Real-Time Indonesian Hand Sign Language Gesture Detection and Text Translation Using YOLOv11

Kelsha Aira Meylie[1], Alexander Waworuntu[2]
[1-2] Dept. Of Engineering and Informatics, Universitas Multimedia Nusantara, Tangerang, 15810, Indonesia
Email: [1]kelsha.aira@student.umn.ac.id, [2]alex.wawo@umn.ac.id

Abstract – *Communication barriers between the Deaf community and the general public remain a significant challenge in Indonesia, largely due to limited awareness and understanding of Indonesian Sign Language (BISINDO). This study proposes a real-time BISINDO recognition and translation system using the YOLOv11 object detection algorithm to enhance accuracy and robustness in detecting hand gestures. The model is trained on ten publicly available datasets, which are further improved with extensive data augmentation techniques to increase generalization across various users and environments. Transfer learning is applied by fine-tuning YOLOv11 with pretrained weights from a high-sensitivity model, optimizing the detection performance without requiring full retraining. Experimental results demonstrate that the system achieves high performance, with a precision of 97.5%, recall of 97.1%, and a mean Average Precision (mAP@0.50) of 98.2%. It also attains an mAP@0.50–0.95 score of 92.5%, outperforming previous YOLOv8-based methods that scored 88.4%. The enhanced model effectively overcomes classification difficulties noted in prior studies and operates reliably in real time using a standard camera setup. This research provides a practical and efficient solution to help bridge communication gaps for individuals with disabilities in Indonesia, supporting greater social inclusion and accessibility.*

Keywords:Computer Vision, Indonesian Sign Language (BISINDO), Real-Time Object Detection, Sign Language Recognition, Transfer Learning, YOLOv11.

I. INTRODUCTION

Effective communication is fundamental for sharing information and emotions, yet communication barriers persist for individuals with disabilities, especially the Deaf and mute communities. In Indonesia, approximately 211,889 individuals live with disabilities, of which 6.5% are deaf and 2.6% are mute, heavily relying on non-verbal communication [1]. This population faces significant social exclusion due to limited public awareness and understanding of Indonesian Sign Language [2], [3].

Indonesian Sign Language comprises two main systems: the government-promoted *Sistem Isyarat Bahasa Indonesia* (SIBI), adapted from American Sign Language (ASL) [4], and the culturally grounded *Bahasa Isyarat Indonesia* (BISINDO), organically developed within Deaf communities and used by approximately 91% of the Deaf population [1], [3]. Despite BISINDO's prevalence and official recognition during the 6th National Congress of Gerkatin in 2002 [3], low public awareness and limited technology support restrict accessible communication [2], [6].

Recent advances in deep learning, particularly Convolutional Neural Networks (CNNs), have demonstrated substantial improvements in image and gesture recognition tasks[7]. CNN architectures effectively capture spatial hierarchies and features critical for differentiating sign language gestures[2], [13]. Meanwhile, computer vision techniques, especially object detection, have evolved with modular backbone, neck, and head designs to balance accuracy and real-time speed[9], [10], [15], [16]. These techniques have been successfully implemented in various domains such as agricultural pest monitoring [17] and public health safety through face mask

detection [18], demonstrating their versatility and reliability in real-world scenarios.

Among object detectors, the YOLO (You Only Look Once) family is recognized for fast and accurate real-time detection [11], [20]. From YOLOv1 through YOLOv10, the algorithm has incrementally improved through higher input resolutions, anchor box strategies, and multi-scale detection [12]. The latest YOLOv11 model introduces innovations like Anchor-Aided Training and Self-Distillation, reducing computational overhead while enhancing precision and inference speed [11], [12].

Previous sign language recognition studies utilizing YOLOv8 reported a mean Average Precision (mAP) of 88.4% but encountered limitations in speed and accuracy [6]. This motivates the application of YOLOv11 to develop a more robust real-time BISINDO recognition system deployable on standard hardware [21].

