Vol 8 No 2 (2026) *e*-ISSN: 2682-8383

Journal of Engineering Technology and Applied Physics

# Quranic Malaysian Sign Language Detection Using Yolo for Real-Time Translation

Abdul Hakim Khahar and Khairi Abdulrahim*

*Department of Electronic/Electrical Engineering, Faculty of Engineering and Built Environment, Universiti Sains Islam Malaysia, 71800, Nilai, Malaysia.*

**Corresponding author*: khairiabdulrahim@usim.edu.my, *ORCiD:* 0000-0002-1751-9250

https://doi.org/10.33093/jetap.2026.8.2.21

*Manuscript Received: 24 September 2025, Revised: 12 November 2025, Accepted: 17 March 2026, Published: 15 September 2026*

***Abstract* — This study presents the development of a real-time Quranic Malaysian Sign Language (MySL) recognition system designed to enhance accessibility to Quranic teachings for the deaf and hard-of-hearing community. Despite the critical necessity for inclusive religious education, the deaf Muslim community faces a profound technological deficit. There is a severe lack of automated systems capable of accurately translating the unique, linguistically complex, and culturally sensitive gestures of Quranic Sign Language in real-time. By leveraging deep learning, the system uses the YOLO (You Only Look Once) object detection model combined with TensorFlow and OpenCV to detect hand gestures representing Quranic letters. A Quranic Sentence Builder was also developed to convert gesture sequences into meaningful sentences. The system achieved high performance, recording 96.54% precision, 95.72% recall, and a 95.26% F1-score. The findings demonstrate the potential of deep learning in supporting inclusive religious education through accurate and efficient sign language translation.**

***Keywords—Deep learning, YOLO, MySL, Quranic sign language, Gesture recognition, Real-Time translation.***

## I. INTRODUCTION

Sign language is a vital communication medium for the deaf and hard-of-hearing, enabling them to convey meaning through visual-manual gestures. Globally, more than 5% of the population is affected by hearing loss, with projections by the World Health Organization suggesting 1 in 10 people may experience hearing disability by 2050 [1]. Among many variants, Quranic Sign Language (QSL) serves as a specialized system tailored to represent the linguistic and religious nuances of the Quran [2].

Traditional sign language recognition systems often face limitations due to gesture complexity, signer variability, and limited annotated datasets. These challenges are especially pronounced in Quranic contexts where gestures carry deep religious meaning and must be interpreted with precision [3]. Recent advancements in deep learning, particularly convolutional neural networks (CNNs) and object detection models like YOLO, offer promising solutions [4].

This project addresses existing limitations by developing a real-time gesture recognition and sentence-building system for Quranic Malaysian Sign Language (MySL). It utilizes YOLOv8 for gesture detection, TensorFlow for deep learning workflows, OpenCV for video stream processing, and a custom Quranic Sentence Builder for converting gestures into meaningful Quranic text output. Figure 1 shows the gestures of MySL obtained through Ibnu Ummi Maktum Research Centre, USIM.


Fig. 1. Quranic Malaysian Sign Language (MySL) gestures.

## II. LITERATURE REVIEW

Journal of Engineering Technology and Applied Physics (2026) 8, 2, 21:157-162
https://doi.org/10.33093/jetap.2026.8.2

Published by MMU PRESS. URL: https://journals.mmupress.com/index.php/jetap/index

The evolution of sign language recognition has been significantly driven by developments in artificial intelligence, particularly deep learning. CNNs and RNNs have demonstrated success in visual and temporal recognition tasks, including sign language [5]. Systems using real-time hand gesture recognition with deep neural networks show promise in accurately identifying sign language alphabets [6].

Sinha *et al.* [7] proposed a real-time translator integrating deep learning with natural language processing, highlighting the benefits of hybrid systems. Similarly, Wadhawan and Kumar [8] conducted a systematic review showing that deep learning outperforms traditional machine learning methods in sign language tasks, especially when supported by large, annotated datasets.

For Quranic Sign Language, Mahmod and Zeki [2] explored classification methods for Quranic recitations and established the need for systems tailored to Islamic content. Abdelghfar *et al.* [5] proposed a CNN-based Qur'anic gesture recognizer, achieving promising accuracy levels on specialized datasets. Yet, challenges remain in ensuring translation fidelity for subtle and closely related hand gestures.

