WQF7006 COMPUTER VISION AND IMAGE PROCESSING
GROUP ASSIGNMENT REPORT
MALAYSIAN SIGN LANGUAGE TRANSLATION CASE STUDY

OCC 1, Group 6: Khor Yin Loon, Hong Jia Herng, Cheong Yi Fong, Chee Zen Yu

Universiti Malaya

ABSTRACT

Automatic sign language recognition systems are essential for improving accessibility for the Deaf and Hard-of-Hearing community. However, most existing solutions focus on widely used sign languages such as American Sign Language, leaving localized languages such as Malaysian Sign Language (MSL) comparatively underexplored. This work presents an end-to-end deep learning framework for MSL gloss recognition based on pose and hand landmark sequences, with an emphasis on effective temporal modeling. Unlike conventional frame-level or early-frame sampling approaches, uniform temporal sampling and attention-based sequence modeling are employed to better capture the dynamic structure of MSL gestures. Three temporal architectures, namely a baseline Long Short-Term Memory (LSTM), a bidirectional LSTM with temporal attention and a Transformer encoder, are systematically evaluated under identical training conditions. Experimental results show that attention-based models outperform the baseline LSTM, with the Transformer encoder achieving a test accuracy of 98.15% and a weighted F1-score of 0.9814 across 30 MSL glosses, while converging faster and using fewer parameters than recurrent counterparts. The proposed model is further integrated into a web-based prototype supporting both real-time webcam input and offline video processing, demonstrating practical applicability for localized sign language recognition. These findings highlight the effectiveness of self-attention mechanisms for MSL recognition and provide a scalable foundation for future assistive communication systems. The source code and models are publicly available at https://github.com/hongjiaherng/wqf7006-msl.

Keywords — *Malaysian Sign Language, sign language recognition, deep learning, temporal modeling, transformer*

1. INTRODUCTION

Malaysian Sign Language (MSL), also known as Bahasa Isyarat Malaysia (BIM), serves as the primary mode of communication for the Deaf and Hard-of-Hearing community in Malaysia. As a natural sign language that has evolved within the Deaf community, MSL possesses its own grammatical structures and syntactic rules, which are distinct from those of spoken Malay or English [1]. Despite its official recognition, proficiency in MSL among the general hearing population remains limited. This linguistic gap presents a significant barrier to social inclusion, restricting the Deaf community's access to essential public services, education and employment opportunities [2], [3]. As such, bridging this communication divide is not only a technical challenge but also a critical social need.

The necessity for automated MSL translation systems is further driven by the limited availability of qualified human interpreters and the immediate nature of many daily interactions [1]. While human interpretation remains essential for complex or nuanced communication, it is often costly and impractical for routine or time-sensitive situations. Moreover, existing technological solutions predominantly focus on widely used sign languages such as American Sign Language (ASL) or International Sign Language (IS), leaving localized languages like MSL comparatively underrepresented [4]. This imbalance has resulted in a lack of accessible and practical AI-based tools for MSL, underscoring the need for real-time translation systems tailored to the Malaysian context.

To address these challenges, this project develops IsyaratAI, a web-based computer vision system designed to translate MSL gestures into text. The primary objective is to build a lightweight and accessible deep learning model capable of accurately recognizing selected BIM glosses. The scope of the project includes both the development of the recognition model and the implementation of a user-friendly prototype deployed on the Streamlit platform, intended to support everyday communication as well as learning and practice scenarios involving MSL. The system supports both pre-recorded video uploads and real-time webcam input, allowing flexibility across different usage scenarios. By providing a scalable and free-to-use translation tool, this project aims to enhance accessibility and promote more inclusive communication for the Deaf community in Malaysia.

2. TEAM ROLES

To ensure effective teamwork and consistent progress throughout the project, tasks were clearly allocated based on individual strengths while maintaining continuous communication and collaborative decision-making among all members. The team adopted a coordinated workflow in which progress was reviewed regularly, allowing members to provide feedback, resolve issues and ensure that all contributions were aligned with the project objectives and coursework milestones.

A. Project Coordination and Reporting

The overall project workflow was overseen by the project manager, Khor Yin Loon, who was responsible for planning task timelines, coordinating responsibilities and ensuring alignment with assessment requirements. This role also involved facilitating regular discussions to track progress across technical and development components. In addition, the project manager consolidated experimental findings, implementation outcomes and technical insights from all team members into a unified academic report and final presentation, ensuring consistency and clarity across all project phases.