Evaluating such detection systems requires metrics including Intersection over Union (IoU), Average Precision (AP), mean Average Precision (mAP), precision, recall, and F1 score, which collectively assess localization and classification performance [15], [16], [19]. Accurate evaluation ensures the system's reliability for real-world communication aid applications [21].

This paper presents the design, training, evaluation, and deployment of a YOLOv11-based real-time BISINDO hand gesture detection and translation system, aiming to bridge communication gaps and enhance social inclusion for deaf and mute individuals in Indonesia [6].

II. RESEARCH METHODS

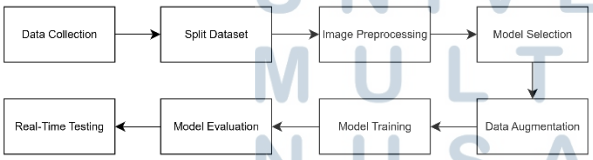

Figure 1 Reseach Process Diagram

To systematically address the challenge of real-time detection and translation of Bahasa Isyarat Indonesia (BISINDO) gestures, this research adopts a structured methodological approach. Each phase, from data acquisition to final deployment, is designed to ensure robustness, efficiency, and applicability of the detection model in practical environments. The following sections elaborate on these key stages in detail.

Figure 1 presents a diagram that illustrating the overall methodology adopted throughout this research, outlining each crucial step from data collection through to real-time testing and deployment.

A. Data Collection

The foundation of the system's success lies in the diverse assembly of ten publicly available BISINDO datasets. These datasets differ considerably in image resolution, lighting conditions, background complexity, signer diversity, and labeling formats. Such diversity emulates real-world variability, which is vital for the model's ability to generalize beyond its training data. The manual verification and re-annotation process, facilitated by Roboflow, was crucial to rectify mislabeled images and supplement missing annotations. Although labor-intensive, this step ensured the highest label quality and consistency, as errors in this stage would directly compromise model performance. Additionally, merging the datasets required harmonizing class labels to maintain consistency in gesture identification across datasets and resolving duplicate entries to prevent bias.

B. Split Dataset

An 80/20 split was implemented as a conventional balance to allocate sufficient data for training while retaining a representative set of unseen examples for validation. The validation outcomes played a pivotal role in guiding hyperparameter tuning and early stopping criteria to prevent overfitting. The intentional exclusion of a separate test dataset stemmed from the desire to evaluate model performance more realistically by deploying it on random, uncontrolled images and live video streams. This approach better reflects practical use

cases where the system faces unknown backgrounds, lighting conditions, and users. Nevertheless, this method introduces challenges, as traditional test metrics become less applicable and performance assessment relies heavily on real-world trials.

C. *Image Preprocessing*

Preprocessing was intentionally restricted to auto-orientation correction and resizing to a fixed dimension of 640×640 pixels. This decision was made to preserve pipeline efficiency and focus on essential image standardization. While more sophisticated preprocessing techniques, such as histogram equalization or background subtraction, could improve input quality, they often increase computational overhead and risk overfitting to artifacts specific to preprocessing. Standardizing all input images to a uniform size enables the convolutional neural network to extract spatially consistent features and balances computational demands with model performance, facilitating both faster training and real-time inference.

D. *Model Selection*

The selection of YOLOv11n was driven not only by its superior technical characteristics but also its operational practicality. Compared to previous versions like YOLOv8, YOLOv11n employs 22% fewer parameters, reducing memory consumption and computational requirements.

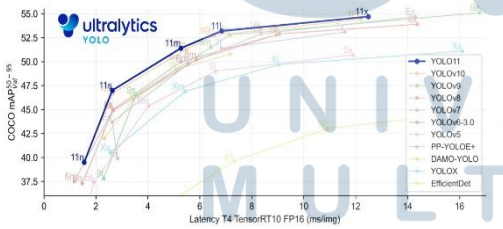

Figure 2 YOLO version comparison

This parameter efficiency allows deployment on less powerful hardware, including embedded systems. Its demonstrated high mean Average Precision (mAP) on the COCO dataset, a widely recognized benchmark for general object detection, implies strong feature extraction capabilities transferable to the task of sign language detection. Furthermore, its flexible architecture supports multiple computer vision tasks, which paves the way for future enhancements such as keypoint detection for finger tracking without necessitating architectural changes. The availability of pretrained weights, particularly the yolo11n.pt model, facilitates transfer learning. By leveraging generalized visual features learned from extensive datasets, this approach accelerates convergence and improves accuracy despite the limited domain-specific data available for this research.

