting their ability to generalize across different contexts.

In Indonesia, Bahasa Isyarat Indonesia (BISINDO) serves as the primary sign language, playing a vital role in facilitating social, educational, and economic participation among the deaf community [15]-[16]. Despite its significance, research on automatic BISINDO recognition remains limited due to the scarcity of standardized and representative datasets. Existing BISINDO datasets predominantly consist of static images of alphabet signs, which lack the temporal dynamics and expressive motion features necessary for real-world recognition. This limitation significantly restricts the development of accurate and generalizable recognition models.

Apart from data limitations, linguistic diversity within Indonesian sign systems also poses an additional challenge. Indonesia officially recognizes two primary variants: BISINDO and Sistem Isyarat Bahasa Indonesia (SIBI) [17]. BISINDO is more widely adopted for its accessibility and cultural inclusivity, while SIBI is heavily influenced by foreign linguistic systems such as ASL [18] tends to be more structured but less representative of the natural signing used by the community [16]. Figure 1 shows the differences between BISINDO, SIBI, and ASL alphabets. The image clearly illustrates that the SIBI alphabet closely resembles American Sign Language (ASL), with many hand movements identical to those seen in ASL. Therefore, BISINDO is considered a more inclusive and culturally relevant communication system within the Indonesian deaf community.



(a) SIBI (b) ASL (c) BISINDO

Figure 1. Different Between SIBI (a), ASL (b), and BISINDO (c)

Although the SIBI dataset has been widely studied [19], BISINDO datasets remain challenging, particularly for modeling temporal gesture sequences that reflect real signing. Visual SLR is inherently complex within the domain of computer vision, as it requires sensitivity to subtle variations in hand shape, trajectory, temporal dependency, and body posture [3][20]. Several previous studies have attempted to address this challenge using image-based BISINDO datasets. For instance, Indra et al. [15] applied traditional image processing techniques such as chain code and Euclidean distance, achieving an accuracy of around 95%. Dwijayanti et al. [17] later adopted Convolutional Neural Networks (CNNs) for static BISINDO alphabet recognition and reported an accuracy of 98.3%, while Gani et al. [21] employed a ResNet-50 architecture with pruning optimization, achieving approximately 95–96% accuracy. These results demonstrate significant progress in BISINDO recognition research. However, existing works primarily rely on static image datasets that do not capture the temporal movement patterns essential to natural signing. These limitations hinder the development of inclusive technologies that can accurately interpret dynamic BISINDO gestures in real-world contexts.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Recent advances in deep learning, particularly with Long Short-Term Memory (LSTM) networks, have shown strong capability in modeling such temporal dependencies within sequential gesture data [22]. LSTM's architecture allows it to retain and selectively forget information across time steps, thereby preserving contextual information throughout a sign sequence [23]. This property is crucial for differentiating visually similar gestures in continuous sign recognition tasks [24]. Alongside temporal modeling, skeletal-based detection frameworks such as MediaPipe have shown strong potential in extracting spatial features relevant to gesture understanding [25].

MediaPipe can detect up to 1,662 landmark points across the face, hands, and body in each frame [26], efficiently capturing critical spatial relationships while maintaining robustness against environmental noise such as lighting or background variation [27]-[28]. This capability enables the extraction of detailed motion patterns essential for accurate gesture classification [29]. When combined, MediaPipe and LSTM offer a complementary hybrid approach: MediaPipe handles precise spatial landmark extraction, while LSTM captures the temporal progression of these landmarks across video frames. Although such frameworks have been effectively applied to ASL and ISL recognition [30], research specific to BISINDO remains sparse. This gap emphasizes the urgent need for localized recognition systems tailored to the linguistic and cultural characteristics of Indonesian signers. This combination allows the model to simultaneously capture spatial and temporal dependencies, two critical aspects often overlooked in prior BISINDO research.

To overcome these limitations, this research presents a deep learning–based BISINDO recognition framework that combines Long Short-Term Memory (LSTM) networks for modeling temporal dynamics with MediaPipe Holistic for extracting spatial skeletal landmarks. The proposed system aims to accurately detect, classify, and translate BISINDO alphabet gestures from video input while maintaining real-time performance. A new dataset of 866 BISINDO alphabet gesture videos was developed as part of this study, consisting of recordings from both expert and non-expert signers to capture stylistic variation and ensure model generalization. Experimental evaluation shows that the proposed model achieved approximately 93% validation accuracy, surpassing conventional static image–based methods and underscoring the effectiveness of integrating skeletal and temporal representations.

This research advances the underexplored field of BISINDO recognition by developing a comprehensive skeletal-based dataset and a hybrid deep learning framework specifically trained on a newly constructed BISINDO dataset. Together, these contributions establish a new empirical foundation for Indonesian Sign Language research and open broader opportunities in computer vision and human–computer interaction. In addition to improving recognition accuracy, this research contributes toward building of more inclusive AI technologies that foster equitable communication accessibility for Indonesia's deaf community.

2. RELATED WORKS

The development of SLR models has gained significant traction in computer science research, largely driven by rapid progress in deep learning technologies. A growing body of research has aimed to enhance both the precision and computational efficiency of sign language interpretation across diverse linguistic contexts through advanced computer vision approaches.

One notable study by Cui et al. [31] introduced a recognition framework that integrates Convolutional Neural Networks (CNN) with Bidirectional Long Short-Term Memory (Bi-LSTM) to interpret American Sign Language (ASL) from continuous video sequences. Utilizing datasets such as RWTH-PHOENIX-Weather 2014 and SIGNUM, their model achieved notable reductions in word error rate (WER), underscoring the advantages of combining spatial and temporal representation learning.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Similarly, Rakshit et al. [32] compared various CNN architectures for static ASL alphabet recognition. Their dataset comprised 78,000 RGB images, with each alphabet sign represented by 3,000 images. The study found that models utilizing ReLU activation functions and dropout regularization achieved accuracies exceeding 98%, emphasizing the importance of hyperparameter tuning and model selection in sign language classification.

Hao et al. [12] proposed a multi-head attention-enhanced Bi-LSTM architecture combined with sensor data for Chinese SLR. Their BMCNN model outperformed traditional machine learning methods with an accuracy exceeding 99%, indicating the potential of hybrid deep learning approaches and multi-modal inputs.

In Arabic SLR, Ismail et al. [33] employed a multi-model CNN approach combining DenseNet121 and VGG16, achieving near-perfect accuracy on a large dataset of 220,000 images. This multi-model strategy showed superior performance over single-model baselines, enhancing robustness in static sign recognition.

Deep learning methods employing 3D-CNN and 2D-CNN architectures have also been investigated for Greek SLR [34]. Their findings indicated that 3D-CNN models were more effective for isolated sign recognition, whereas 2D-CNN models excelled in continuous SLR scenarios.

The emergence of landmark-based detection frameworks, such as MediaPipe, has opened new avenues for SLR. Sánchez-Vicinaiz et al. [26] demonstrated the application of MediaPipe combined with CNNs for fingerspelling detection in Mexican Sign Language, achieving promising real-time performance with low computational cost.

Similarly, Bora et al. [35] developed a real-time Assamese SLR model using MediaPipe for feature extraction and a feedforward neural network for classification. The dataset contained 2,094 labeled samples, covering nine static gestures representing Assamese vowels and consonants. The proposed model achieved 99% recognition accuracy, validating MediaPipe's utility in real-time gesture classification for regional sign languages.

In the Indian context, the integration of MediaPipe with LSTM networks has been further explored for continuous SLR. Srivastava et al. [36] implemented MediaPipe Holistic combined with a six-layer LSTM model (three LSTM and three Dense layers) for Indian SLR using a dataset of 45 gestures, achieving 88.23% accuracy. Their work highlighted MediaPipe's capability in capturing dynamic motion sequences while reducing model complexity. Similarly, Khartheesvar et al. [30] applied MediaPipe Holistic and LSTM on the INCLUDE and INCLUDE-50 datasets, reporting accuracies of 94.8% and 87.4%, respectively. Their study emphasized the use of data augmentation techniques to enhance model generalization, particularly in distinguishing highly similar gesture classes. Table 1 summarizes several previous studies on SLR, highlighting the datasets used, sign language domain, employed methods, and reported accuracies.

As shown in Table 1, these reviewed studies demonstrate significant progress in SLR across diverse languages. Despite these advancements, research focusing on BISINDO remains limited. Nonetheless, research specifically targeting BISINDO remains relatively scarce. The integration of LSTM networks with MediaPipe-based feature extraction has proven to be a highly effective strategy for enhancing both accuracy and real-time performance, indicating its potential for BISINDO recognition.

To overcome the scarcity of publicly available BISINDO datasets, this research introduces a deep learning framework that integrates MediaPipe Holistic for spatial landmark extraction with LSTM networks for temporal pattern learning. The model is applied to a newly developed dataset of BISINDO alphabet gestures, aiming to advance research in this underexplored area and support the development of inclusive communication technologies in Indonesia.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Table 1. Summary of Previous Studies on SLR Across Different Languages and Methods.