B. Data Engineering and Model Training

The technical core of the project was led by the AI Engineering team under the primary direction of Hong Jia Herng, who served as the lead AI engineer. He was responsible for the critical data preprocessing stage and the end-to-end training of deep learning architectures. This included transforming raw MediaPipe landmarks into structured temporal tensors and systematically evaluating model performance metrics to ensure reliable classification accuracy. Supporting this core development, Cheong Yi Fong, also an AI engineer, contributed by conducting targeted experimentation on alternative model architectures and hyperparameter configurations. These experiments provided comparative insights that informed the final model selection process, enabling the team to identify the most effective neural network configuration for recognizing the 30 selected MSL glosses.

C. Software Integration and Deployment

The integration and deployment of the trained model were handled by Chee Zen Yu, who was responsible for embedding the finalized deep learning model into a functional application workflow. This role involved integrating feature extraction and inference components into a streamlined pipeline. Throughout this process, close communication with the AI Engineering team was maintained to address compatibility issues and refine model outputs, ensuring that the final prototype accurately demonstrated the practical functionality of the MSL recognition system.

3. DATA COLLECTION

A. Collaborative Data Acquisition

The dataset was developed through a collective effort involving all course participants during supervised recording sessions. MSL videos were captured under direct instructor guidance within a controlled environment to ensure consistent camera framing, lighting conditions and signer visibility. This controlled setup was essential for preserving both manual gestures and non-manual markers such as body posture. Each recorded video was annotated at the gloss level, with labels assigned based on the intended MSL sign demonstrated during recording. To ensure annotation consistency, recordings followed predefined gloss definitions provided during the sessions. Although the complete dataset comprised 90 distinct signs, the project focused on the selection of the top 30 glosses with the highest sample counts for model development, including common signs such as *beli*, *jangan*, *makan*, *perempuan*, *lelaki* and *apa_khabar*. This selection strategy ensured sufficient data representation per class, thereby improving training stability and reducing the risk of overfitting caused by data sparsity.

B. Landmark Extraction and Feature Engineering

Prior to feature extraction, the raw video data underwent a data cleaning process to remove unsuitable samples. Videos containing incomplete gestures or inconsistent signing were excluded from the dataset. During landmark extraction, each video was processed frame by frame using the MediaPipe Holistic model [5]. Frames in which pose or hand landmarks could not be reliably detected were discarded to preserve feature quality and ensure consistency across samples. For each valid frame, a comprehensive feature vector was extracted. This vector consisted of pose landmarks capturing non-manual features such as body posture and upper-body movement, as well as hand landmarks representing manual signing movements. Specifically, 33 pose landmarks were extracted with corresponding x, y and z coordinates and visibility scores, while 21 landmarks were extracted for each hand using x, y and z coordinates, resulting in a total of 258 feature dimensions per frame. By combining both manual and non-manual features, the extracted representation provides a holistic spatial description of each MSL gesture.

C. Temporal Sampling and Tensor Organization

To standardize temporal input length across all samples, a uniform temporal sampling strategy was applied. From the cleaned sequence of detected frames in each video, 30 frames were selected at equal intervals, ensuring that the complete progression of each gesture was captured while maintaining a fixed input size. Following sampling, each gesture sequence was assigned its corresponding numeric label and frame-level feature vectors were organized into ordered temporal sequences. The final dataset was structured into tensors of

shape (N, T, D), where N represents the number of samples, T = 30 denotes the number of time steps and D = 258 corresponds to the feature dimensionality. This standardized tensor format enables efficient integration into the model training pipeline while preserving the temporal dynamics essential for sign recognition.

4. MODEL TRAINING

This section describes the deep learning techniques employed to develop the MSL recognition system, including model selection rationale, architectural design, training configuration and evaluation methodology. Three temporal sequence models were implemented and systematically compared, including a baseline Long Short-Term Memory (LSTM) network, a bidirectional LSTM (Bi-LSTM) with temporal attention and a Transformer encoder. These architectures were selected to progressively explore increasingly expressive temporal modeling strategies for sign language data. All models were trained and evaluated under identical experimental conditions to ensure a fair and controlled comparison.