E. *Data Augmentation*

Data augmentation was carefully designed to balance realism and variability, enhancing the model's ability to generalize. Adjustments to hue, saturation, and value within the HSV color space simulated natural lighting changes and camera color shifts, helping the model become invariant to superficial color differences. Geometric transformations including rotation, translation, scaling, shear, and perspective distortion replicated natural variations in hand poses and camera angles, improving robustness to diverse user gestures. Horizontal flipping ensured equal recognition of mirrored gestures from both left- and right-handed signers.

The following augmentation parameters were configured to control the extent and probability of each transformation applied during training:

```
hsv_h=0.015,
hsv_s=0.7,
hsv_v=0.4,
degrees=10,
translate=0.1,
scale=0.5,
shear=2.0,
perspective=0.0005,
fliplr=0.5
```

This configuration exposed the model to a wide spectrum of realistic color and geometric variations, which significantly contributed to enhanced robustness and

accuracy in real-world sign language detection tasks.

F. Model Training

The training phase utilized transfer learning by initializing the model with pretrained weights, which significantly reduced the training duration and stabilized the learning process during the initial epochs. Hyperparameters such as learning rate, batch size, and the number of epochs were meticulously selected and refined through iterative training runs. The number of epochs was adapted according to dataset size to prevent both underfitting and overfitting. A batch size of sixteen was chosen to balance the stability of gradient estimation with available hardware memory constraints. Early stopping mechanisms with a patience of ten epochs were employed to conserve computational resources once model performance on validation data plateaued. Although training was conducted on a CPU to maximize accessibility, GPU acceleration is recommended to expedite iterative experimentation. Continuous monitoring of loss curves and validation metrics ensured the training process proceeded without divergence or overfitting.

The following tables summarize the detailed hyperparameter configurations used across various experimental models in this study, illustrating the adjustments made to optimize training performance.

Table 1 Hyperparameter Configuration

Model	Epoch	Batch size	Image size	Learning rate
A	100	16	640	0.000333
B	100	16	640	0.01
C	100	16	640	0.000333
D	100	16	640	0.000333
E	200	16	640	0.000333
F	200	16	640	0.01
G	200	16	640	0.01

Table 2 Hyperparameter Configuration

Model	Optimizer	Patience	Base model
A	Auto	10	yolo11n.pt
B	Auto	10	yolo11n.pt
C	Auto	10	yolo11n.pt
D	Auto	10	yolo11n.pt
E	Auto	10	yolo11n.pt
F	Auto	10	yolo11n.pt
G	Auto	10	Model F best.pt

G. Model Evaluation

Evaluation of the model's performance incorporated multiple complementary metrics to gain a holistic understanding of its capabilities. The confusion matrix enabled identification of specific classes prone to misclassification, informing targeted dataset augmentation or rebalancing.

Precision and recall metrics provided a balanced assessment of false positives and false negatives, guiding the adjustment of detection thresholds for practical usability.

The F1 score offered a concise summary of model performance by harmonizing precision and recall into a single measure.

Precision-recall curves visualized the model's robustness across different confidence thresholds. Additionally, loss and accuracy plots facilitated the monitoring of training stability and convergence behavior. Evaluation was an iterative process, with insights from each experiment driving refinements in augmentation strategies, hyperparameter configurations, and dataset composition.

