Buletin Ilmiah Sarjana Teknik Elektro
Vol. 7, No. 4, December 2025, pp. 774-788
ISSN: 2685-9572, DOI: 10.12928/biste.v7i4.14507

Evaluation of the Effectiveness of Hand Gesture Recognition Using Transfer Learning on a Convolutional Neural Network Model for Integrated Service of Smart Robot

Faikul Umam [1], Ach. Dafid [2], Hanifudin Sukri [3], Yuli Panca Asmara [4], Md Monzur Morshed [5], Firman Maolana [6], Ahcmad Yusuf [7]
[1,2,6,7] Department of Mechatronics Engineering, Faculty of Engineering,Universitas Trunojoyo Madura, Indonesia
[3] Department of Information Systems, Faculty of Engineering, Universitas Trunojoyo Madura, Indonesia
[4] INTI International University, FEQS, Nilai, Malaysia
[5] University of Dhaka, Dhaka-1000, Bangladesh

ARTICLE INFORMATION

Article History:

Received 20 August 2025
Revised 19 October 2025
Accepted 08 November 2025

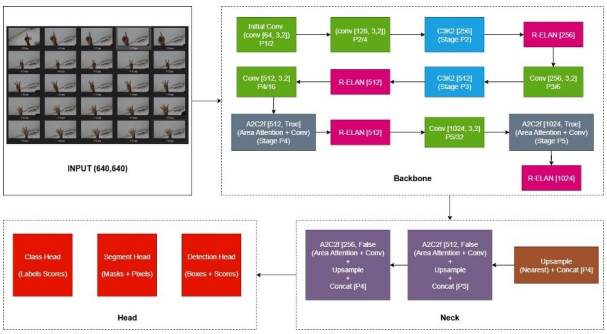
Keywords:

Transfer Learning;
Hand Gesture Recognition;
Convolutional Neural Network;
Service Robot Integrated;
Innovation;
Information and Communication Technology;
Infrastructure

Corresponding Author:

Faikul Umam,
Department of Mechatronics Engineering Faculty of Engineering, Universitas Trunojoyo Madura, Indonesia.
Email: faikul@trunojoyo.ac.id

ABSTRACT



This study aims to develop and evaluate the effectiveness of a transfer learning model on CNN with the proposed YOLOv12 architecture for recognizing hand gestures in real time on an integrated service robot. In addition, this study compares the performance of MobileNetV3, ResNet50, and EfficientNetB0, as well as a previously funded model (YOLOv8) and the proposed YOLOv12 development model. This research contributes to SDG 4 (Quality Education), SDG 9 (Industry, Innovation and Infrastructure), and SDG 11 (Sustainable Cities and Communities) by enhancing intelligent human–robot interaction for educational and service environments. The study applies an experimental method by comparing the performance of various transfer learning models in hand gesture recognition. The custom dataset consists of annotated hand gesture images, fine-tuned to improve model robustness under different lighting conditions, camera angles, and gesture variations. Evaluation metrics include mean Average Precision (mAP), inference latency, and computational efficiency, which determine the most suitable model for deployment in integrated service robots. The test results show that the YOLOv12 model achieved an mAP@0.5 of 99.5% with an average inference speed of 1–2 ms per image, while maintaining stable detection performance under varying conditions. Compared with other CNN-based architectures (MobileNetV3, ResNet50, and EfficientNetB0), which achieved accuracies between 97% and 99%, YOLOv12 demonstrated superior performance. Furthermore, it outperformed previous research using YOLOv8 (91.6% accuracy), confirming its effectiveness for real-time gesture recognition.

Document Citation:

F. Umam, A. Dafid, H. Sukri, Y. P. Asmara, M. M. Morshed, F. Maolana, and A. Yusuf, "Evaluation of the Effectiveness of Hand Gesture Recognition Using Transfer Learning on a Convolutional Neural Network Model for Integrated Service of Smart Robot," *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 7, no. 4, pp. 774-788, 2025, DOI: 10.12928/biste.v7i4.14507.