A. Model Architectures

Baseline LSTM: The baseline model employs a stacked unidirectional LSTM architecture to establish a reference performance level for temporal modeling. LSTMs are well-suited for sequential data due to their ability to capture long-term temporal dependencies, making them a common baseline for gesture and sign language recognition tasks [6].

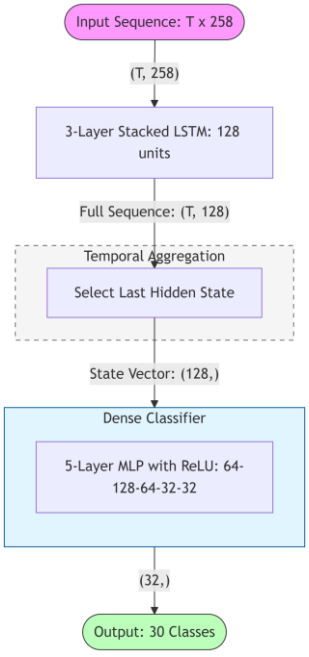

Figure 1: Baseline model featuring a 3-layer stacked LSTM. Temporal aggregation is performed via last-timestep extraction, followed by a 5-layer MLP for classification.

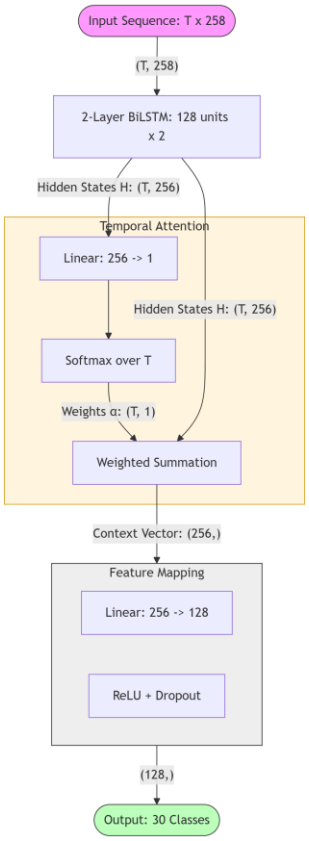

Figure 2: Bi-LSTM architecture with a temporal attention mechanism. The model computes a weighted sum of hidden states (T, 256) to generate a fixed-length context vector for the dense head.

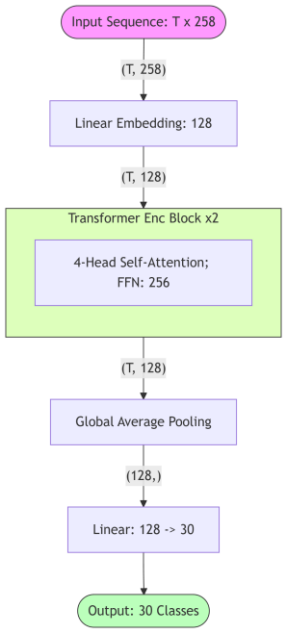

Figure 3: Transformer encoder model using 4-head self-attention. Global average pooling is applied across the temporal dimension to derive the latent representation.

The architecture consists of three stacked LSTM layers with a hidden dimension of 128. The hidden state from the final time step is used as a fixed-length representation of the input gesture sequence. This representation is passed through a fully connected classification head composed of six linear layers with ReLU activation applied after each intermediate layer, allowing the model to learn nonlinear decision boundaries. The baseline model serves as a comparative benchmark for assessing the impact of bidirectional processing and attention mechanisms.

Bi-LSTM with Temporal Attention: To enhance temporal representation learning, this model incorporates bidirectional recurrent processing together with a temporal attention mechanism. A two-layer Bi-LSTM with a hidden dimension of 128 per direction is employed to capture contextual information from both past and future frames within a gesture sequence. This design is particularly suitable for sign language recognition, where the interpretation of a gesture depends on its complete temporal context rather than unidirectional motion alone [7]. Following that, the sequence of hidden states generated by the Bi-LSTM is passed to a temporal attention module. Attention weights are computed for each time step using a linear projection followed by Softmax normalization, enabling the model to emphasize frames that contribute most significantly to the recognition of a given sign. Subsequently, the attention-weighted hidden states are aggregated into a single sequence-level representation through a weighted summation. This representation is then passed to a fully connected classification head with ReLU activation and dropout, improving model robustness and reducing overfitting.