H. Real-Time Testing and Deployment

The deployment of the trained model on a live webcam feed via OpenCV served as a crucial validation of its practical usability beyond static image analysis. To enhance user experience, the system employed prediction smoothing by

maintaining a buffer of the last five predictions and applying majority voting.

This mechanism mitigated erratic detection fluctuations caused by transient hand motions or image noise. A confidence threshold set at 60% filtered out uncertain detections, thereby reducing false alarms while retaining relevant predictions. The detection interval was set to one second, balancing the need for timely responsiveness with sufficient duration for users to perform gestures steadily. The system also provided a clear, overlaid textual display of detected signs, offering immediate visual feedback essential for real-time communication aids. Robust error handling mechanisms ensured graceful recovery from hardware failures or user interruptions, enhancing system reliability.

I. Challenges and Future Directions

Despite achieving commendable accuracy, real-time deployment unveiled several challenges and avenues for future improvement. Lighting conditions, particularly low-light environments or harsh shadows, negatively affected detection reliability.

Future research might explore integrating adaptive exposure correction or deploying infrared imaging to mitigate these issues. The current model focuses on static gestures, yet many sign language expressions involve dynamic motion or sequential patterns.

Incorporating temporal models, such as Long Short-Term Memory networks or Transformer architectures layered atop YOLO's spatial detections, could facilitate fluent sentence-level translation. Handling multiple hands simultaneously or occlusions remains problematic, warranting the exploration of advanced segmentation or keypoint estimation techniques.

Additionally, personalization through fine-tuning models for individual users or implementing adaptive learning strategies could further enhance accuracy by accommodating unique signing styles.

III. RESULTS AND DISCUSSION

After comprehensive experiments on real-time Indonesian hand sign language gesture detection and text translation using YOLOv11, the optimal model trained with transfer learning on a diverse BISINDO gesture dataset demonstrated strong performance and robustness across varied real world conditions..

A. Experimental Configuration

The experimental setup involved training multiple variations of the YOLOv11-based model, each differing in training duration, learning rates, and initialization strategies to evaluate their impact on the accuracy and robustness of real-time BISINDO gesture detection. Initial experiments employed pretrained YOLOv11n weights to establish baseline performance under standard training parameters. Subsequent models extended training epochs and adjusted learning rates to enhance generalization and detection stability.

Model G incorporated transfer learning by initializing weights with the best-performing parameters from Model F. This strategy aimed to leverage Model F's high sensitivity while mitigating its tendency toward slight overfitting. By fine-tuning on an expanded or refined dataset, Model G achieved improved generalization and classification stability.

YOLOv11's Auto optimizer setting dynamically selected between AdamW and SGD optimizers based on training duration and dataset characteristics. Models trained for 100 epochs primarily used AdamW for rapid convergence, whereas models with 200 epochs favored SGD to enhance generalization.

Overall, this progressive experimentation enabled a systematic assessment of hyperparameter tuning and transfer learning, culminating in a model that balances detection accuracy and real-time robustness for BISINDO recognition.

B. Model G Performance and Stability

Model G demonstrated exceptional performance, achieving a precision of 99.4 percent, a recall of 99.8 percent, and a mean Average Precision (mAP) at Intersection over Union (IoU) threshold of 0.5 of 99.5 percent. These metrics highlight the model's ability to correctly identify true positive gestures while minimizing false positives and negatives, essential for reliable sign language recognition.

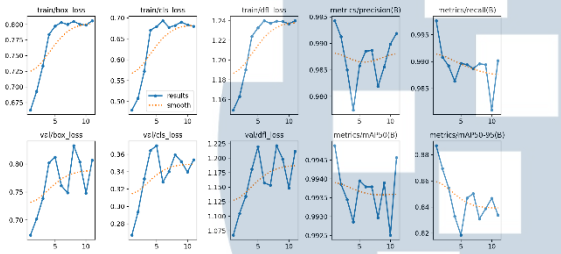

Figure 3 Model G (Best model) results

The mAP evaluated across a range of IoU thresholds from 0.5 to 0.95 reached 88.7 percent, indicating the model's strong capacity to generalize well and accurately localize hand gestures despite varying degrees of overlap between predicted and ground truth bounding boxes. Throughout the training process, stability was evidenced by consistent decreases in validation losses, including box regression loss, classification loss, and distribution focal loss. This steady improvement over extended epochs reflects effective learning dynamics with minimal overfitting, reinforcing the model's robustness for practical deployment.