Study	Language	Dataset / Source	Method	Reported Accuracy
Cui et al. [32]	American Sign Language	RWTH-PHOENIX-Weather 2014, SIGNUM	CNN + Bi-LSTM	91%
Rakshit et al. [33]	American Sign Language (Alphabet)	78,000 RGB images (26 classes)	CNN with ReLU + Dropout	98%
Hao et al. [13]	Chinese Sign Language	10 gestures, >10,000 samples (sensor-based)	Multi-head Attention Bi-LSTM (BMCNN)	99%
Ismail et al. [34]	Arabic Sign Language	220,000 images	Multi-model CNN	99%
Adaloglou et al. [35]	Greek Sign Language	Public dataset, 10,295 sentences	3D-CNN and 2D-CNN	89% / 85%
Sánchez-Vicinaiz et al. [27]	Mexican Sign Language	336 test images	MediaPipe + CNN	84%
Bora et al. [36]	Assamese Sign Language	2,094 labeled samples gesture	MediaPipe + Feedforward NN	99%
Srivastava et al. [37]	Indian Sign Language	45 gesture videos	MediaPipe Holistic + 6-layer LSTM	88.23%
Khartheesvar et al. [31]	Indian Sign Language	INCLUDE / INCLUDE-50, 2 public datasets	MediaPipe Holistic + LSTM	94.8% / 87.4%

3. RESEARCH METHOD

The methodology involves four primary stages, each of these stages contributes to building an efficient and accurate model for recognizing BISINDO gestures based on skeletal data and temporal learning. As shown in Figure 2.

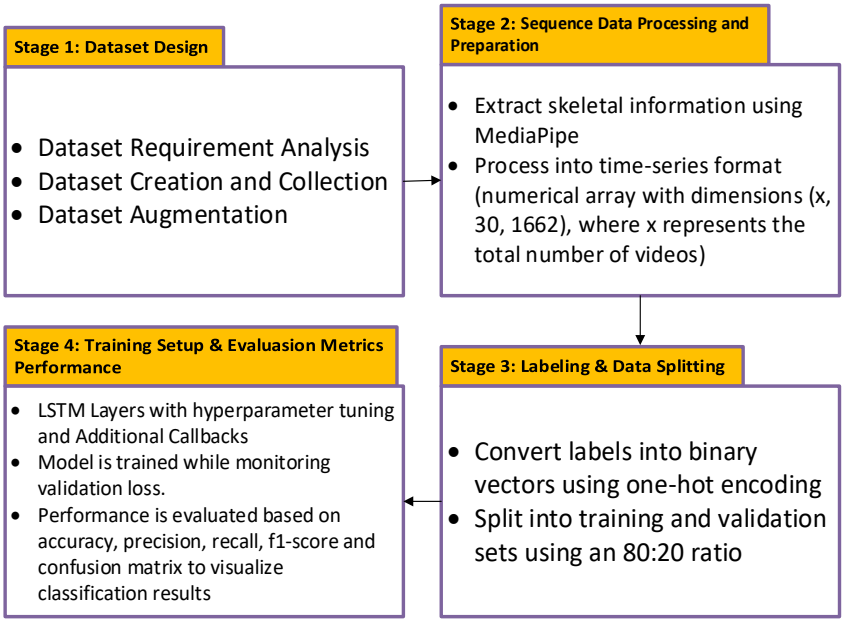


Figure 2. Research Methodology

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

3.1. Stage 1: Dataset Design

A comprehensive BISINDO alphabets videos dataset was collected to support the training and evaluation of the proposed recognition model. The dataset consists of video sequences of BISINDO alphabet gestures performed by three individuals. This selection of three actors ensures variability in gesture presentation, enriching movement data and reflecting the diversity of gestures in the BISINDO community. The videos were captured using a standard webcam operating at a minimum frame rate of 30 frames per second, ensuring adequate spatial and temporal resolution for precise gesture analysis. Each video lasts between 1 and 2 seconds, providing enough time for each gesture to be clearly expressed.

To improve the model's ability to generalize and remain resilient to variations, several data augmentation strategies such as random rotation and scaling were applied. These approaches increase the diversity of training samples, allowing the model to handle differences in gesture appearance, thereby improving its recognition performance under diverse conditions. In total, 866 videos were gathered and used for both training and testing phases.

3.2. Stage 2: Sequence Data Processing

The MediaPipe Holistic framework was utilized to extract spatial keypoints from every frame, providing real-time detection of 1,662 landmarks distributed across the face, hands, and upper body. which are detailed in Table 2. MediaPipe converts them into a skeletal representation for each frame [37], allowing consistent tracking even in scenarios where certain body parts are partially occluded or not entirely visible. This comprehensive landmark detection enables detailed analysis of pose and gestures, which are essential for BISINDO recognition.

Table 2. MediaPipe Holistic Landmark.

Body Part	Landmarks Tracked	Total Points	Coordinates Collected
Pose	33 points	33	X, Y, Z for full body
Hands	21 points	42 (2 hands)	X, Y, Z for each point in each finger
Face	468 points	468	X, Y, Z for facial points

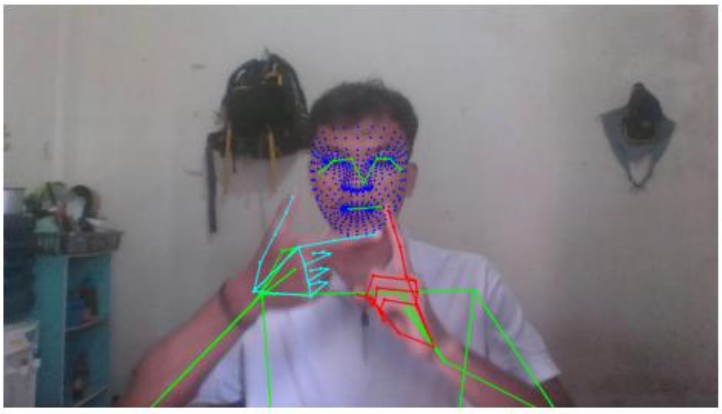


Figure 3. An example frame with skeletal features extraction using MediaPipe Holistic

The table summarizes the key body parts tracked by MediaPipe and the total number of landmarks used for gesture recognition. The coordinates collected are in 3D space (X, Y, Z), for every frame, the extracted features include 468 facial landmarks, 42 hand landmarks (21 per hand), and 33 upper body pose landmarks. Each landmark is represented by three-dimensional coordinates (x, y, z), with pose landmarks additionally including visibility scores. The total number of features per frame sums to 1,662, reflecting the combined spatial data from all detected points. This method effectively captures the

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

essential features of gestures while minimizing their sensitivity to background noise or lighting conditions. Figure 3 illustrates an example frame after MediaPipe processing, showing the extracted skeletal landmarks.

The skeletal data, which consists of the coordinates of these keypoints, is then preprocessed into a time-series format. For each video consisting of 30 frames, the resulting feature array has a shape of (30, 1662), effectively capturing the temporal dynamics of spatial features. When multiple videos are considered within a class for instance, 10 videos the combined data shape becomes (10, 30, 1662), which is suitable for sequence-based models such as LSTM networks. In total, 866 videos were successfully extracted, resulting in an overall data shape of (866, 30, 1.662).

3.3. Stage 3: Labelling and Data Splitting

To prepare the dataset for training, the class labels were converted into binary vectors through a process known as one-hot encoding. Each video label was then converted into a numerical label (e.g., {A: 0, 'B': 1, 'C': 2}) and subsequently transformed into one-hot encoded format using Keras's to_categorical function. For instance, the encoded results are as follows: label 0 → [1, 0, 0], label 1 → [0, 1, 0], and label 2 → [0, 0, 1], and so forth. This transformation ensures that the labels are compatible with the Softmax activation function used in the model's output layer. Initially, class labels, in string format, were mapped to integer indices. These integer indices were then converted to one-hot encoded vectors using the Keras to_categorical function, as shown in Figure 4.

```
array([[1, 0, 0, ..., 0, 0, 0],
       [1, 0, 0, ..., 0, 0, 0],
       [1, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 1],
       [0, 0, 0, ..., 0, 0, 1],
       [0, 0, 0, ..., 0, 0, 1]])
```

Figure 4. Labels after converted into binary using one-hot encoding.

All gesture samples in the dataset represent isolated alphabetic signs (A–Z) rather than continuous or word-level signing sequences to maintain consistency in labeling. Following this, the dataset was split into two subsets: training and testing sets, following an 80:20 ratio. This was done using the train_test_split function from the scikit-learn library. Results of splitting are shown in Table 3.

Table 3. Number of training and validation set

Category	Value
Training Set	692
Validation Set	174
Total	866

To guarantee proportional representation of each class in both the training and validation datasets, stratified sampling was employed. This method maintains balanced class distribution, which is crucial for stable model learning and objective performance assessment. Consequently, the model was trained on a representative sample, promoting better generalization.

3.4. Stage 4: Training Setup and Evaluation Metrics

After dividing the dataset into training and validation subsets, the proposed BISINDO recognition model was constructed using a stacked LSTM neural network, which is particularly effective for

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

capturing spatiotemporal features from sequential input data [38], the LSTM network architecture also needs to be set up to capture both spatial features of body movements and temporal dependencies between consecutive frames [39][23]. This allows the model to learn dynamic gesture patterns over time. The core mechanism of the LSTM is based on the concept of cell state [40]-[41], which is shown in Figure 5.