Transformer Encoder: This model adopts a Transformer-based architecture, replacing recurrent processing with self-attention mechanisms for temporal dependency modeling. Transformer architectures are well suited for sequential data due to their ability to capture long-range temporal relationships efficiently while enabling parallel computation across time steps, which is advantageous for fixed-length gesture sequences [8]. Input landmark features are first projected into a 128-dimensional embedding space using a linear transformation. The embedded sequence is then processed by a Transformer encoder composed of two stacked encoder layers. Each encoder layer employs multi-head self-attention with four attention heads, followed by a position-wise feedforward network with an intermediate dimension of 256. Dropout is applied within the encoder layers to improve regularization during training. Following that, the resulting sequence representations are aggregated across the temporal dimension using global average pooling to produce a fixed-length feature vector. This pooled representation is subsequently passed to a linear classification layer to generate the final class logits.

The baseline LSTM, Bi-LSTM with temporal attention and Transformer encoder models contain approximately 0.5M, 0.8M and 0.3M trainable parameters respectively. The baseline LSTM establishes a reference level of performance for temporal sequence modeling, while the Bi-LSTM with temporal attention and Transformer encoder represent deliberate architectural optimizations beyond this baseline. Specifically, bidirectional processing and temporal attention enable the model to exploit full-sequence contextual information and emphasize informative frames, whereas the Transformer encoder further enhances temporal representation through self-attention mechanisms capable of modeling long-range dependencies with reduced reliance on recurrent computation. This comparison highlights different architectural approaches to balancing parameter efficiency and temporal representational capability across the evaluated models.

B. Training Configuration

All models were trained using an identical experimental configuration to ensure a fair and controlled comparison. The input to each model consisted of fixed-length sequences of 30 frames, obtained through uniform temporal sampling of the original gesture recordings. Training was performed using the Adam optimizer with a learning rate of 1×10^{-3} and categorical cross-entropy was employed as the loss function. Models were trained with a batch size of 32 for a maximum of 100 epochs.

To improve training stability, gradient clipping with a maximum norm of 1.0 was applied. Early stopping with a patience of 20 epochs and model checkpointing based on validation loss were implemented to mitigate overfitting. The dataset was split into training and validation sets in a 9:1 ratio, with validation performance used for model selection and evaluation. Model performance was evaluated using classification accuracy and weighted F1-score, providing a balanced assessment of recognition performance across all sign classes.

5. FINDINGS

This section presents and analyzes the experimental results obtained from the evaluated models, including the effect of preprocessing strategies and the comparative performance of different model architectures.

A. Effect of Preprocessing Strategy

The impact of preprocessing strategy was first evaluated by comparing two frame sampling approaches using the baseline LSTM model, selecting the first 30 valid frames from each video (*first_30*) and uniformly sampling 30 valid

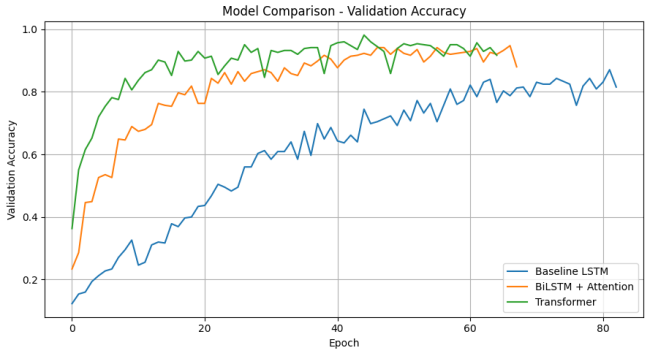

Figure 4: Validation accuracy curves across training epochs for the baseline LSTM, Bi-LSTM with attention and Transformer models using the uniform_30 preprocessing strategy.