1. INTRODUCTION

With the increasingly digital era developing, the integration of artificial intelligence in robotics systems has become a crucial element in enhancing the efficiency of services in various sectors, including education. Intelligent robot service integration is a potential innovation that optimizes interaction between humans and machines through more intuitive systems, such as Hand Gesture Recognition (HGR) technology. This allows robots to understand human visual cues without the need for verbal commands, thereby increasing the convenience of accessing academic, administrative, and student affairs in a college environment. The Convolutional Neural Network (CNN) approach has proven to be an effective method for detecting hand movements, but training the learning model from scratch often requires a large amount of computational power and time [1]-[4]. Therefore, transfer learning becomes a strategic solution to adopt the advantages of the CNN model that has been trained previously on large datasets, thereby increasing accuracy as well as efficiency in time and power processing.

Although CNN-based methods have been widely used for gesture recognition, their implementation in integrated service robots across diverse environments remains challenging, often reducing detection accuracy. Many models have been developed previously, but they still have limitations in adapting to variations in gesture execution, hand appearance, lighting conditions, and user positioning. Some studies have previously used YOLO and conventional CNNs to detect objects in static environments, but there has been little exploration of implementing transfer learning in the context of introducing gestures for interactive robots [5]-[10]. Several study latest show that method transfer learning with architecture such as MobileNetV3 [11]-[16], ResNet50 [17]-[23], and EfficientNet [24]-[31] can increase efficiency detection without sacrifice accuracy. However, their application in real-time environments and adaptation to dynamic conditions within campus contexts remain open challenges that require further investigation.

To overcome this problem, research proposes the implementation of a new method, transfer learning on CNN models the proposed YOLOv12 architecture, to improve the effectiveness of introducing movement in human-robot interaction. This approach utilizes a previously trained CNN model with large datasets, which is then customized for the scenario service campus, thereby speeding up the training process while increasing accuracy for introduction gestures, such as hand gestures. The model to be reviewed encompasses a range of modern CNN architectures, including MobileNetV3, ResNet50, and EfficientNetB0. The models are known to have optimal performance in task visual classification. Evaluation will be done by comparing model performance in various conditions of lighting, camera angle, and hand gesture variations.

From the perspective of the Sustainable Development Goals (SDGs), the contribution study includes: (1) Enhancing access and quality of education through intelligent robot-based interactions (SDG 4: Quality Education); (2) Promoting innovative technology in inclusive digital infrastructure (SDG 9: Industry, Innovation, and Infrastructure); and (3) Delivering sustainable technological solutions for smart campuses (SDG 11: Sustainable Cities and Communities). The research contribution is a comparative evaluation of transfer learning models for real-time hand gesture recognition in integrated service robots, demonstrating the potential of YOLOv12 as an effective solution for adaptive and robust human–robot interaction in educational environments.

2. METHODS

Transfer learning is one of the approaches in the field of machine learning that enables the use of pre-trained weights and architectures from models trained on large-scale datasets such as ImageNet or COCO It is applied to new tasks that have characteristics similar to those of the original task [32]-[34]. This approach is particularly useful when training data is limited for a new task, or when a large source of power is needed for computations from the beginning of the training process. In the context of computer vision, transfer learning provides flexibility for using more advanced model architectures without needing to start training from scratch, thereby saving significant time and energy in computing [35]-[41].

In this research, transfer learning is implemented on the Integrated Service Robot system (Robolater), designed to recognize hand gestures of users through a camera as the primary visual perception device, as illustrated in Figure 1. The detection model previously using YOLOv8 showed adequate performance under standard environmental conditions [42]. However, to increase detection accuracy under diverse lighting conditions and camera angles, fine-tuning and adaptation were carried out on the proposed YOLOv12 model. YOLOv12 in this study is an internal development based on modifications of the YOLOv8 backbone and detection head, designed to improve small-object recognition and inference efficiency in dynamic environments. This process was not performed by training from scratch but rather by leveraging YOLOv8 weights and adapting them into the YOLOv12 framework. YOLOv12 was selected because it offers improved

inference efficiency, better handling of small-scale features, and more robust object detection under varied lighting conditions compared to YOLOv8, as shown in Figure 2.