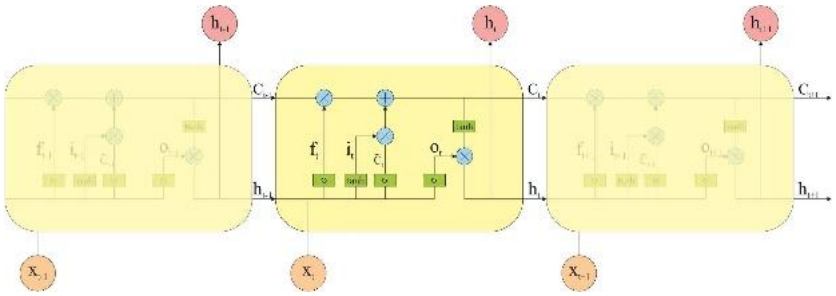


Figure 5. LSTM cell state gate.

This cell state is regulated by three main gates. The input gate, which determines how much of the current input should be added to the memory cell, is calculated in Equation 1.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (1)$$

The forget gate, which controls what portion of the previous memory should be discarded, is described in Equation 2.

$$f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big) \quad (2)$$

The output gate, which decides how much of the memory should be passed to the output, is given in Equation 3.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (3)$$

3.4.1. Training Setup

The model was designed based on a stacked LSTM neural network, which is well-suited for learning spatiotemporal features from sequential input data [42]. The model architecture consists of the following components:

- Input Layer: Accepts sequences of 1,662 landmark features per time step.
- Stacked LSTM Layers: three LSTM layers with tunable units, incorporating dropout regularization to prevent overfitting.
- Dense Layers: One or two fully connected layers for nonlinear transformation of LSTM outputs.
- Output Layer: Softmax activation producing probabilities over 26 classes.

Several hyperparameters tuning experiments were conducted to optimize model, such as dropout rate, learning rate, batch size, and LSTM units. Beyond standard regularization methods like dropout and L2, techniques such as adaptive learning rate adjustment, early stopping, and checkpointing were applied to minimize overfitting and improve model generalization [43].

The configuration of 500, 400, and 300 LSTM units was determined empirically through multiple tuning experiments. Different combinations of layer depth and unit size were evaluated, ranging from two to five LSTM layers with 128–512 units each. The selected three-layer configuration achieved the best balance between training stability, validation accuracy, and computational efficiency. A deeper network tended to overfit due to the relatively small dataset size, while shallower networks were insufficient to capture long-term temporal dependencies. The progressive reduction in unit size (500 → 400 → 300) promotes hierarchical temporal abstraction, enabling earlier layers to learn coarse motion patterns while later layers focus on fine-grained gesture transitions.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

The use of the tanh activation function in LSTM cells rather than the more commonly used ReLU was also an empirical decision. Within recurrent structures, tanh provides better control over gradient flow and prevents exploding values during long-sequence training, which proved crucial for BISINDO gestures characterized by variable motion speed. In contrast, ReLU activation was retained in dense layers for computational efficiency and faster convergence during feature refinement.

The processed skeletal data were used to train t he network using categorical cross-entropy as the objective function, while the Adam optimizer was applied for weight adjustment during learning. Adam was selected due to its simplicity and strong computational performance [44]. The complete architecture of the LSTM-based BISINDO recognition system, including layers, parameters, and complexity, is summarized in Table 4.

Table 4. LSTM Architecture

Layers	Output Shape
LSTM 1	(x, 30, 500)
LSTM 2	(x, 30, 400)
LSTM 3	(x, 300)
Dense 1	(x, 250)
Batch Normalization	(x, 250)
Dropout	(x, 250)
Dense 2	(x, 150)
Output Dense	(x, 26)

The proposed model architecture is composed of three sequentially stacked LSTM layers designed to learn hierarchical temporal relationships, where the number of neurons gradually decreases across layers:

- First LSTM layer: This initial layer contains 500 units and is configured with return_sequences=True, enabling the full temporal sequence to be passed to the next layer. It utilizes the hyperbolic tangent (tanh) activation function and applies a recurrent dropout rate of 20% to minimize overfitting.
- Second LSTM layer: Comprising 400 units, this layer also maintains return_sequences=True with the same activation and dropout configuration, allowing continued propagation of sequential information.
- Third LSTM layer: The final recurrent layer includes 300 units and outputs only the last time step (return_sequences=False), effectively summarizing the overall temporal dependencies.

After the LSTM stack, the extracted temporal features are processed through fully connected dense layers:

- A dense layer with 250 neurons activated by ReLU is introduced, followed by batch normalization to improve training stability and convergence speed.
- Dropout with a rate of 30% is applied for regularization.
- The subsequent dense layer consists of 150 ReLU-activated neurons, which enhance the learned feature abstractions.
- The final output layer includes 26 neurons corresponding to the total number of gesture classes employing a softmax activation to produce normalized class probabilities.

LSTM employs the tanh activation function as it produces balanced values within the range of -1 to 1. This function facilitates long-term learning and effectively regulates the flow of information in sequential data [45]. Meanwhile, ReLU is widely used in dense hidden layers because it introduces non-linearity while preserving computational efficiency. The ReLU activation function enables the network to transmit only positive signals while suppressing negative ones, effectively addressing the vanishing

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

gradient phenomenon and improving training efficiency [46]. In multi-class classification, the softmax function is applied in the output layer to convert raw model outputs into normalized probability values. This process allows the network to express the likelihood of each class, with the highest probability representing the predicted category [47].

The training configurations utilized for this model encompass hyperparameter settings, callback functions such as early stopping and learning rate schedulers. In addition, they encompass the evaluation metrics employed to assess model performance. These are comprehensively detailed and summarized in Table 5.

Table 5. Model Configuration

Configuration Name	Value
Optimizer	Adam
Learning Rate	0.0001
Early Stopping	patience=50
ReduceLROnPlateau	patience=5, factor=0.2
Batch Size	32

To ensure the dependability of the model and assess the consistency of its performance, a three-fold cross-validation approach was employed on the training subset, which comprised 80% of the total dataset. In this method, the training subset was split into three equal segments, where in each iteration two segments were used for model training and the remaining one for validation. The process was carried out three times so that every sample acted once as a validation instance. The average accuracy obtained from all folds was then considered as an indicator of the model's generalization capability. Afterward, the model was trained again using the entire 80% training data and subsequently tested on the remaining 20% to determine its final evaluation metrics.

The experiments were conducted using the Kaggle kernel environment, which provides access to NVIDIA Tesla P100 GPUs for accelerated computation, enabling efficient processing of the computationally intensive LSTM training. The implementation was carried out in Python, utilizing the TensorFlow and Keras libraries to build and train the LSTM neural network. GPU acceleration through TensorFlow significantly reduced training time and facilitated efficient handling of the large sequential BISINDO dataset.

3.4.2. Evaluation Protocol

To evaluate how well the model performed, a consistent assessment framework was applied. Several quantitative indicators were considered, including overall accuracy, class-level precision, sensitivity (recall), and the harmonic mean between them (F1-score). These measures collectively describe how effectively the model distinguishes among different gesture categories. Furthermore, a multi-class confusion matrix was generated to visualize class-wise performance, identifying both correctly recognized gestures and misclassifications [48]. The matrix outlines the frequency of correctly and incorrectly predicted instances for each category, encompassing all forms of true and false predictions [49]. All evaluation results were obtained using built-in analytical utilities available in Scikit-learn, specifically those for classification summary and confusion analysis. By integrating both numerical evaluation and visual inspection, this framework enables a more comprehensive understanding of how the model behaves in real classification scenarios, particularly for gestures with similar visual characteristics such as alphabet signs sharing comparable hand configurations that are often misinterpreted. Overall, this evaluation approach offers a comprehensive understanding of the model's capabilities, limitations, and its ability to generalize across diverse signer characteristics and motion patterns.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

The overall system pipeline of the proposed BISINDO recognition framework is depicted in Figure 6. This figure presents an overview of the entire workflow, starting from data acquisition through video recording, followed by feature extraction using the MediaPipe Holistic frameworkThe extracted data are then subjected to several preprocessing steps and organized into sequential structures prior to being utilized in the training and validation phases of the stacked LSTM model. In the final stage, the system's effectiveness is assessed through quantitative evaluation measures, including overall accuracy and a confusion matrix representation.

This schematic representation highlights how spatial and temporal components are integrated across each stage to form an end-to-end recognition pipeline from raw video input to final alphabet-level gesture classification output.