YOLO-based approaches have become increasingly favored for their real-time performance and localization accuracy [4]. Integration of OpenCV with TensorFlow enables efficient preprocessing and deployment pipelines [9]. Recent research also emphasizes the importance of Quran-specific datasets to improve system generalization and reduce misclassification in similar gestures [5, 10].

Despite these advancements, key challenges include class imbalance, gesture variability, signer diversity, and low-resource languages like QSL. This study contributes by creating a scalable MySL recognition pipeline using optimized YOLOv8 architecture and a real-time sentence builder interface.

Extensive research utilizing Long Short-Term Memory (LSTM) networks, MediaPipe, and Transformers [13 - 15] has propelled American and generic Arabic Sign Language recognition forward. These systems are overwhelmingly tailored toward vernacular, daily communication. Conversational sign language permits a degree of fluidity and approximation. Conversely, Quranic education demands rigid, exacting adherence to Tajweed rules and specific orthographic representations. The existing studies that do address Islamic education frequently rely on intrusive, expensive hardware like flex-sensor gloves, or they are restricted to highly constrained, non-real-time environments.

Therefore, the explicit research gap identified by this study is the profound absence of robust, vision-based, computationally lightweight edge-systems capable of performing real-time, high-fidelity detection of domain-specific religious gestures without requiring the user to purchase specialized hardware. This article contributes on:

1. The engineering of a high-precision, low-latency real-time detection pipeline tailored exclusively for the nuances of Quranic MySL using the state-of-the-art YOLOv8 architecture.
2. The conceptualization and deployment of a deterministic "Quranic Sentence Builder," an innovative pedagogical interface that utilizes fixed time-windows to stabilize beginner inputs, completely bypassing the computational latency associated with sequential RNN/LSTM networks.

## III. METHODOLOGY

### A. *Research Tools*

The system was developed using TensorFlow 2.10 for model training and inference, OpenCV 4.11 for video capture and image processing, and YOLOv8.3 for object detection. Python and Jupyter Notebook served as the main programming and testing environment. The system was deployed using a webcam for real-time gesture input, integrated into a Streamlit interface for user interaction. Figure 2 shows the project system flowchart.

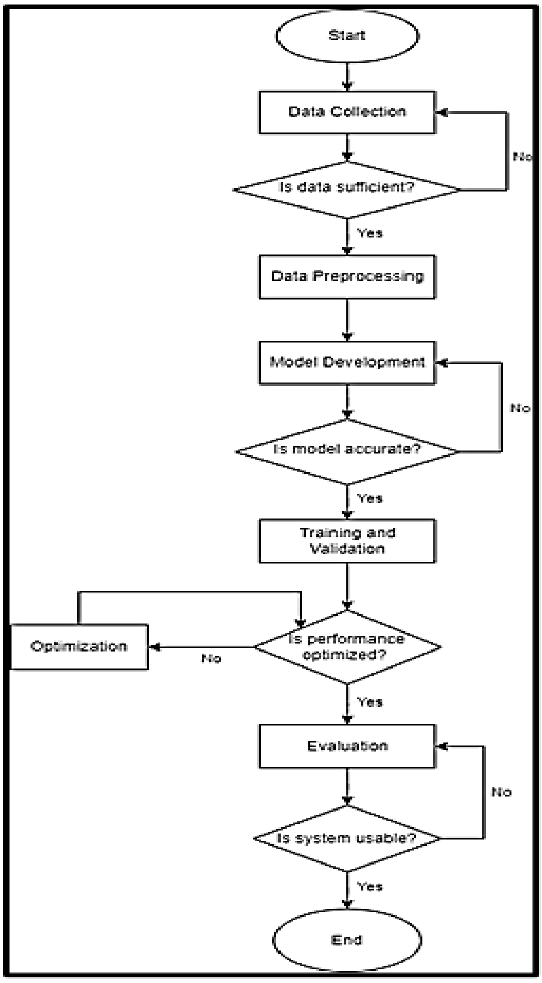



Fig. 2. Project system flowchart.

YOLOv8 was chosen as it introduces an anchor-free detection mechanism. Traditional object detectors rely on predefined anchor boxes of various aspect ratios, which can struggle to map onto the highly variable, non-rigid morphological structures of human hands. By utilizing anchor-free bounding box regression and decoupled classification heads, YOLOv8 predicts the center of an object directly and calculates its bounding parameters dynamically. This makes it exceptionally adept at localizing the rapid, fluid geometries of sign language gestures. Furthermore, YOLOv8 processes the entire visual frame globally in a single forward pass through the neural network. This global context awareness allows the model to implicitly learn the relationship between the hand gesture and the surrounding bodily posture, drastically reducing background false positives and optimizing inference speed for real-time video streams.