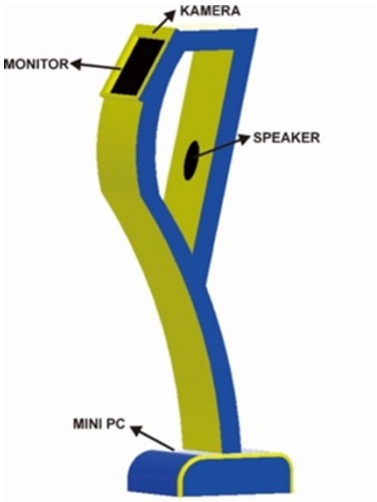


Figure 1. Illustration of an Integrated Service of Smart Robot

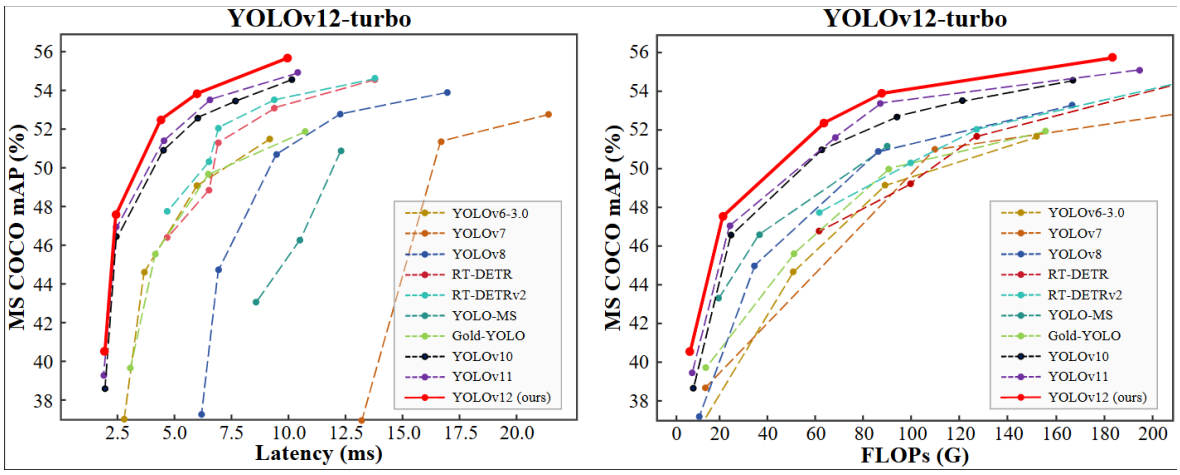


Figure 2. Comparison of YOLO Models

The Convolutional Neural Network (CNN) method is applied to the YOLOv12 model, based on the principle of extracting hierarchical features from images through convolution, pooling, and non-linear activation, thereby enabling the representation of complex spatial patterns. In the YOLOv12 framework [43]-[50], the methodology This implemented in a way end-to-end with three component central Figure 3: (1) backbone, which functions as extractor feature for catch visual information from level low until height; (2) neck, which does fusion multi- scale features to maintain context spatial in various size object; and (3) head, which is basically direct predicting bounding box, objectless scores, and classification class. The training process utilizes a function loss combination (localization, classification, and confidence) and adaptive algorithm optimization, so that it produces a model table of detecting objects in real-time with high computational efficiency. Thus, YOLOv12 is confirmed as a single-stage detector that integrates feature extraction and prediction in a unified framework.

The YOLOv12 model was trained using a transfer learning approach from YOLOv8 with a custom internally curated hand gesture dataset containing 1,025 images across five gesture classes. Each image was annotated with bounding boxes and divided into training (70%), validation (15%), and test (15%) sets to ensure balanced evaluation. Training was conducted on a dedicated GPU machine (NVIDIA RTX 3080, 10 GB VRAM) with data augmentation techniques, such as flipping, brightness/contrast adjustment, and light blurring, to increase robustness against variations in lighting conditions, gesture shapes, and camera perspectives. The training was run for 100 epochs with a batch size of 16 and image resolution of 640 × 640

pixels, ensuring convergence without overfitting. For benchmarking purposes, three other CNN architectures (MobileNetV3, ResNet50, EfficientNetB0) were also trained and evaluated under the same dataset and augmentation settings. This allows for a fair comparison of model performance against the proposed YOLOv12 architecture.