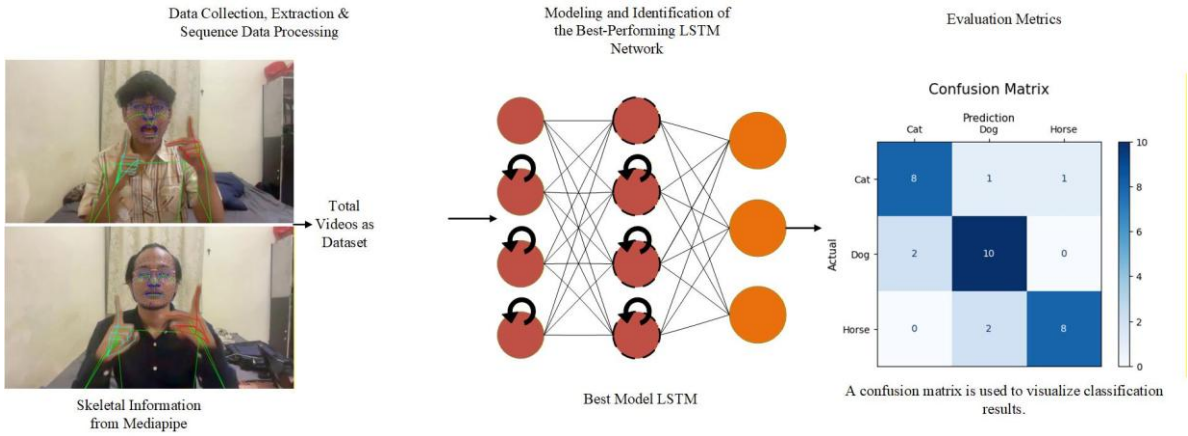


Figure 6. Pipeline of the proposed BISINDO recognition system.

4. RESULTS

4.1. Experimental Results

Training and validation accuracy across epochs showed significant improvement, demonstrating stable convergence and effective learning behavior. The training process was configured for up to 400 epochs, with an early stopping criterion applied after 50 consecutive epochs without improvement. Training automatically stopped at epoch 112 when the validation accuracy plateaued. These results indicate that the model successfully captured BISINDO gesture variations while maintaining robustness against unseen data. Early stopping effectively prevented overfitting, as shown in Figure 7.

```
22/22 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step - accuracy: 0.9591 - loss: 0.2046
Epoch 109: val_loss did not improve from 0.27226
22/22 ━━━━━━━━━━━━━━━━━━━━ 1s 30ms/step - accuracy: 0.9593 - loss: 0.2044
- val_accuracy: 0.9310 - val_loss: 0.2750 - learning_rate: 4.0960e-13
Epoch 110/400
22/22 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step - accuracy: 0.9617 - loss: 0.2219
Epoch 110: val_loss did not improve from 0.27226
22/22 ━━━━━━━━━━━━━━━━━━━━ 1s 29ms/step - accuracy: 0.9619 - loss: 0.2213
- val_accuracy: 0.9310 - val_loss: 0.2750 - learning_rate: 4.0960e-13
Epoch 111/400
22/22 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step - accuracy: 0.9456 - loss: 0.2342
Epoch 111: val_loss did not improve from 0.27226
22/22 ━━━━━━━━━━━━━━━━━━━━ 1s 30ms/step - accuracy: 0.9462 - loss: 0.2331
- val_accuracy: 0.9310 - val_loss: 0.2751 - learning_rate: 4.0960e-13
Epoch 112/400
21/22 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step - accuracy: 0.9664 - loss: 0.2119
Epoch 112: val_loss did not improve from 0.27226
22/22 ━━━━━━━━━━━━━━━━━━━━ 1s 30ms/step - accuracy: 0.9663 - loss: 0.2120
- val_accuracy: 0.9310 - val_loss: 0.2749 - learning_rate: 4.0960e-13
```

Figure 7. Training stopped at epoch 112, with accuracy reaching 93%.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

During training, the loss curves exhibited a steady decrease in training loss, while the validation loss fluctuated slightly before stabilizing toward the end, as shown in Figure 8. This demonstrates that the LSTM network achieved approximately 93% validation accuracy, confirming its ability to model the temporal dependencies within BISINDO gesture sequences effectively.

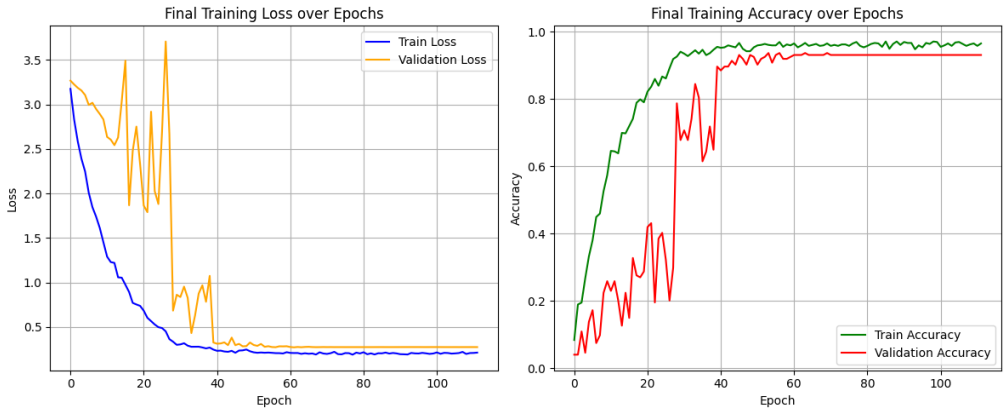


Figure 8. Training and Validation Loss Through Training

Figure 8 presents the progressive reduction of training and validation loss values, confirming efficient model optimization without major fluctuations. The plateauing of validation loss near the end aligns with early stopping activation, ensuring convergence at an optimal point. The relatively small gap between training and validation accuracy further demonstrates good generalization capability, suggesting reliable performance on unseen BISINDO gestures. The results indicate that:

- Smooth convergence in the loss curves indicates proper model optimization without abrupt overfitting
- The early stopping mechanism successfully halted training at the optimal point, ensuring model stability.
- Achieving a final validation accuracy of 93% demonstrates the model's dependable performance in practical BISINDO recognition scenarios.

To further validate the model's robustness a three-fold cross-validation strategy was implemented on the training portion of the dataset. In each iteration, one fold functioned as the validation subset, while the remaining two folds were utilized for model training, ensuring that every sample participated in the evaluation process. The model achieved a mean validation accuracy of 87.72% with a standard deviation of ±1.73%, as summarized in Table 6 and visualized in Figure 9. The small degree of variation among folds indicates stable and reliable performance across multiple data partitions, suggesting that the model's accuracy was not biased toward any specific train–test configuration.

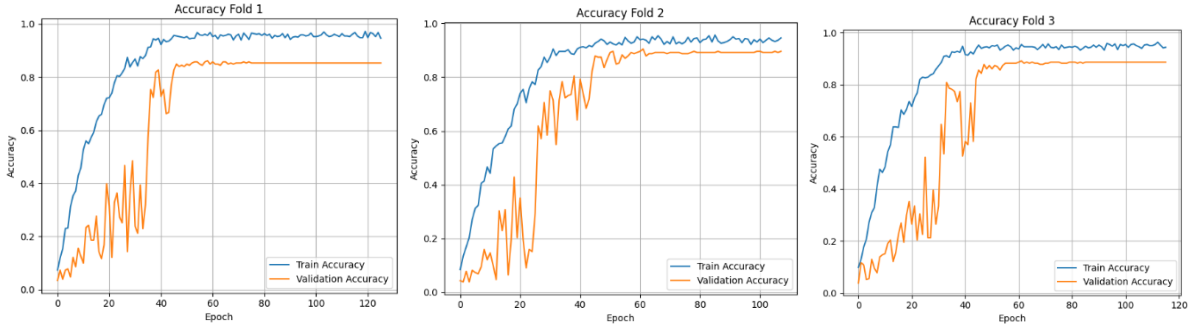


Figure 9. Three-fold cross-validation accuracy

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Table 6. Summary Fold Dataset Performance

Fold	Loss	Accuracy
1	0.47137969732284546	0.8528138399124146
2	0.41434383392333984	0.8917748928070068
3	0.39189937710762024	0.886956512928009
Mean Accuracy	0.8771817485491434	
Std Dev Accuracy	0.017342633876482137	

The small deviation (~1%) confirms that the model's generalization capability is consistent and not sensitive to the specific data configuration. This procedure enhances reliability by providing a more robust estimate of real-world generalization performance. In addition to validation consistency, several architecture configurations were evaluated to determine the most effective structure for BISINDO gesture recognition. Both a 2-layer and a 3-layer stacked LSTM model were trained under identical settings for fair comparison. As presented in Table 7, the 3-layer model achieved slightly higher performance across all evaluation metrics, reaching 93.1% accuracy, 94.2% precision, 93.88% recall, and 93.47% F1-score, outperforming the 2-layer model which achieved 91.95% accuracy. The smoother convergence pattern of the 3-layer network, illustrated in Figures 8 and 10, suggests improved learning stability and enhanced capacity for capturing long-term dependencies between sequential gestures.