### B. Data Collection

Fig. 3. Image capture for letter 'Aleef'.

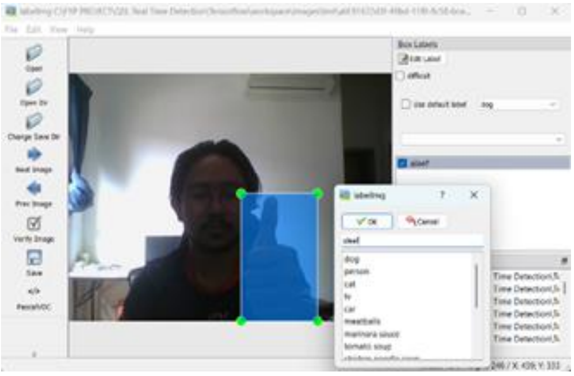


Fig. 4. Image capture for letter 'Aleef'.

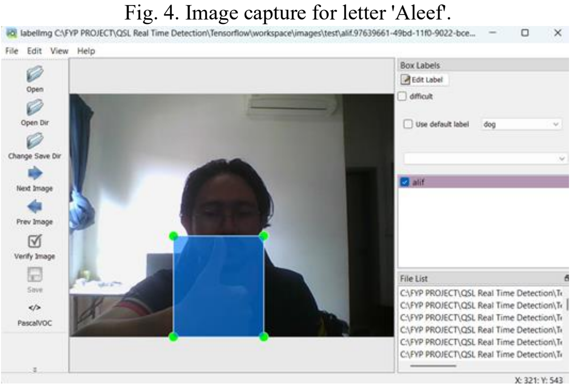



Fig. 5. Image labelling dataset with different angle of gesture.

The dataset consisted of 3,000+ images of MySL gestures representing Quranic letters. These were captured from multiple angles and lighting conditions to enhance model robustness as shown in Fig. 3 and Fig. 4. To improve performance, the dataset was supplemented with an Arabic Sign Language dataset comprising 10,000 images of similar hand gestures. Each gesture was labelled using bounding boxes as shown in Fig. 5. This supplementary data was not intended to replace MySL but to serve as a morphological bridge for transfer learning. Because QSL and ArSL share fundamental phonetic representations of the Arabic alphabet, their manual alphabets exhibit substantial geometric overlap. By pre-training or co-training the model on the larger ArSL dataset, the YOLO architecture develops highly robust feature extraction kernels capable of isolating intricate finger positions.

### C. Data Preprocessing

Images were normalized, augmented (rotated, flipped, brightness-adjusted), and annotated with bounding boxes. Data augmentation ensured model generalization across varied real-world scenarios.

### D. Model Development

The YOLOv8.3 was trained using transfer learning and pre-trained weights for efficiency. The model was optimized through classification and localization loss minimization. A custom "Quranic Sentence Builder" component assembled detected characters into meaningful phrases based on time-controlled logic which is 2s per gesture and 3s pause between words. The model development phase centred around the YOLOv8.3 architecture, selected for its capability to perform real-time object detection with high speed and accuracy. YOLO processes the entire image in a single forward pass, offering significant advantages over multi-stage detection models like R-CNN [4, 9]. Transfer learning was utilized by initializing with pre-trained weights, significantly reducing training time and improving convergence, particularly for the relatively small Quranic MySL dataset [5, 6].

The detection model was trained on the labelled dataset, with parameters optimized using a combination of stochastic gradient descent and adaptive learning rate scheduling. TensorFlow's built-in tools enabled efficient training, while OpenCV was used for real-time webcam integration and video stream processing [9].

A custom "Quranic Sentence Builder" was implemented to convert detected letters into words and full sentences. This module employs a simple time-window mechanism: gestures are recorded within a 2-second detection window, and a 3-second pause between inputs is used to denote word boundaries. This approach ensures smooth sentence formation while reducing recognition overlap and jitter [5, 7].

### E. Training and Validation

The dataset was split evenly between training and testing sets (50:50) to validate model generalization. Due to class imbalance and limited gesture samples, extensive data augmentation such as horizontal flipping, brightness shifts, and rotation was applied [10].