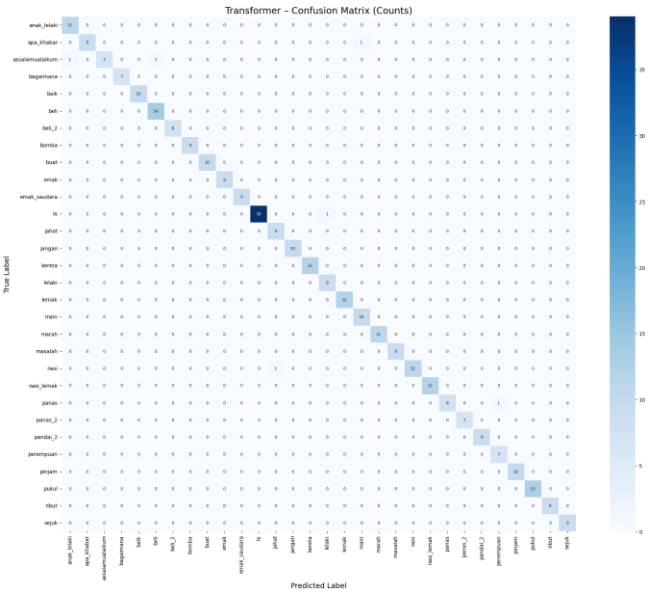

Figure 5: Confusion matrix of the Transformer model on the test set using the uniform_30 preprocessing strategy.

frames across the entire video duration (*uniform_30*). The performance comparison is reported in Table 1, which summarizes the test loss and test accuracy for both strategies.

Table 1: Test performance comparison between first 30-frame and uniform frame sampling strategies.

Frame Strategy	Test Loss	Test Accuracy
first_30	0.9803	0.7631
uniform_30	0.7391	**0.8492**

The results show that the *uniform_30* strategy achieves both lower test loss and higher test accuracy compared to selecting the first 30 frames. This indicates that uniform temporal sampling more effectively captures the overall temporal structure of MSL gestures, whereas relying solely on the initial frames may omit important motion cues that occur later in the gesture sequence, leading to reduced recognition performance. Based on this observation, the *uniform_30* preprocessing strategy was adopted for all subsequent model evaluations.

B. Overall Performance

The classification performance of the baseline LSTM, Bi-LSTM with attention and Transformer models is summarized in Table 2, which reports test accuracy and F1-score for each model.

Table 2: Test accuracy and F1-score comparison of evaluated models using the uniform_30 preprocessing strategy.

Model Architecture	Test Accuracy	Test F1-score
Baseline LSTM	0.8308	0.8247
Bi-LSTM with Attention	0.9415	0.9413
Transformer	**0.9815**	**0.9814**

Both attention-based models substantially outperform the baseline LSTM across all reported metrics, demonstrating the importance of enhanced temporal modeling for MSL recognition. The Transformer model achieves the highest test accuracy and F1-score, while the Bi-LSTM with attention attains strong but lower performance. In contrast, the baseline LSTM exhibits reduced recognition accuracy, highlighting the limitations of unidirectional recurrent modeling without explicit attention mechanisms.

In addition to final test performance, the validation accuracy trends across training epochs for all models are illustrated in Figure 4. The comparison shows that all three models converge at different rates and trigger early stopping at different epochs. The Transformer model converges the fastest, reaching its optimal validation performance and stopping at epoch 65, while the recurrent models require more epochs to stabilize. This observation suggests more efficient learning dynamics for the Transformer architecture under the same training configuration, likely due to its ability to model long-range temporal dependencies through self-attention.

C. Confusion Matrix Analysis

To analyze class-level prediction behavior, the confusion matrix of the best-performing Transformer model is examined and shown in Figure 5. The confusion matrix exhibits a strong diagonal structure, indicating that most gesture instances are correctly classified across all 30 gloss classes. Off-diagonal entries are negligible, suggesting limited confusion between different classes.

The few observed misclassifications are concentrated within a small subset of classes. For example, class 2 (*assalamualaikum*) contains two misclassified instances,

which are predicted as *anak_lelaki* and *beli*, respectively. Despite these errors, class 2 maintains a precision of 1.00, indicating that all predictions assigned to this class are correct. The reduced recall for this class reflects missed detections rather than incorrect positive predictions. Most other classes achieve perfect or near-perfect classification performance. These observations are consistent with the high overall test accuracy and weighted F1-score reported earlier, indicating that the remaining errors are limited and localized rather than systematic.

D. Class-Specific Analysis

To further investigate the remaining classification errors, a focused analysis was conducted on class 2 (*assalamualaikum*), which achieves the lowest F1-score among all evaluated classes. According to the classification report, this class attains a precision of 1.00 and a recall of 0.78, resulting in an F1-score of 0.88. These results indicate that while predictions assigned to this class are consistently correct, a small number of ground-truth instances are misclassified as other glosses.