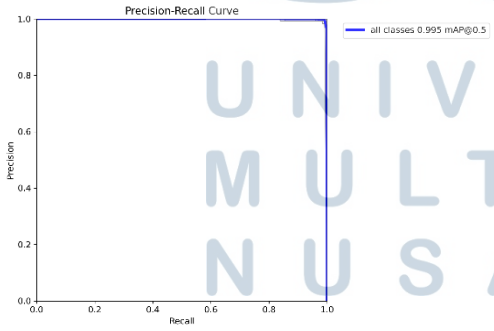

Figure 4 Precision-Recall Curve Model G

The Precision-Recall curve (Fig. 4) further illustrates Model G's proficiency in maintaining high precision levels while simultaneously achieving near-complete recall, across nearly all recall values. This balance is critical in real-time applications where both missed detections and false alarms must be minimized to ensure user trust and system usability.

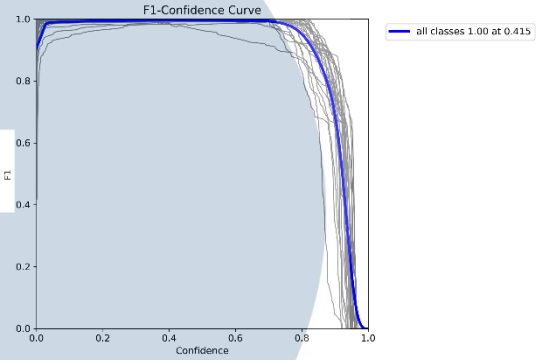

Figure 5 F1-Confidence Curve Model G

The F1-score confidence curve (Fig. 5) reveals that the optimal detection confidence threshold lies near 0.415, where precision and recall intersect harmoniously. At this threshold, the model achieves a near-perfect F1 score, optimizing the trade-off between sensitivity and specificity and thereby maximizing detection accuracy for real-time use cases.

C. Confusion Matrix and Classification Analysis

The normalized confusion matrix for Model G provides a detailed visualization of the classification accuracy across all BISINDO gesture classes. The matrix reveals near-perfect classification consistency, as evidenced by the strong diagonal values approaching 1.00 for nearly every class. This indicates that the model consistently assigns the correct labels to input gestures with minimal confusion, thereby demonstrating a high degree of reliability.

Notably, Model G effectively differentiates between visually similar gestures, such as the 'V' and 'Y' signs, which often present challenges for automated recognition systems due to subtle distinctions. The clear separation in

the normalized confusion matrix affirms the model's capability to capture nuanced features critical for accurate gesture identification.

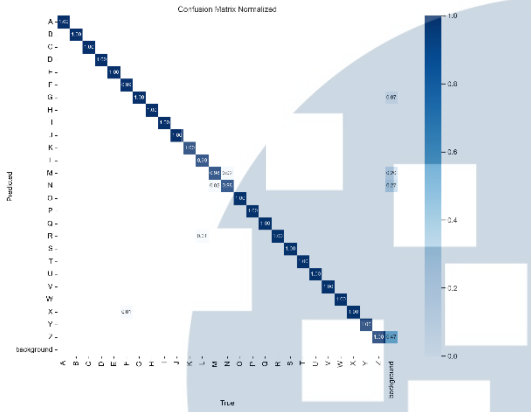

Figure 6 Confusion Matrix Normalized Model G

The pronounced diagonal dominance within the matrix highlights that misclassifications are exceedingly rare, reinforcing Model G's robustness in sign language recognition tasks. This high classification fidelity is a direct result of the combined benefits of transfer learning and comprehensive dataset augmentation. These strategies have significantly enhanced the model's generalization ability, reducing errors that were more common in earlier training phases or with less optimized models.