Figure 4. Research methodology flowchart. The process starts with dataset preparation, annotation, and dataset splitting. Data augmentation is applied to enhance robustness. Transfer learning is conducted using YOLOv8 weights, followed by fine-tuning YOLOv12 with hyperparameter tuning. Model performance is evaluated using accuracy, mAP, precision, recall, and latency, then compared with MobileNetV3, ResNet50, and EfficientNetB0. The final optimized model is deployed on the Robolater system.

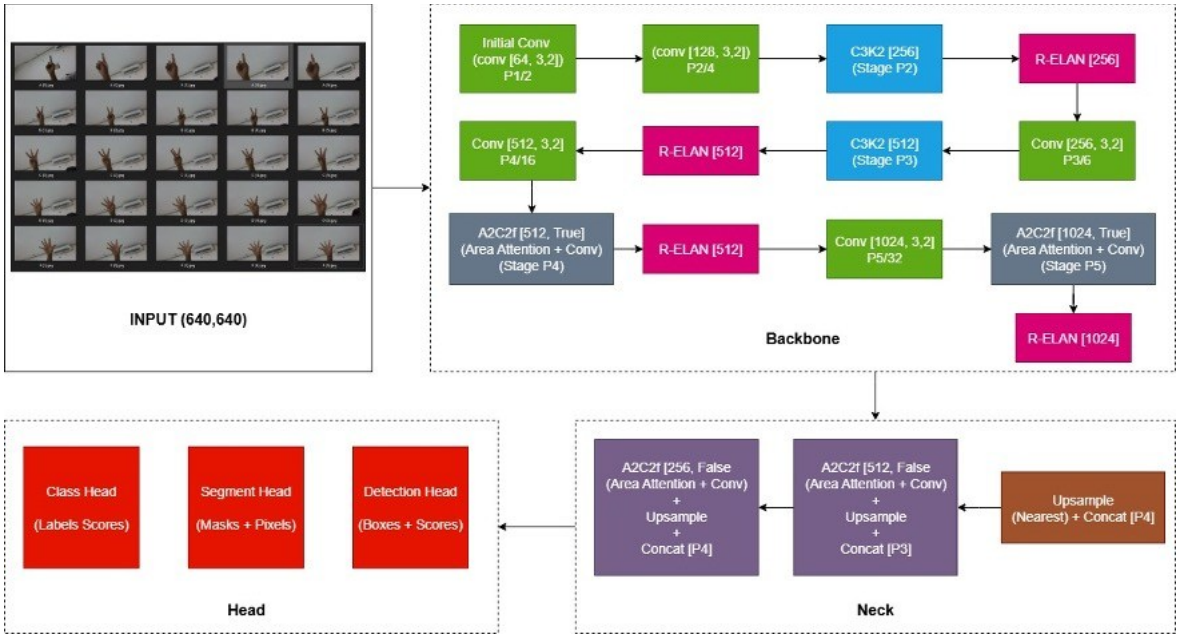


Figure 3. YOLOv12 Architecture

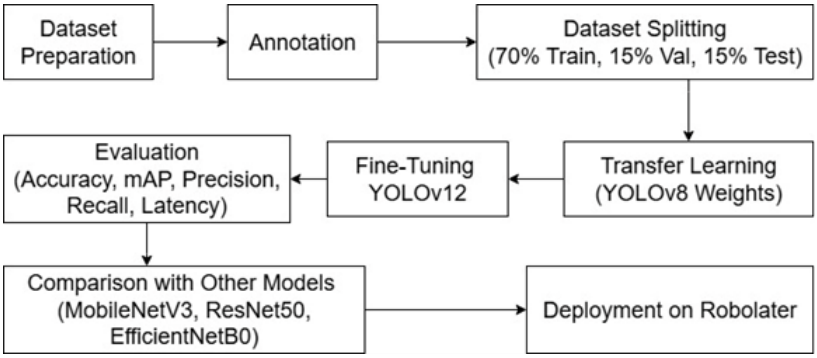


Figure 4. Is a Research Methodology Flowchart

3. RESULT AND DISCUSSION