Specifically, two samples from class 2 are misclassified by the Transformer model, with predictions corresponding to class 0 (*anak_lelaki*) and class 22 (*panas*). Qualitative inspection of these misclassified samples reveals overlapping hand configurations and temporal motion patterns with the predicted classes. In the first case, the misclassified *assalamualaikum* sequence (*assalamualaikum_4_5_2*) exhibits a left-hand configuration resembling a salute-like pose around frame 16, which closely matches the hand posture observed around frame 38 in the ground-truth *anak lelaki* sequence, as shown in Figure 6. In the second case, the misclassified *assalamualaikum* sequence (*assalamualaikum_3_5_2*) shows a downward hand waving

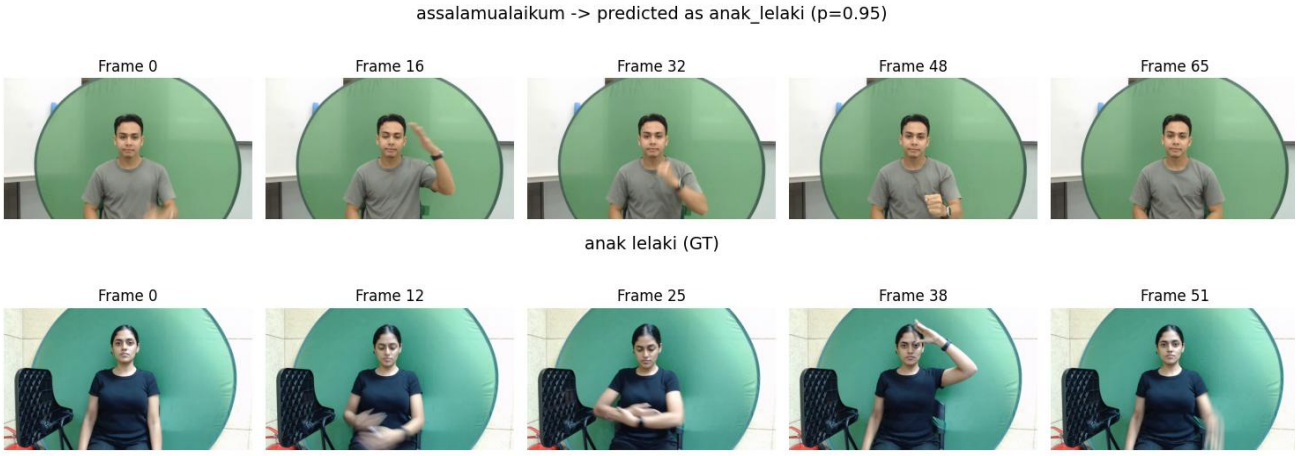

Figure 6: Comparison between a misclassified *assalamualaikum* sequence (*assalamualaikum_4_5_2*), predicted as *anak_lelaki* with a confidence of 0.95 and a ground-truth *anak_lelaki* sample. Both sequences exhibit a visually similar left-hand configuration resembling a salute-like pose, observed around frame 16 in the misclassified *assalamualaikum* sequence (upper row) and around frame 38 in the ground-truth *anak_lelaki* sequence (lower row).