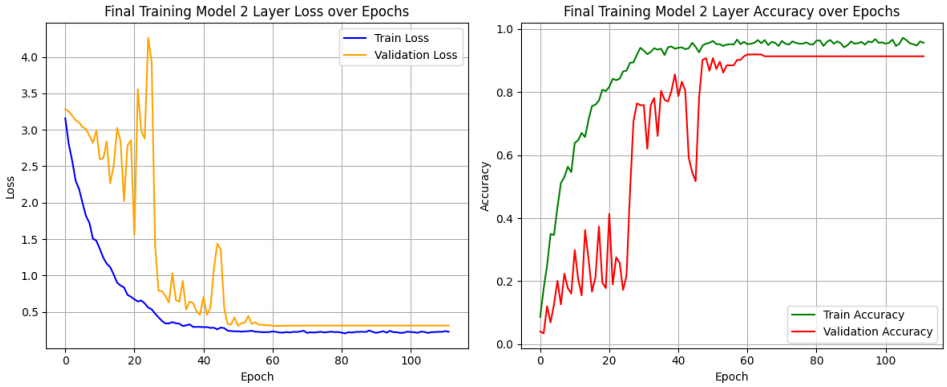


Figure 10. Training and Validation Loss Through Training in 2-Layer LSTM

Table 7. Performance Comparison Between Architectures

Architecture	Accuracy	Precission	Recall	F1-Score
2 Layer LSTM	93.1%	94.2%	93.88%	93.47%
3 Layer LSTM	91.95%	93.71%	92.78%	92.70%

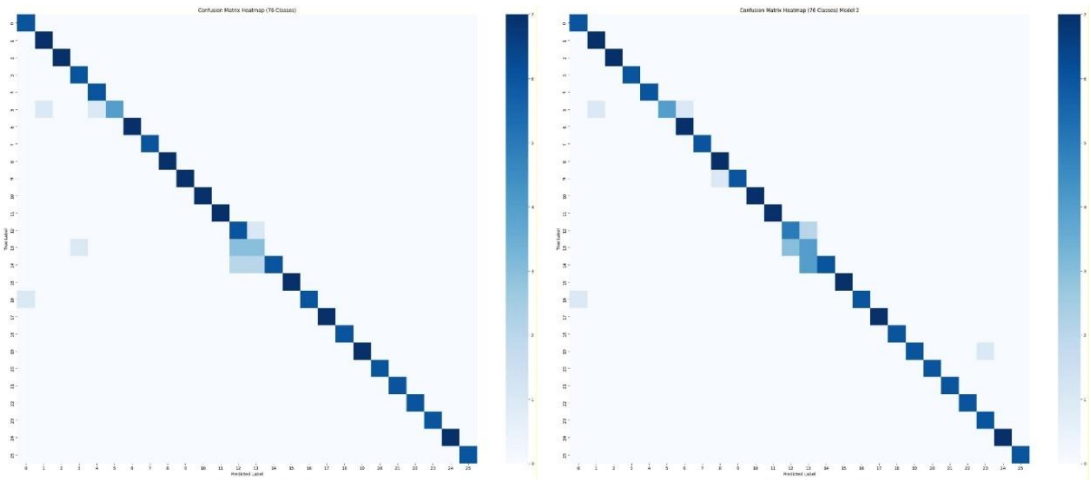
While deeper networks typically increase computational cost, the 3-layer configuration remained computationally efficient. It contained approximately 20.18 million parameters (~77 MB total), with 6.7 million trainable parameters, and required only 0.847 seconds per epoch during training. Inference time averaged 87.76 milliseconds per 50 samples, which is faster than the 2-layer model. As summarized in Table 8, these findings highlight that the three-layer model achieves an optimal balance between accuracy and computational cost, rendering it well-suited for real-time BISINDO recognition tasks.

To further assess model performance across classes, confusion matrices and classification reports were analyzed for both architectures. As illustrated in Figure 11 and detailed in Table 9, most blue intensity in the confusion matrix is concentrated along the main diagonal, indicating that the majority of predictions correctly matched the true labels (true positives). High diagonal concentration reflects strong class-level precision for gestures such as A, B, and C, which are visually distinct and well captured by the LSTM network.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Table 8. Computational Performance Comparison Between Architectures

Architecture	Total params (M)	Trainable Params (M)	Training Time (s/epoch)	Inference Time (ms/sample)
2 Layer LSTM	20,178,880	6,726,126	0.847	87.765
3 Layer LSTM	16,214,080	5,404,526	0.83	85.93



(a) 3 layer LSTM (b) 2 layer LSTM

Figure 11. Confusion Matrix Between Architectures

However, several misclassifications are also evident, represented by lighter blue regions outside the diagonal. These correspond to false positives and false negatives, particularly between letters such as M and N, as well as 12 and 13 within the binary class, which share highly similar hand shapes and finger orientations in BISINDO. Such overlap often leads to temporal ambiguity when hand movement transitions occur too quickly or the pose estimation is imperfect. Additionally, lower recall for certain classes 12 and 13 can be attributed to dataset imbalance and limited training samples, which reduce feature diversity and cause the model to generalize less effectively. The combined effects of visual similarity, skeletal jitter from MediaPipe detection, and uneven class distribution highlight areas for further improvement. Addressing these issues through targeted data augmentation, class-weighting strategies, and more balanced data collection would likely enhance recognition accuracy for underrepresented gestures.

Both architectures demonstrate strong recognition performance across most BISINDO gesture classes, though the 3-layer LSTM consistently achieved slightly higher recall and F1-scores for complex gesture transitions. The residual misclassifications are primarily associated with class imbalance in the dataset categories with fewer training examples exhibited lower recall due to limited feature representation during training. Addressing this issue could involve data augmentation, class-weighting, or transfer learning strategies to further enhance recognition accuracy for underrepresented gestures.

Overall, the findings demonstrate that the proposed three-layer stacked LSTM architecture effectively captures spatiotemporal dependencies within BISINDO gesture sequences, achieving a validation accuracy of 93% and demonstrating strong generalization and real-time efficiency. The integration of quantitative evaluations (cross-validation and performance metrics) with qualitative analysis (confusion matrix) offers a comprehensive assessment of the model's robustness, stability, and readiness for potential real-world deployment. In comparison with previous BISINDO recognition studies that utilized static image datasets, such as those by Indra et al. [15], Dwijayanti et al. [17], and Gani et al. [21], which reported accuracies ranging from 95% to 98%, the proposed sequential skeletal-

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

based LSTM model demonstrates superior adaptability and robustness in dynamic gesture scenarios. Unlike static image classifiers that rely solely on spatial cues, the proposed model effectively captures temporal dependencies and motion transitions, enabling consistent real-time recognition performance with 93% accuracy at 30 FPS. These results confirm that the proposed approach offers stronger generalization for real-world BISINDO interpretation, particularly in continuous and signer-independent settings.

Table 9. Classification Report Between Architectures

Class	Precision 3 Layer	Precision 2 Layer	Recall 3 Layer	Recall 2 Layer	F1-Score 3 Layer	F1-Score 2 Layer	Support (total videos each class in valid set)
0	0.86	0.86	1.00	1.00	0.92	0.92	6
1	0.88	0.88	1.00	1.00	0.93	0.93	7
2	1.00	1.00	1.00	1.00	1.00	1.00	7
3	0.86	1.00	1.00	1.00	0.92	1.00	6
4	0.86	1.00	1.00	1.00	0.92	1.00	6
5	1.00	1.00	0.67	0.67	0.80	0.80	6
6	1.00	0.88	1.00	1.00	1.00	0.93	7
7	1.00	1.00	1.00	1.00	1.00	1.00	6
8	1.00	0.88	1.00	1.00	1.00	0.93	7
9	1.00	1.00	1.00	0.86	1.00	0.92	7
10	1.00	1.00	1.00	1.00	1.00	1.00	7
11	1.00	1.00	1.00	1.00	1.00	1.00	7
12	0.55	0.62	0.86	0.71	0.67	0.67	7
13	0.50	0.40	0.43	0.57	0.46	0.47	7
14	1.00	1.00	0.60	0.60	0.75	0.75	10
15	1.00	1.00	1.00	1.00	1.00	1.00	7
16	1.00	1.00	0.86	0.86	0.92	0.92	7
17	1.00	1.00	1.00	1.00	1.00	1.00	7
18	1.00	1.00	1.00	1.00	1.00	1.00	6
19	1.00	1.00	1.00	0.86	1.00	0.92	7
20	1.00	1.00	1.00	1.00	1.00	1.00	6
21	1.00	1.00	1.00	1.00	1.00	1.00	6
22	1.00	1.00	1.00	1.00	1.00	1.00	6
23	1.00	0.86	1.00	1.00	1.00	0.92	6
24	1.00	1.00	1.00	1.00	1.00	1.00	7
25	1.00	1.00	1.00	1.00	1.00	1.00	6
Total Support							174

4.2. Real Time Testing

To assess the real-world usability of the proposed BISINDO recognition model, real-time experiments were carried out using live video captured through a regular laptop webcam, as presented in Figure 12. The experiment employed the best-performing LSTM model (obtained from the validation phase) deployed in a local Python environment. The MediaPipe Holistic framework was continuously activated during live operation to extract 3D skeletal landmarks from each video frame in real time. For each prediction cycle, the system processed a continuous stream of 30 frames (approximately one second of motion), which were then fed into the trained LSTM model to generate instantaneous alphabet predictions. This setup enabled real-time recognition while maintaining smooth frame capture and consistent model responsiveness.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Real-time inference was performed on a laptop equipped with a mid-range GPU, which demonstrated stable performance throughout the testing process. The use of TensorFlow's GPU acceleration and efficient data streaming allowed the system to maintain an average frame rate of approximately 30 FPS, ensuring that gesture sequences were captured and classified without perceptible delay. The overall real-time testing configuration and workflow are illustrated in Figure 12.

Figure 12. Real-time BISINDO recognition testing using webcam-based input.