3.1. Training Model

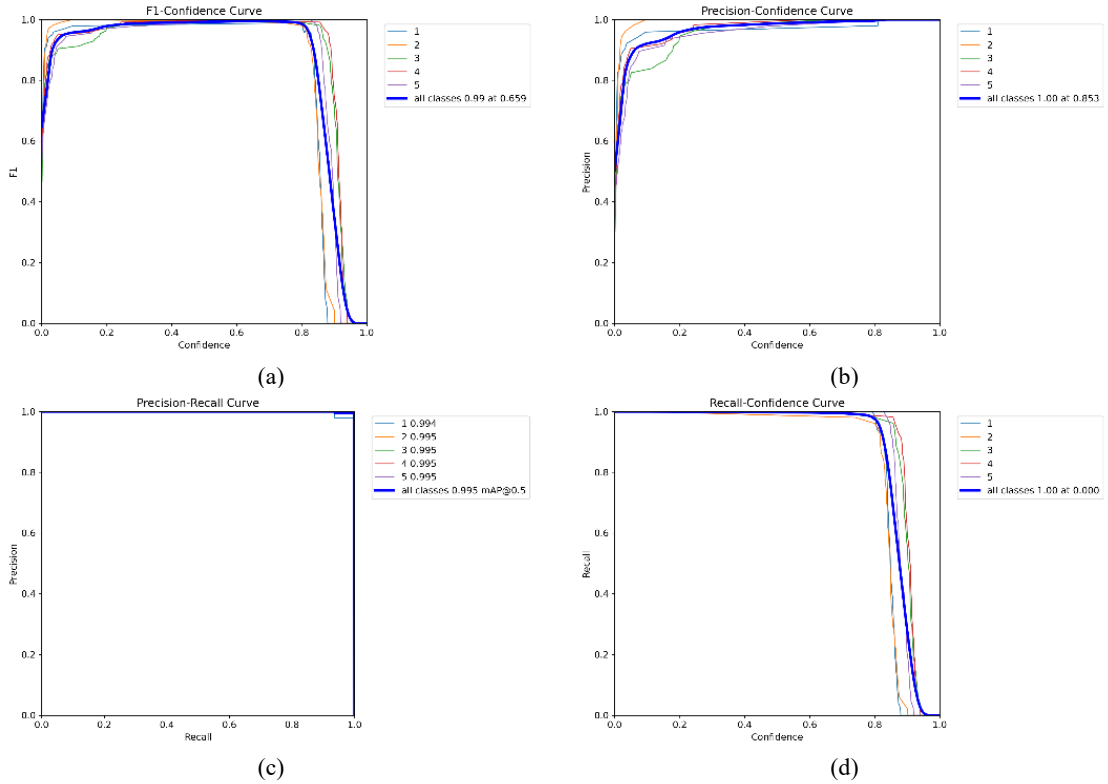
This involves preparing the YOLOv12 model for detecting hand gestures in real-time using a transfer learning approach from the previous model (YOLOv8). An internal gesture dataset, containing 1,025 annotated images across five gesture classes (1, 2, 3, 4, 5), was split into training, validation, and test sets. The training process was run on a GPU machine with augmentation techniques such as flip, brightness/contrast, and light blur, to increase robustness against lighting and camera variation. Figure 5 shows the training accuracy and loss curves, indicating convergence with accuracy above 0.9 for all gesture classes.