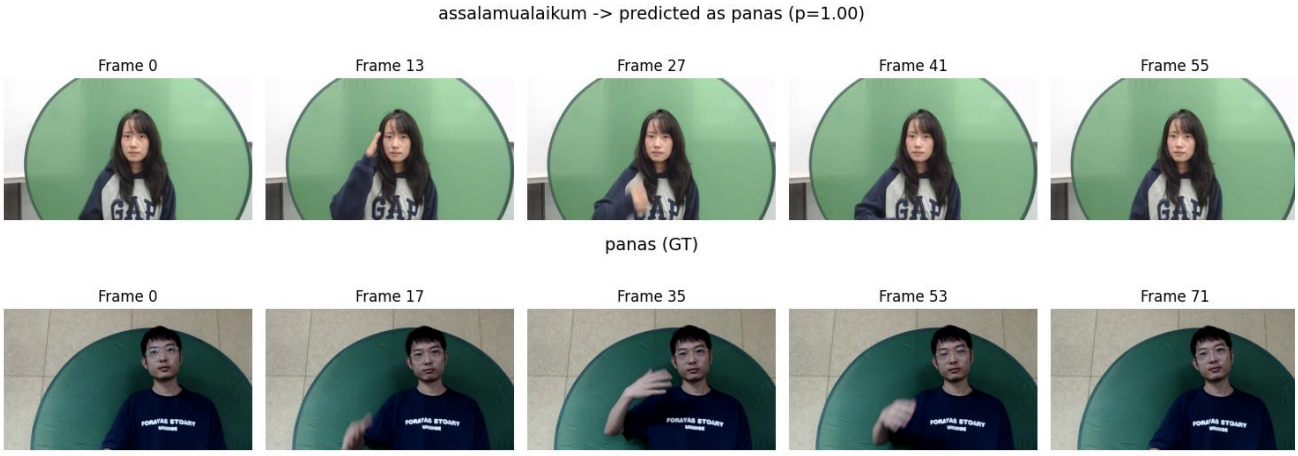

Figure 7: Comparison between a misclassified *assalamualaikum* sequence (*assalamualaikum_3_5_2*), predicted as *panas* with a confidence of 1.00 and a ground-truth *panas* sample. Both sequences exhibit a similar downward hand waving motion, occurring approximately from frames 13 to 41 in the misclassified *assalamualaikum* sequence (upper row) and from frames 35 to 53 in the ground-truth *panas* sequence (lower row).

motion from approximately frames 13 to 41, which is visually similar to the pattern observed between frames 35 and 53 in the ground-truth *panas* sequence, as illustrated in Figure 7.

These examples suggest that the observed misclassifications are associated with localized temporal segments where gesture dynamics overlap across classes. Additionally, the relatively small number of test samples for class 2 may further contribute to the reduced recall. Despite these localized errors, class 2 remains well separated in the confusion matrix and the overall per-class performance of the Transformer model remains high.

D. Application Mock-up

The mock-up application was developed using the Streamlit framework, which facilitates the seamless deployment of the trained machine learning model within an interactive web-based user interface. The application is hosted on the cloud to eliminate technical barriers commonly associated with AI adoption, such as complex local installations or the need for high-end hardware. This design ensures broad accessibility, allowing users to access the system through a standard web browser and internet connection. Figure 8 illustrates the deployed application interface, demonstrating the recognition of the MSL sign *jangan*.

IsyaratAI provides two core functional modes that enable practical use of the proposed recognition model, namely asynchronous video processing and real-time inference. The video upload feature allows users to submit pre-recorded gesture sequences for offline analysis, while the real-time inference mode utilizes the user's webcam to capture live gestures and instantly display corresponding MSL text translations on-screen. These two modes allow the system to support both static and live interaction scenarios without requiring specialized equipment.

The application is intended to support several key user groups within the MSL ecosystem. Deaf and Hard-of-Hearing MSL users may use the system to translate their sign language gestures into text, enabling basic communication with non-MSL users in everyday situations. In addition, hearing individuals who are learning MSL or who frequently interact with MSL users, may utilize the application to better understand and practice MSL gestures during live interactions or structured practice sessions. In educational contexts, teachers and students in special education and MSL learning environments may adopt the application as a supplementary teaching and learning aid. The system can be used to demonstrate, recognize and practice individual MSL glosses, providing immediate feedback that supports skill development. Through these targeted use cases, the application demonstrates how the proposed recognition model can be deployed in a practical and socially meaningful manner, rather than solely as a technical prototype.