During live operation, the system effectively identified a wide range of BISINDO alphabet gestures with notable precision and responsiveness. Recognition accuracies exceeded 93% for letters A and Z, demonstrating strong robustness of the trained model under non-controlled conditions. For letter X, the system achieved 84% accuracy, which was still consistent across multiple trials. The minor performance difference primarily resulted from intentional system optimization for real-time responsiveness, where continuous prediction and frame streaming were prioritized over batch processing. This trade-off ensured minimal latency and stable interactive performance.

Overall, the real-time evaluation confirmed that combining MediaPipe Holistic for spatial feature extraction with an LSTM-based temporal modeling approach enables efficient system performance, even outside controlled laboratory conditions. The framework consistently delivered reliable results under varying lighting environments, camera perspectives, and signer diversity, demonstrating strong adaptability to real-world contexts. These findings reinforce the model's robustness and highlight its potential for future expansion into continuous sign recognition, where full phrases or sentences are dynamically interpreted rather than isolated alphabet gestures. In conclusionthe findings from this live testing validate the practicality of deploying the proposed BISINDO recognition framework in real-world use cases such as assistive communication for the deaf community, learning platforms, and inclusive public service technologies.

5. DISCUSSIONS

5.1. Impact of Dataset Quality

The performance and generalization ability of deep learning models are highly dependent on the quality and diversity of the data used for training. A total of 886 videos with 30 frames per sequence were used in this study. These videos represented 26 alphabet gestures and were recorded by three signers using a standard webcam. While the dataset provided a solid foundation for training, it lacked sufficient variation in lighting conditions, background settings, and signer characteristics. Such limitations may lead to occasional misclassifications particularly among gestures with similar visual structures and restrict the model's capacity to generalize effectively to unfamiliar or more dynamic environments.

Data diversity and size play a particularly important role in SLR tasks. Larger datasets with varied signers, environments, and recording conditions help models learn more robust spatial and temporal representations, thereby reducing overfitting and improving resilience to noise such as motion blur or

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

partial occlusion [50]. Moreover, datasets that capture a wide demographic range of signing styles are essential to ensure inclusivity in capturing different ways of signing and fairness in recognition performance.

In particular, certain BISINDO gestures such as M, and N, or 12 and 13 in binary class were found to be more difficult to classify accurately. These gestures share highly similar hand shapes and finger configurations, leading to overlapping skeletal landmark patterns. Moreover, since their temporal movements are minimal, the LSTM receives fewer dynamic cues to distinguish them effectively, resulting in confusion between classes with highly similar temporal-spatial trajectories. This finding aligns with prior studies such as [22][25], which noted that fine-grained hand articulation remains one of the most persistent challenges in SLR. Addressing this issue will require either higher-resolution temporal sampling or multimodal fusion to better capture subtle differences in hand motion.

In this context, while the present dataset enabled the development of a functional recognition system with promising accuracy, expanding its scope would likely result in substantial performance gains. Incorporating more samples per gesture class, balanced class distributions, and diverse real-world conditions would not only enhance accuracy but also improve reliability in real-time applications. Ultimately, the dataset serves as the backbone of the model's success, and its continued expansion and refinement represent a key direction for advancing BISINDO recognition research.

5.2. Ablation Study on LSTM Layers

The comparison between different LSTM configurations revealed how model depth and sequence learning capacity influence performance and stability. Both 2-layer and 3-layer architectures were trained using identical hyperparameters, allowing for a fair assessment of structural differences. Results showed that the three-layer LSTM consistently produced superior accuracy and F1-score values compared to its two-layer counterpart. The addition of an extra recurrent layer enhanced the model's ability to capture extended temporal relationships across sequential gesture data. This deeper configuration enabled smoother convergence and better generalization when evaluated through 3-fold cross-validation, where mean validation accuracy reached 87.72% with low variance, indicating that the model's effectiveness was not biased toward any specific data partition.

However, the improvement in recognition accuracy between the two configurations was moderate, and further increasing the depth beyond three layers did not yield significant benefits. Instead, it introduced risks of overfitting, as reflected by fluctuations in validation loss observed during preliminary trials. This was also reflected in the training curves, where the shallower network converged faster but exhibited slightly higher variance across epochs. This suggests that optimal temporal abstraction can be achieved with a moderate number of layers, where representational power remains balanced with the dataset scale. These findings are consistent with previous studies [28][30], which noted similar diminishing returns in gesture recognition tasks when network depth exceeds the effective temporal complexity of the input data.

From a computational standpoint, the 3-layer LSTM maintained efficient training and inference times, as presented in Table 8. GPU acceleration through TensorFlow allowed the deeper architecture to operate in near real time with minimal latency. This demonstrates that the proposed configuration successfully combines higher representational capacity with practical computational feasibility. Overall, the results indicate that the 3-layer stacked LSTM efficiency, making it a structure well suited for real-time BISINDO gesture recognition under resource-constrained environments.

5.3. Advantages of the Approach

The proposed approach demonstrates several distinct advantages that address common challenges in SLR research. Unlike many vision-only methods in previous recognition studies that rely heavily on

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

raw pixel intensities or handcrafted features, the integration of MediaPipe Holistic provides a lightweight yet highly descriptive skeletal representation of hand, face, and body landmarks. This significantly reduces computational complexity while retaining critical spatial information, allowing the system to operate efficiently in real-time scenarios.

Furthermore, employing a multi-layer LSTM architecture strengthens temporal modeling capabilities, allowing the system to effectively capture both short- and long-term dependencies within gesture sequences. This design mitigates limitations often encountered in models that focus primarily on static gesture recognition or lack sufficient temporal depth. The carefully balanced layer configuration, combined with dropout and batch normalization, ensures stable training and prevents overfitting even when applied to relatively modest datasets.

Another advantage lies in the training optimization strategy, including early stopping and adaptive learning rate scheduling, which improves convergence and generalization. These strategies allow the model to achieve high accuracy despite dataset limitations such as class imbalance and limited signer diversity. Finally, the modular and scalable nature of the framework makes it adaptable to larger and more diverse datasets, as well as extensible to additional gesture classes without extensive architectural modifications. This adaptability is particularly valuable for underrepresented sign languages such as BISINDO, where resources remain scarce but the need for inclusive communication technologies is urgent.

From an academic perspective, this study reinforces the theoretical validity of skeletal feature–based temporal modeling as a robust and efficient alternative to conventional image-based recognition. Similar to findings reported by [51][27], it empirically demonstrates that compact spatiotemporal representations derived from body landmarks can achieve comparable accuracy to image-intensive models while greatly improving computational efficiency and interpretability, thereby contributing to the broader development of multimodal sign recognition systems.

5.4. Scientific Implications for Informatics

The findings of this study extend beyond an application-level contribution and carry several scientific implications for informatics research. First, the results empirically validate that skeleton-based temporal modeling can serve as an efficient and robust alternative to image-intensive recognition pipelines. By transforming raw video into compact landmark sequences, the approach reduces input dimensionality while preserving discriminative spatiotemporal cues, which decreases computational load and improves interpretability for downstream analysis [50]. This has theoretical relevance for representation learning: it supports the view that appropriately engineered geometric features can allow sequence models (e.g., LSTM, Transformer) to learn temporal abstractions with fewer parameters and lower sample complexity compared to pixel-domain models.

Second, from a methodological perspective, the study demonstrates how combining a lightweight preprocessing stage (MediaPipe Holistic) with temporal neural architectures yields a practical trade-off between accuracy, latency, and model size. This trade-off is central to informatics concerns about deploying AI within resource-constrained environments (mobile, edge devices). The empirical evidence here suggests a pathway for designing multimodal pipelines where geometric and appearance cues are selectively fused depending on use-case constraints (e.g., latency-critical vs. accuracy-critical tasks).

Third, the work provides actionable insight for the design of inclusive AI systems. By focusing on BISINDO, an underrepresented sign language the research highlights the importance of domain-specific datasets and evaluation procedures in informatics research that aims for fairness and real-world impact. The skeleton-first approach supports fairness objectives by limiting dependence on background

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

and lighting differences that often disproportionately impact marginalized groups in vision systems, thereby advancing fairness and inclusivity within AI-driven multimodal informatics.

5.5. Limitations and Future Works

The results demonstrate that skeletal data improves robustness against environmental variations, enabling the recognition of hand gestures within temporal sequences. With high classification performance for certain gestures, the trained model achieved a validation accuracy of approximately 93%. Despite achieving high accuracy in recognizing the BISINDO alphabet signs, several limitations must be acknowledged. One significant constraint is the dataset's diversity, as the current model relies on a relatively limited number of contributors and signers.

This issue may cause overfitting, where the model performs well on training data but struggles with variations in gesture style, lighting, or background. Real-time evaluation confirmed its practical capability, achieving over 90% accuracy in recognizing BISINDO alphabet gestures under near real-time conditions. However, motion blur and partial occlusions remain common challenges in sign language recognition. Although the LSTM-based model effectively captures temporal features, it may not fully represent subtle or complex motions. The fixed input length of 30 frames also limits flexibility for gestures with different temporal durations, affecting overall generalization.