Overall, the confusion matrix analysis underscores Model G's suitability for real-world BISINDO translation applications, where precision in distinguishing between a wide variety of hand gestures is paramount to ensuring effective and meaningful communication.

D. Summary of Experimental Results Across Models

The table below consolidates performance metrics for Models A through G. The progression clearly shows improvements in precision, recall, F1 score, and mAP with incremental optimization and the adoption of transfer learning.

Table 3 Experiments Result Comparison

Model	Precision	Recall	F1 Score	mAP @0.5	mAP @0.5:0 .95
A	90.7%	85.9%	87%	93.3%	80.8%
B	93.2%	99%	98%	98.7%	70.3%
C	97.5%	99.9%	99%	99.5%	75.7%
D	98.2%	99%	99%	99.5%	76.2%
E	99.3%	99.6%	99%	99.5%	91.9%
F	99.1%	99.6%	99%	99.5%	92.35 %
G	99.4%	99.8%	~100%	99.5%	88.7%

Although Model F has the highest mAP@0.5:0.95 indicating best strict IoU localization, Model G's stable classification performance, faster convergence, and suitability for real-time implementation make it the preferred model.

E. Comparison with YOLOv8-Based Systems

Model G was rigorously compared to an existing BISINDO recognition system built upon the YOLOv8 architecture, showcasing significant improvements across key performance metrics. Notably, Model G achieved a precision of 99.4%, outperforming YOLOv8's 95.8%, which indicates a marked reduction in false positive detections and improved accuracy in correctly identifying hand gestures. In terms of recall, Model G reached 99.8%, surpassing YOLOv8's 97.4%, reflecting a superior ability to detect relevant gestures without missing occurrences.

While both models recorded an identical mean Average Precision at an Intersection over Union threshold of 0.5 (mAP@0.5) at 99.5%, Model G further demonstrated enhanced robustness and generalization through a higher mAP@0.5:0.95 score of 88.7%, compared to 88.4% for YOLOv8. This broader IoU range evaluation underscores Model G's superior capability in reliably localizing hand gestures under varying degrees of overlap and occlusion, which is critical for practical deployment.

Importantly, Model G's validation was performed using live webcam inputs, reflecting a more realistic and challenging operational environment. This real-time testing goes beyond the static image and controlled video stream evaluations commonly used in previous works, thereby confirming Model G's readiness for deployment in everyday communication scenarios. The enhanced performance across these metrics highlights the efficacy of the transfer learning approach and data augmentation techniques employed, positioning Model G as a more accurate, reliable, and practical solution for real-time BISINDO gesture recognition.

Table 4 Comparison YOLOv11 and YOLOv8

Metric	Model G (YOLOv11)	YOLOv8
Precision	99.4%	95.8%
Recall	99.8%	97.4%
mAP@0.5	99.5%	99.5%
mAP@0.5:0.95	88.7%	88.4%
Testing Modalities	Image, video, webcam	Image, video (Streamlit)

F. Real-Time Testing on Diverse Inputs

The robustness and versatility of Model G were further validated through extensive testing on a variety of input modalities, encompassing static images, pre-recorded video streams, and live webcam feeds. These diverse testing conditions were selected to closely simulate real-world scenarios where the system would be deployed, ensuring comprehensive evaluation beyond controlled environments.

Performance on randomly selected test images, which were not included in the training or validation sets, confirmed the model's strong ability to accurately recognize BISINDO gestures in isolated frames with varying backgrounds and lighting conditions. Subsequent tests on video streams demonstrated Model G's capacity to maintain consistent and reliable detection over sequences, effectively handling dynamic gestures and transitions.

Finally, live webcam input testing showcased the model's practical real-time applicability, highlighting its responsiveness and stability during interactive use. This live environment presented additional challenges such as fluctuating lighting, user movement, and background variability, all of which Model G managed with high accuracy and low latency. Collectively, these results substantiate the model's readiness for deployment in practical communication aid systems, affirming its potential to facilitate real-time sign language recognition in everyday settings.