Transfer learning helped improve performance in limited-data scenarios. Pre-trained weights from larger sign language datasets allowed the model to generalize better despite the unique characteristics of Quranic MySL gestures [6].

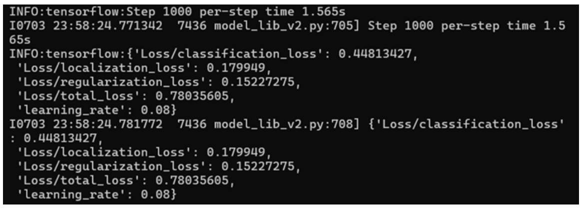
Early stopping and learning rate reduction strategies were employed to prevent overfitting and to stabilize training loss curves [7]. Evaluation was conducted using standard classification metrics: precision, recall, F1-score, and confusion matrix [11, 12].

## IV. RESULTS

### A. *Training Results*

Training logs from 100 to 20,000 epochs revealed a consistent drop in classification and localization loss. At epoch 1000 shown in Fig. 6, the classification loss decreased to 0.44 and localization loss to 0.17. Final training at epoch 20,000 yielded a total loss of 0.25, indicating effective optimization.

This performance aligns with previous deep learning-based gesture recognition efforts [5, 6], confirming YOLO’s suitability for real-time religious sign language applications.



```
INFO:tensorflow:Step 1000 per-step time 1.565s
I0703 23:58:24.771342  7436 model_lib_v2.py:705] Step 1000 per-step time 1.5
65s
INFO:tensorflow:{'Loss/classification_loss': 0.44813427,
 'Loss/localization_loss': 0.179949,
 'Loss/regularization_loss': 0.15227275,
 'Loss/total_loss': 0.78035605,
 'learning_rate': 0.08}
I0703 23:58:24.781772  7436 model_lib_v2.py:708] {'Loss/classification_loss'
: 0.44813427,
 'Loss/localization_loss': 0.179949,
 'Loss/regularization_loss': 0.15227275,
 'Loss/total_loss': 0.78035605,
 'learning_rate': 0.08}
```

Fig. 6. Training results at epoch 1,000.

### B. *Gesture Detection Output*

The model successfully detected key Quranic signs such as "Seen", "Laam", and "Meem" with > 90% confidence scores as shown in Fig. 7. The bounding boxes were accurately placed, and real-time predictions were displayed via the Streamlit interface [9, 10].

However, some confusion was observed between visually similar gestures like "Ra" and "Nun", especially under low-light conditions or hand occlusions. These misclassifications suggest the need for additional gesture samples or enhanced feature extraction, consistent with challenges identified in prior studies [2, 5, 10].

(a)

(b)


Fig. 7. (a) Letters output, ‘Seen’ (top) and (b) ‘Laam’(bottom) for MySL gesture.

Table I. A synthesis of current empirical literature regarding the performance of object detection architectures in real-time sign language recognition environments [13 - 17].

| Detection Architecture | Fundamental Mechanism / Feature Extraction | Average Precision (mAP@50) | Inference Latency / Speed | Suitability for Real-Time QSL Deployment |
|---|---|---|---|---|
| **YOLOv8 (Proposed)** | Single-stage, anchor-free global spatial detection | ~85.6% - 98.6% | ~1.3ms - 10ms (90+ FPS) | **Optimal.** Achieves an exceptional balance, providing high-fidelity bounding box localization with ultra-low computational latency suitable for edge devices. |
| **Faster R-CNN** | Two-stage region proposal and classification network | ~89.3% - 94.0% | ~54ms (8-15 FPS) | **Suboptimal.** Yields the highest accuracy for extremely small or occluded features, but the two-stage process imposes severe latency, rendering real-time video streaming impossible on standard hardware. |
| **SSD (Single Shot Detector)** | Single-stage multi-scale feature grid | ~78.2% - 85.0% | ~16ms (30-60 FPS) | **Moderate.** Operates faster than R-CNN but suffers from lower |

| | | | | |
|---|---|---|---|---|
| | | | | spatial resolution, resulting in poorer localization accuracy for visually similar, fine-grained gestures. |
| **MediaPipe Hands** | 3D landmark topology extraction (21 keypoints) | ~64.0% - 95.0% | ~20ms (50 FPS) | **Suboptimal as standalone.** Highly efficient for tracking isolated hands, but extremely vulnerable to tracking failure during heavy self-occlusion or against complex, noisy backgrounds without prior spatial bounding. |
| **Vision Transformers (ViT/Swin)** | Global attention mechanisms replacing convolutions | ~92.0% - 97.6% | High Latency (<15 FPS on edge) | **Suboptimal.** Delivers state-of-the-art accuracy, but the massive parameter count and sequence-to-sequence memory demands prohibit real-world mobile deployment without specialized AI accelerators. |