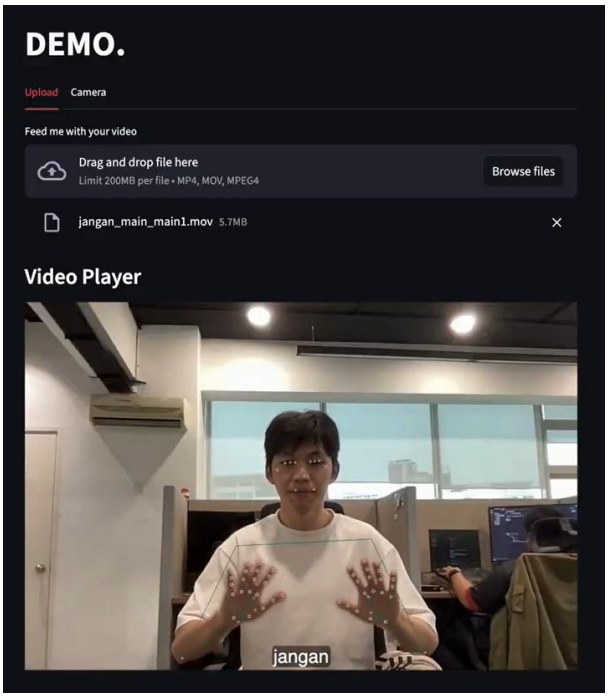

Figure 8: Demonstration of the sign language gloss *jangan* using the deployed IsyaratAI application.

6. CONCLUSION

In this project, an attention-based deep learning framework is presented for MSL recognition using pose and hand landmark sequences. Three temporal models including a baseline LSTM, a bidirectional LSTM with temporal attention and a Transformer encoder were evaluated under identical training conditions. Experimental results demonstrate that attention-based models significantly outperform the baseline LSTM, with the Transformer encoder achieving the best performance, attaining a test accuracy of 98.15% and an F1-score of 0.9814 across 30 MSL glosses. Despite being trained on a dataset collected in a controlled environment and limited to isolated gestures, the proposed approach effectively captures the temporal dynamics of MSL gestures through uniform temporal sampling and self-attention mechanisms. Future work may focus on expanding the gesture vocabulary and incorporating more diverse signer-independent and in-the-wild data to improve robustness under real-world conditions. In addition, further optimization for real-time deployment on edge devices is required to enable practical and scalable use.

7. DECLARATION OF AI TOOLS

AI-based tools were used in this project solely as supportive aids. Writing assistance tools were employed to improve the clarity, structure and grammatical correctness of the report, without contributing to or altering the technical

content, experimental results, or research conclusions. Coding assistance tools were used during model development to support debugging and verification of implementation logic. No AI-based tools were used for experimental design, model architecture selection, data preprocessing decisions, training configuration, evaluation methodology or result interpretation. All research activities were designed, implemented and carried out entirely by the authors.

8. REFERENCES

[1] I. Z. Saiful Bahri *et al.*, "Interpretation of Bahasa Isyarat Malaysia (BIM) Using SSD-MobileNet-V2 FPNLite and COCO mAP," *Information 2023*, vol. 14, no. 6, May 2023, doi: 10.3390/INFO14060319.

[2] A. A. Chong, V. Yee, R. Bee, and M. Hussain, "Language Barriers in Deaf-Centred Classroom: Perspectives from Malaysian Deaf Adults," *Journal of Special Needs Education*, vol. 11, p. 2021, 2021.

[3] T. L. Ta and K. S. Leng, "Challenges Faced by Malaysians with Disabilities in the World of Employment," *Disability, CBR & Inclusive Development*, vol. 24, no. 1, pp. 6–21, May 2013, doi: 10.5463/dcid.v24i1.142.

[4] D. Li, C. R. Opazo, X. Yu, and H. Li, "Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison," *Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, pp. 1448–1458, Mar. 2020, doi: 10.1109/WACV45572.2020.9093512.

[5] S. Srivastava, S. Singh, Pooja, and S. Prakash, "Continuous Sign Language Recognition System Using Deep Learning with MediaPipe Holistic," *Wireless Personal Communications 2024 137:3*, vol. 137, no. 3, pp. 1455–1468, Jul. 2024, doi: 10.1007/S11277-024-11356-0.

[6] B. Sundar and T. Bagyammal, "American Sign Language Recognition for Alphabets Using MediaPipe and LSTM," *Procedia Comput. Sci.*, vol. 215, pp. 642–651, Jan. 2022, doi: 10.1016/J.PROCS.2022.12.066.

[7] S. Das, S. K. R. Biswas, and B. Purkayastha, "An Expert System for Indian Sign Language Recognition Using Spatial Attention–based Feature and Temporal Feature," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 23, no. 3, Mar. 2024, doi: 10.1145/3643824.

[8] L. T. Woods and Z. A. Rana, "Modelling Sign Language with Encoder-Only Transformers and Human Pose Estimation Keypoint Data," *Mathematics 2023, Vol. 11,* vol. 11, no. 9, May 2023, doi: 10.3390/MATH11092129.