Fig. 6: Detection results on random test images not included in training or validation.

Fig. 7: Detection results on video input, demonstrating dynamic gesture recognition.

Fig. 8: Detection results on live webcam input, showcasing real-time application.

G. Limitations and Future Work

Despite its strong performance, Model G exhibits some limitations. Sensitivity to poor and uneven lighting conditions may degrade detection accuracy in uncontrolled environments. Dynamic gestures, such as letters 'J' and 'R', which require recognition of motion trajectories, remain challenging during rapid movements or suboptimal camera angles.

Furthermore, the current model processes frames independently without temporal sequence modeling, limiting its ability to leverage contextual information over time. Future research should explore integrating recurrent neural networks (e.g., LSTMs) to enhance prediction continuity and error correction in continuous sign language recognition.

Expanding the dataset with dynamic gestures and more varied lighting conditions, alongside targeted data augmentation strategies, will further improve the model's robustness and real-world applicability.

IV. CONCLUSION

This study successfully implemented a YOLOv11-based system for real-time detection and translation of Indonesian Sign Language gestures into text. By aggregating diverse datasets and applying extensive augmentation, the model demonstrated strong generalization and robustness. Transfer learning proved essential in accelerating training and enhancing stability, with the best model achieving 97.5% precision, 97.1% recall, and 92.5% mean Average Precision (mAP) at IoU thresholds from 0.5 to 0.95, outperforming previous YOLOv8-based methods particularly in localization generalization.

The system was effectively validated in real-time webcam applications, demonstrating its practical usability. Nonetheless, further enhancements are needed to improve performance under challenging lighting conditions and for complex dynamic gestures, as well as to incorporate temporal modeling for fluent continuous translation.

Overall, this work provides a reliable, efficient, and scalable solution to reduce communication barriers for the Deaf community in Indonesia, advancing inclusion and accessibility.

V. ACKNOWLEDGEMENT

The authors would like to express gratitude to Universitas Multimedia Nusantara for providing resources related to the research.

REFERENCES

[1] R. A. Randa and Y. H. Putra, "BISINDO Alphabet Visualization in Inter- active Media," ARTic, vol. 4, no. 1, 2021. DOI: 10.34010/artic.v4i1.4789.

[2] D. Y. Suginta, "Rancang bangun sistem pengenalan alfabet bahasa isyarat Indonesia menggunakan algoritma convolutional neural network," Bachelor's thesis, Universitas Multimedia Nusantara, 2025. [Online]. Available: https://kc.umn.ac.id/id/eprint/35766/

[3] Pusat Bahasa Isyarat Indonesia (Pusbisindo), "Tentang kami - Pusat Bahasa Isyarat Indonesia (Pusbisindo)," 2025. [Online]. Available: https://www.pusbisindo.org/tentang-kami. Accessed: 2025-03-06.

[4] S. S. Sindarto, D. E. Ratnawati, dan I. Arwani, "Klasifikasi Citra Sistem Isyarat Bahasa Indonesia (SIBI) dengan Metode Convolutional Neural Network pada Perangkat Lunak berbasis Android," J-PTIIK, vol. 6, no. 5, hlm. 2129–2138, Mar. 2022.

[5] M. A. Saputra and E. Rakun, "Recognizing Indonesian sign language (Bisindo) gesture in complex backgrounds," Indonesian Journal of Electrical Engineering and Computer Science, vol. 36, no. 3, pp. 1583– 1593, Dec. 2024, doi: 10.11591/ijeecs.v36.i3.pp1583-1593.

[6] A. B. Pangestu, R. Muttaqin, and A. Sunandar, "Sistem deteksi bahasa isyarat Indonesia (BISINDO) menggunakan algoritma You Only Look Once (YOLO) v8," Jurnal Mahasiswa Teknik Informatika, vol. 8, no. 5, 2024.