Consequently, future work should focus on improving dataset diversity by incorporating videos from a wider range of speakers and recording environments. Expanding the dataset to include more regional variations on BISINDO could enhance the model's robustness. Moreover, incorporating multimodal inputs, such as skeletal tracking and audio features, could improve recognition by complementing visual cues with context. Other promising directions include exploring other model architectures like Transformer and Bi-LSTM, both of which have shown impressive sequential data processing performance. Adding these models to the model would enhance its ability to capture complex dependencies between signs while still maintaining efficiency. Lastly, ensuring accessibility and usability for the deaf community requires practical considerations such as inference optimization and implementation in assistive technologies. Future extensions in these directions would strengthen the scientific and societal value of BISINDO recognition, enabling the development of intelligent, inclusive, and accessible AI-driven communication systems.

6. CONCLUSION

This research introduced and assessed a deep learning–driven BISINDO recognition framework that combines Long Short-Term Memory (LSTM) networks with the MediaPipe Holistic pipeline to capture both spatial and temporal aspects of gesture motion. The proposed architecture successfully identified 26 BISINDO alphabet signs, reaching a validation accuracy of 93% and achieving over 93% recognition accuracy in real-time experiments for multiple gesture categories. These outcomes demonstrate the model's effectiveness in learning temporal dynamics and its reliability when deployed under practical, real-world conditions.

Beyond its empirical performance, this research contributes academically by extending the application of LSTM-based temporal modeling within computer vision and human–computer interaction domains specifically for low-resource sign languages such as BISINDO. The findings demonstrate that skeletal-based temporal representations can serve as an efficient and interpretable alternative to pixel-based recognition pipelines, offering a promising direction for multimodal sign language research.

Several challenges remain, including class imbalance, visual similarity between certain gestures, and occasional performance drops under real-time conditions such as occlusion or motion blur.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Addressing these will require larger, more diverse datasets and advanced sequence models such as Bi-LSTM or Transformer architectures.

Looking ahead, this framework provides a strong foundation for scalable and inclusive sign language technologies. Future work may focus on extending recognition beyond isolated alphabets to continuous signing, integrating multimodal cues such as facial expressions and speech, and optimizing inference for deployment on mobile and embedded systems. In essence, this study not only advances the technical understanding of skeletal-based spatiotemporal learning but also contributes toward more accessible AI-driven communication tools for the Indonesian deaf community.

CONFLICT OF INTEREST

The authors declare that there are no conflicts of interest among the authors or with the research object presented in this paper.

ACKNOWLEDGEMENT

The author expresses deep gratitude to all parties who have contributed to the completion of this study.

REFERENCES

[1] R. Yunita, E. B. Nababan, and M. S. Lydia, "Indonesian Dynamic Sign Language Recognition for Individuals with Sensory Disabilities using LSTM," in *2024 4th International Conference of Science and Information Technology in Smart Administration (ICSINTESA)*, Balikpapan, Indonesia: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 417–420. doi: 10.1109/ICSINTESA62455.2024.10748114.

[2] S. Baghavathi Priya, P. V. R. Subba Rao, and T. S. Madeswaran, "Enhancing Sign Language Recognition: A CNN-BiLSTM Approach for Accurate Gesture Interpretation," in *2023 International Conference on Next Generation Electronics (NEleX)*, Vellore, India: Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/NEleX59773.2023.10421113.

[3] R. Rastgoo, K. Kiani, and S. Escalera, "Sign Language Recognition: A Deep Survey," *Expert Syst Appl*, vol. 164, p. 113794, Feb. 2021, doi: 10.1016/j.eswa.2020.113794.

[4] M. H. Ismail, S. A. Dawwd, and F. H. Ali, "Arabic Sign Language Detection Using Deep Learning Based Pose Estimation," in *2021 2nd Information Technology To Enhance e-learning and Other Application (IT-ELA)*, Baghdad, Iraq: Institute of Electrical and Electronics Engineers Inc., 2021, pp. 161–166. doi: 10.1109/IT-ELA52201.2021.9773404.

[5] N. F. Attia, M. T. F. S. Ahmed, and M. A. M. Alshewimy, "Efficient deep learning models based on tension techniques for sign language recognition," *Intelligent Systems with Applications*, vol. 20, p. 200284, Nov. 2023, doi: 10.1016/j.iswa.2023.200284.

[6] S. Shinde, P. Mahalle, S. Panchal, S. Mahalle, A. Pandit, and P. Tonpe, "Sign language recognition using deep learning," in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kamand, India: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1–5. doi: 10.1109/ICCCNT61001.2024.10725481.

[7] Y. Zhang and X. Jiang, "Recent Advances on Deep Learning for Sign Language Recognition," *Computer Modeling in Engineering & Sciences*, vol. 139, no. 3, pp. 2399–2450, Mar. 2024, doi: 10.32604/cmes.2023.045731.

[8] R. Alzohairi, R. Alghonaim, W. Alshehri, S. Aloqeely, M. Alzaidan, and O. Bchir, "Image based Arabic Sign Language Recognition System," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 185–194, 2018, doi: 10.14569/IJACSA.2018.090327.

[9] D. G. Enikeev and S. A. Mustafina, "Sign language recognition through Leap Motion controller and input prediction algorithm," *J Phys Conf Ser*, vol. 1715, no. 1, p. 012008, Jan. 2021, doi: 10.1088/1742-6596/1715/1/012008.

[10] K. Pattanaworapan, K. Chamnongthai, and J.-M. Guo, "Hand gesture recognition using codebook model and Pixel-Based Hierarchical-Feature Adaboosting," in *2013 13th International*

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

Symposium on Communications and Information Technologies (ISCIT), Surat Thani, Thailand, 2013, pp. 544–548. doi: 10.1109/ISCIT.2013.6645918.

[11] Z. J. Liang, S. Bin Liao, and B. Z. Hu, "3D convolutional neural networks for dynamic sign language recognition," *Comput J*, vol. 61, no. 11, pp. 1724–1736, Nov. 2018, doi: 10.1093/comjnl/bxy049.

[12] W. Hao, C. Hou, Z. Zhang, X. Zhai, L. Wang, and G. Lv, "A sensing data and deep learning-based sign language recognition approach," *Computers and Electrical Engineering*, vol. 118, p. 109339, Aug. 2024, doi: 10.1016/j.compeleceng.2024.109339.

[13] I. D. Mienye, T. G. Swart, and G. Obaido, "Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications," *Information*, vol. 15, no. 9, p. 517, Aug. 2024, doi: 10.3390/info15090517.

[14] M. De Coster, P. Rabaey, S. Verlinden, M. Van Herreweghe, and J. Dambre, "Frozen Pretrained Transformers for Neural Sign Language Translation," in *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, Association for Machine Translation in the Americas, 2021, pp. 88–97.

[15] D. Indra, Purnawansyah, S. Madenda, and E. P. Wibowo, "Indonesian sign language recognition based on shape of hand gesture," in *Procedia Computer Science*, Surabaya, Indonesia: Elsevier B.V., Jul. 2019, pp. 74–81. doi: 10.1016/j.procs.2019.11.101.

[16] I. P. Sari, "Closer Look at Image Classification for Indonesian Sign Language with Few-Shot Learning Using Matching Network Approach," *International Journal on Informatics Visualization*, vol. 7, no. 3, pp. 638–643, Sep. 2023, doi: 10.30630/joiv.7.3.1320.

[17] S. Dwijayanti, S. Inas Taqiyyah, H. Hikmarika, and B. Yudho Suprapto, "Indonesia Sign Language Recognition using Convolutional Neural Network," *Int J Adv Comput Sci Appl*, vol. 12, no. 10, pp. 415–422, 2021, doi: 10.14569/IJACSA.2021.0121046.

[18] Sutarman, M. A. Majid, and J. M. Zain, "A review on the development of Indonesian sign language recognition system," *Journal of Computer Science*, vol. 9, no. 11, pp. 1496–1505, 2013, doi: 10.3844/jcssp.2013.1496.1505.

[19] I Dewa Made Bayu Atmaja Darmawan, Linawati, G. Sukadarmika, N. M. A. E. D. Wirastuti, and R. Pulungan, "Temporal Action Segmentation in Sign Language System for Bahasa Indonesia (SIBI) Videos Using Optical Flow-Based Approach," *Jurnal Ilmu Komputer dan Informasi*, vol. 17, no. 2, pp. 195–202, Jun. 2024, doi: 10.21609/jiki.v17i2.1284.

[20] A. R. M. Oropesa, G. L. R. Felicen, and J. A. De Guzman, "SENYAS: A Filipino Sign Language Recognition System Using MediaPipe and CNN-LSTM," in *TENCON 2024 - 2024 IEEE Region 10 Conference (TENCON)*, Singapore, Singapore: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 956–960. doi: 10.1109/TENCON61640.2024.10902785.

[21] R. A. Gani and T. A. Budi Wirayuda, "Recognizing Indonesian Sign Language (BISINDO) Alphabet Using Optimized Deep Learning," in *ICADEIS 2025 - 2025 International Conference on Advancement in Data Science, E-learning and Information System: Integrating Data Science and Information System, Proceeding*, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/ICADEIS65852.2025.10933226.

[22] R. M. Abdulhamied, M. M. Nasr, and S. N. Abdulkader, "Real-time recognition of American sign language using long-short term memory neural network and hand detection," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 1, pp. 545–556, Apr. 2023, doi: 10.11591/ijeecs.v30.i1.pp545-556.