### C. Quranic Sentence Builder Output

The Quranic Sentence Builder effectively converted sequential gesture inputs into complete Quranic phrases. For example, it generated the phrase "بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ" through real-time character recognition and temporal spacing, as shown in Fig. 8. This confirms the practicality of a time-based builder mechanism, similar to methods proposed in earlier gesture-to-text systems [7]. Table I shows the comparison analysis matrix benchmarking from the current literature, demonstrating the optimal equilibrium of latency and precision achieved by the YOLO framework.

Fig. 8. Quranic sentence builder output for MySL gesture.

### D. Confusion Matrix and Evaluation

The confusion matrix in Fig. 9 revealed high classification accuracy for most signs. Misclassification rates were higher among similar gestures, especially when background interference or partial visibility occurred. These patterns reflect the same issues observed in other CNN-based sign language models [6, 8].

Evaluation metrics also showed strong overall performance with F1 score of 95.26% and precision score of 96.54%. These metrics exceed those reported in comparable systems for Arabic or Bangla sign language [6, 7], validating the robustness of the proposed model.

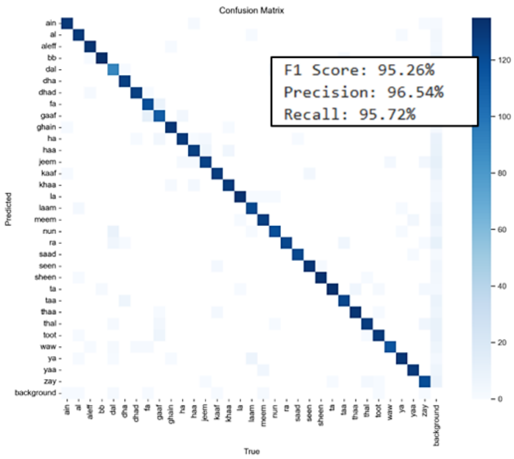


Fig. 9. Confusion matrix for MySL gesture recognition.

## V. CONCLUSION

In conclusion, this research developed a real-time Quranic Malaysian Sign Language (MySL) recognition and translation system using YOLO and deep learning techniques. The system achieved high accuracy in gesture recognition and sentence construction, confirming the feasibility of using AI for inclusive religious education.

The integration of YOLO with a sentence-building mechanism demonstrated promising results for real-time applications, supporting the deaf community in accessing Quranic teachings. This system aligns with Islamic values by enhancing religious accessibility and promoting inclusive learning.

The most critical aspect of real-world validation here is theological, not technological. In Islamic theology, the Quran is regarded as *Kalām Allāh*—the literal and immutable word of God—whose recitation, pronunciation, and manual representation are governed by strict rules to preserve its sanctity and meaning. In Arabic morphology, even a minor alteration in a root letter can radically change meaning and potentially lead to theological distortion. Thus, if an AI system misclassifies a gesture, it is not a simple technical error but a misrepresentation of sacred text.

Ethical integration therefore demands adherence to Islamic principles of *Amanah* (trustworthiness) and *Hifz al-Din* under *Maqasid al-Shariah*. The AI-driven QSL system must be explicitly positioned as a *musāʿid* (supportive pedagogical tool), not a *muḥakkim* (autonomous authority). It is designed to assist beginners in practicing basic orthography, not to replace certified human instructors. Maintaining human-in-the-loop oversight ensures that the technology enhances, rather than compromises, the integrity of Islamic education.

## FUNDING STATEMENT

The research leading to this paper was partly supported by Universiti Sains Islam Malaysia under the grant number PPPI/USIM/FKAB/USIM/16124.

## AUTHOR CONTRIBUTIONS

Abdul Hakim Khahar: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Validation
Khairi Abdulrahim: Validation, Formal AnalysisWriting–Original Draft Preparation, Supervision, Writing–Review & Editing.

## CONFLICT OF INTERESTS

No conflict of interests was disclosed.