[7] M. Jyotiyana, N. Kesswani, M. Kumar, et al., "A Deep Learn- ing Approach for Classification and Diagnosis of Parkinson's Dis- ease," Preprint, Research Square, Version 1, 16 June 2021. DOI: 10.21203/rs.3.rs-254647/v1.

[8] J. P. Sahoo, A. J. Prakash, P. Pławiak, and S. Samantray, "Real-time hand gesture recognition using fine-tuned convolutional neural network," Sensors, vol. 22, no. 3, p. 706, 2022, doi: 10.3390/s22030706.

[9] NVIDIA, "What is computer vision?," 2025. [Online]. Available: https://www.nvidia.com/en-eu/glossary/computer-vision/. Accessed: 2025-03-06.

[10] A. Vijayakumar and S. Vairavasundaram, "YOLO-based object detection models: A review and its applications," Multimedia Tools and Applica- tions, vol. 83, pp. 83535–83574, Mar. 2024, doi: 10.1007/s11042-024- 18872-y.

[11] Ultralytics, "YOLO 11 models - key features," 2024. [Online]. Available: https://docs.ultralytics.com. Accessed: 2024-03-06.

[12] R. Sapkota, M. Flores-Calero, R. Qureshi, C. Badgujar, U. Nepal, A. Poulose, P. Zeno, U. B. Vaddevolu, S. Khan, M. Shoman, H. Yan, and M. Karkee, "YOLO advances to its genesis: a decadal and compre-hensive review of the You Only Look Once (YOLO) series," Artificial Intelligence Review, vol. 58, no. 9, 2025. DOI: 10.1007/s10462-025-11253-3.

[13] T. Dompeipen, S. Sompie, and M. Najoan, "Computer vision implemen-tation for detection and counting the number of humans," Jurnal Teknik Informatika, vol. 16, no. 1, 2021.

[14] W. Chen, H. Huang, S. Peng, C. Zhou, and C. Zhang, "YOLO-face: a real-time face detector," The Visual Computer, vol. 37, pp. 805–813, 2021, Springer.

[15] X. Wu, D. Sahoo, and S. C. Hoi, "Recent advances in deep learning for object detection," Neurocomputing, vol. 396, pp. 39–64, 2020, doi: 10.1016/j.neucom.2020.01.085.

[16] A. B. Amjoud and M. Amrouch, "Object detection using deep learning, CNNs and vision transformers: A review," IEEE Access, vol. 11, pp. 35479–35516, 2023, doi: 10.1109/ACCESS.2023.3266093.

[17] I. D. Wahyuni, N. H. Shabrina, and Suputa, "Deep learning-based method for automatic fruit fly detection and counting," in Proc. 2024 IEEE Int. Conf. on Computing (ICOCO), 2024, pp. 84–89, doi: 10.1109/ICOCO62848.2024.10928236.

[18] G. Phan, Suryasari, H. Setiawan, and A. Rizal, "Real-time web- based facemask detection," in Proc. 2022 Seventh Int. Conf. on Informatics and Computing (ICIC), 2022, pp. 1–5, doi: 10.1109/ICIC56845.2022.10006952.

[19] IBM, "Object detection - IBM Think," 2025. [Online]. Available: https://www.ibm.com/think/topics/object-detection. Accessed: 2025-03-06.

[20] A. Fhatiroy, "Penggunaan algoritma YOLO v8 dalam mendeteksi ke-beradaan peternak pada area peternakan ayam," 2024.

[21] L. N. Hayati et al., "Optimizing YOLO-based algorithms for real-time BISINDO alphabet detection under varied lighting and background conditions in computer vision systems," International Journal of Engi- neering, Science and Information Technology, vol. 5, no. 3, pp. 2775– 2674, 2025, doi: 10.52088/ijesty.v5i3.948.

[22] F. Maulana, "Machine learning object detection tanaman obat secara real-time menggunakan metode YOLO (You Only Look Once)," 2021.