[23] X. Chen *et al.*, "The importance of short lag-time in the runoff forecasting model based on long short-term memory," *J Hydrol (Amst)*, vol. 589, p. 125359, Oct. 2020, doi: 10.1016/j.jhydrol.2020.125359.

[24] L. Yongyi, L. Cewu, and T. Chi Keung, "Online Video Object Detection Using Association LSTM," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, Dec. 2017, pp. 2363–2371. doi: 10.1109/ICCV.2017.257.

[25] K. B. Tran, U. D. Nguyen, and Q. T. Huynh, "Continuous Sign Language Recognition Using MediaPipe," in *2023 International Conference on Advanced Technologies for Communications (ATC)*, Da Nang, Vietnam: IEEE Computer Society, 2023, pp. 493–498. doi: 10.1109/ATC58710.2023.10318855.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

[26] T. J. Sánchez-Vicinaiz, E. Camacho-Pérez, A. A. Castillo-Atoche, M. Cruz-Fernandez, J. R. García-Martínez, and J. Rodríguez-Reséndiz, "MediaPipe Frame and Convolutional Neural Networks-Based Fingerspelling Detection in Mexican Sign Language," *Technologies (Basel)*, vol. 12, no. 8, p. 124, Aug. 2024, doi: 10.3390/technologies12080124.

[27] Y. Farhan and A. Ait Madi, "Real-time Dynamic Sign Recognition using MediaPipe," in *2022 IEEE 3rd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, Fez, Morocco: Institute of Electrical and Electronics Engineers Inc., 2022, p. 1. doi: 10.1109/ICECOCS55148.2022.9982822.

[28] A. Tripathi, S. Makhloga, S. Singh, S. Semwal, and V. Tomar, "SLRMPCMC: Sign Language Recognition using Mediapipe and Cross-model Comparison," in *2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT)*, Greater Noida, India: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1–6. doi: 10.1109/ICEECT61758.2024.10738932.

[29] L. A. Ranee, F. L. Marshillong, S. A. Lyndoh, and A. K. Maji, "Khasi Sign Language Recognition using Google's Mediapipe and Deep Learning Feedforward Neural Network Approach," in *International Conference on Machine Learning and Data Engineering*, Dehradun, India: Procedia Computer Science, Elsevier B.V., Aug. 2025, pp. 3619–3629. doi: 10.1016/j.procs.2025.04.617.

[30] G. Khartheesvar, M. Kumar, A. K. Yadav, and D. Yadav, "Automatic Indian sign language recognition using MediaPipe holistic and LSTM network," *Multimed Tools Appl*, vol. 83, no. 20, pp. 58329–58348, Jun. 2024, doi: 10.1007/s11042-023-17361-y.

[31] R. Cui, H. Liu, and C. Zhang, "A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training," *IEEE Trans Multimedia*, vol. 21, no. 7, pp. 1880–1891, Jul. 2019, doi: 10.1109/TMM.2018.2889563.

[32] P. Rakshit, S. Paul, and S. Dey, "Sign language detection using convolutional neural network," *J Ambient Intell Humaniz Comput*, vol. 15, no. 4, pp. 2399–2424, Apr. 2024, doi: 10.1007/s12652-024-04761-7.

[33] M. H. Ismail, S. A. Dawwd, and F. H. Ali, "Static hand gesture recognition of Arabic sign language by using deep CNNs," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 1, pp. 178–188, Oct. 2021, doi: 10.11591/ijeecs.v24.i1.pp178-188.

[34] N. Adaloglou *et al.*, "A Comprehensive Study on Deep Learning-based Methods for Sign Language Recognition," *IEEE Trans Multimedia*, vol. 24, pp. 1750–1762, Apr. 2021, doi: 10.1109/TMM.2021.3070438.

[35] J. Bora, S. Dehingia, A. Boruah, A. A. Chetia, and D. Gogoi, "Real-time Assamese Sign Language Recognition using MediaPipe and Deep Learning," in *International Conference on Machine Learning and Data Engineering*, Dehradun, India: Procedia Computer Science, Elsevier B.V., Sep. 2022, pp. 1384–1393. doi: 10.1016/j.procs.2023.01.117.

[36] S. Srivastava, S. Singh, Pooja, and S. Prakash, "Continuous Sign Language Recognition System Using Deep Learning with MediaPipe Holistic," *Wirel Pers Commun*, vol. 137, no. 3, pp. 1455–1468, Aug. 2024, doi: 10.1007/s11277-024-11356-0.

[37] V. K. Chaitanya, M. Lolla, A. Barik, V. Kondapaneni, and O. K. Sikha, "Bharatnatyam Pose and Mudra Recognition Using MediaPipe and Deep Features," in *2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, Greater Noida, India: Institute of Electrical and Electronics Engineers Inc., 2022, pp. 635–641. doi: 10.1109/ICCCIS56430.2022.10037655.

[38] K. Navendu and V. Sahula, "Word Level Sign Language Recognition Using MediaPipe and LSTM-GRU Network," in *2024 IEEE International Symposium on Smart Electronic Systems (iSES)*, New Delhi, India: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 13–18. doi: 10.1109/iSES63344.2024.00014.

[39] M. Sankara Mahalingam, N. Suresh Kumar, C. Harika, C. Harika, C. S. Reddy, and D. P. Kalyan, "Sign to Text: Automated Sign Language Interpretation using LSTM and Computer Vision," in *2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, Pudukkottai, India: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1414–1419. doi: 10.1109/ICACRS62842.2024.10841493.

Vol. 7, No. 2, April 2026, Page. 903-925
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5337

[40] H. Yoo, I. Goncharenko, and Y. Gu, "Real-Time Dynamic Sign Language Recognition Using LSTM Based on MediaPipe Hand Data," in *2023 International Conference on Consumer Electronics - Taiwan (ICCE-Taiwan)*, PingTung, Taiwan: Institute of Electrical and Electronics Engineers Inc., 2023, pp. 17–18. doi: 10.1109/ICCE-Taiwan58799.2023.10226687.

[41] B. Sundar and T. Bagyammal, "American Sign Language Recognition for Alphabets Using MediaPipe and LSTM," in *4th International Conference on Innovative Data Communication Technology and Application*, Coimbatore, Tamil Nadu, India: Procedia Computer Science, Elsevier B.V., Nov. 2022, pp. 642–651. doi: 10.1016/j.procs.2022.12.066.

[42] D. Li, C. Rodriguez Opazo, X. Yu, and H. Li, "Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Snowmass, CO, USA: IEEE, 2020, pp. 1448–1458. doi: 10.1109/WACV45572.2020.9093512.

[43] T. Fan, L. Zheheng, and Z. Dongsheng, "A deep network based integrated model for disease named entity recognition," in *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Kansas City, MO, USA: IEEE, 2017, pp. 618–621. doi: 10.1109/BIBM.2017.8217723.

[44] J. Huang, J. Chaijaruwanich, and V. Chouvatut, "Video-based Sign Language Recognition with R(2+1)D and LSTM Networks," in *2024 16th International Conference on Knowledge and Smart Technology (KST)*, Krabi, Thailand: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 214–219. doi: 10.1109/KST61284.2024.10499646.

[45] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark," *Neurocomputing*, vol. 503, pp. 92–108, Sep. 2022, doi: 10.1016/j.neucom.2022.06.111.

[46] S. Prabu, T. K. Sridhar, S. Sridharan, D. Sukesh, and J. Rajavel, "Revolutionizing Communication: A Hybrid Deep Learning Framework for Enhanced Sign Language Recognition," in *2024 International Conference on Data Science and Network Security (ICDSNS)*, Tiptur, India: Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1–6. doi: 10.1109/ICDSNS62112.2024.10690996.

[47] M. A. As'ari, N. A. J. Sufri, and G. S. Qi, "Emergency sign language recognition from variant of convolutional neural network (CNN) and long short term memory (LSTM) models," *International Journal of Advances in Intelligent Informatics*, vol. 10, no. 1, pp. 64–78, Feb. 2024, doi: 10.26555/ijain.v10i1.1170.

[48] D. Hand and P. Christen, "A note on using the F-measure for evaluating record linkage algorithms," *Stat Comput*, vol. 28, no. 3, pp. 539–547, May 2018, doi: 10.1007/s11222-017-9746-6.

[49] P. Das, T. Ahmed, and M. F. Ali, "Static Hand Gesture Recognition for American Sign Language using Deep Convolutional Neural Network," in *2020 IEEE Region 10 Symposium (TENSYMP)*, Dhaka, Bangladesh: Institute of Electrical and Electronics Engineers Inc., Jun. 2020, pp. 1762–1765. doi: 10.1109/TENSYMP50017.2020.9230772.

[50] G. Luo, S. Yang, G. Tian, C. Yuan, W. Hu, and S. J. Maybank, "Learning human actions by combining global dynamics and local appearance," *IEEE Trans Pattern Anal Mach Intell*, vol. 36, no. 12, pp. 2466–2482, Dec. 2014, doi: 10.1109/TPAMI.2014.2329301.

[51] A. F. Alnabih and A. Y. Maghari, "Arabic sign language letters recognition using Vision Transformer," *Multimed Tools Appl*, vol. 83, pp. 81725–81739, Mar. 2024, doi: 10.1007/s11042-024-18681-3.