## ETHICS STATEMENTS

Our publication ethics follow The Committee of Publication Ethics (COPE) guideline. https://publicationethics.org/

## REFERENCES

[1] F. M. Talaat *et al.*, "Real-time Arabic Avatar for Deaf-mute Communication Enabled by Deep Learning Sign Language Translation," *Comput. and Electr. Eng.*, vol. 119, Part A, pp. 109475, 2024.

[2] M. A. Mahmod and A. Zeki, "Quranic Sign Language for Deaf People: Quranic Recitation Classification and Verification," *Int. J. Percept. and Cognit. Comput.*, vol. 4, no. 1, pp. 7–11, 2018.

[3] F. S. Alamri, A. Rehman, S. B. Abdullahi and T. Saba, "Intelligent Real-life Key-pixel Image Detection System for Early Arabic Sign Language Learners," *Peer J. Comput. Sci.*, vol. 10, pp. e2063, 2024.

[4] J. Han *et al.*, "You Only Look Once–Aluminum: A Detection Model for Complex Aluminum Surface Defects Based on Improved YOLOv8," *Symmetry*, vol. 17, no. 5, pp. 724, 2025.

[5] H. A. AbdElghfar *et al.*, "A Model for Qur'anic Sign Language Recognition Based on Deep Learning Algorithms," *J. Sensors*, vol. 2023, no. 1, pp. 9926245, 2023.

[6] R. B. Rafiq, S. A. Hakim and T. Tabashum, "Real-time Vision-based Bangla Sign Language Detection using Convolutional Neural Network," in *2021 Int. Conf. Adv. Comput. and Commun.*, Kochi, Kakkanad, India, pp. 1–5, 2021.

[7] K. Sinha, A. O. Miranda and S. Mishra, "Real-time Sign Language Translator," in *Cognit. Informat. and Soft Comput.: Proc. CISC 2021*, Springer Nature Singapore, pp. 477–489, 2022.

[8] A. Wadhawan and P. Kumar, "Sign Language Recognition Systems: A Decade Systematic Literature Review," *Arch. Computat. Meth. Eng.*, vol. 28, no. 3, pp. 785–813, 2021.

[9] P. Verma and K. Badli, "Real-Time Sign Language Detection using Tensor Flow, Open CV and Python," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 5, pp. 4483–4488, 2022.

[10] S. Srivastava, A. Gangwar, R. Mishra and S. Singh, "Sign Language Recognition System using TensorFlow Object Detection API," in *Int. Conf. Adv. Netw. Technol. and Intell. Comput.*, Cham: Springer, pp. 634–646, 2021.

[11] G. Tharwat, A. M. Ahmed and B. Bouallegue, "Arabic Sign Language Recognition System for Alphabets Using Machine Learning Techniques," *J. Electr. Comput. Eng.*, vol. 2021, pp. 2995851, 2021.

[12] B. J. Erickson and F. Kitamura, "Performance Metrics for Machine Learning Models," *Radiol. Artif. Intell.*, vol. 3, no. 3, pp. e200126, 2021.

[13] C. Luna-Jiménez, M. Gil-Martín, R. Kleinlein, R. San-Segundo and F. Fernández-Martínez, "Interpreting Sign Language Recognition using Transformers and Mediapipe Landmarks," in *Proc. 25th Int. Conf. Multimodal Interact.*, Paris, France, pp. 373–377, 2023.

[14] A. H. Althubiti and H. Algethami, "Dynamic Gesture Recognition using A Transformer and Mediapipe," *Int. J. Adv. Comput. Sci. and Appl.*, vol. 15, no. 6, pp. 1424-1439, 2024.

[15] C. Artamma and M. Rahardi, "L2IC and MobileViT-XXS for BISINDO Alphabet Recognition," *J. Appl. Informat. and Comput.*, vol. 9, no. 6, pp. 3410–3418, 2025.

[16] "Comparing YOLOv8, SSD, and Faster-RCNN for Real-time Object Detection," ReadyTensor AI. [Available Online on 20 April 2026] https://app.readytensor.ai/publications/comparing-yolov8-ssd-and-fasterrcnn-for-realtime-object-detection-IbA4gAvuaYW8.

[17] "YOLOv8 vs Faster R-CNN: A Comparative Analysis," Keylabs AI Blog. [Available Online on 20 April 2026] https://keylabs.ai/blog/yolov8-vs-faster-r-cnn-a-comparative-analysis/.