The validation results indicate that the YOLOv12 model achieved very high performance across all gesture classes. For class 1, the precision reached 0.975 with a perfect recall of 1.000, resulting in an mAP50 of 0.994 and an mAP50-95 of 0.751. Class 2 showed similar performance, with a precision of 0.981, recall of

1.000, mAP50 of 0.995, and mAP50-95 of 0.779. Furthermore, class 3 achieved the highest precision value of 0.999 with a recall of 1.000, mAP50 of 0.995, and mAP50-95 of 0.790. For class 4, the precision was 0.991 and recall was 0.997, with an mAP50 of 0.995 and mAP50-95 of 0.776. Meanwhile, class 5 demonstrated a precision of 0.989, recall of 1.000, mAP50 of 0.995, and mAP50-95 of 0.826. Overall, the model achieved an average precision of 0.992, recall of 0.996, mAP50 of 0.995, and mAP50-95 of 0.784. These results confirm that YOLOv12 is capable of detecting nearly all hand gestures with very high accuracy and minimal classification errors. The difference between the near-perfect mAP50 and the slightly lower mAP50-95 indicates a limitation in bounding box localization precision, particularly under extreme lighting conditions or tilted camera angles. Nevertheless, with an average inference speed of 1.58 ms per image, this model is highly suitable for real-time detection requirements in service robot systems can be seen in Figure 6.

```
Validating runs/detect/train/weights/best.pt...
Ultralytics 8.3.63 🚀 Python-3.12.11 torch-2.8.0+cu126 CUDA:0 (Tesla T4, 15095MiB)
YOLOv12s summary (fused): 376 layers, 9,076,143 parameters, 0 gradients, 19.3 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 8/8 [00:05<00:00,  1.58it/s]
                   all        250        250      0.992      0.996      0.995      0.784
                     1         47         47      0.975          1      0.994      0.751
                     2         51         51          1      0.983      0.995      0.779
                     3         52         52      0.999          1      0.995       0.79
                     4         57         57          1      0.997      0.995      0.776
                     5         43         43      0.989          1      0.995      0.826
Speed: 0.3ms preprocess, 9.7ms inference, 0.0ms loss, 2.9ms postprocess per image
Results saved to runs/detect/train
```

Figure 5. Model Training Results



(a) (b)

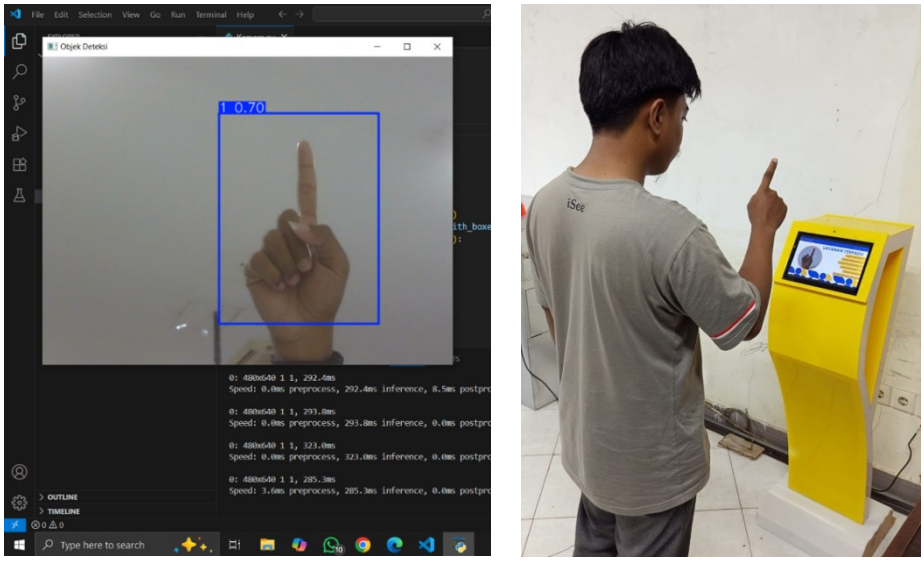
(c) (d)

Figure 6. (a) F1-Confidence Graph, (b) Precision–Confidence graph, (c) Precision–Recall graph, (d) Recall–Confidence Graph

3.2. Experiment and Results Analysis (YOLOv12)

Stage experiment conducted with a running YOLOv12 model on a connected Mini PC with a robot camera (Figure 7). The model was tested in real-world conditions to recognize various hand gestures directly in front of the camera under normal lighting. Test results show that the system can detect hand gestures with good accuracy, indicated by the appearance of colored bounding boxes around the hand region, along with class labels and confidence values displayed above them. In the example shown in Figure 7, the system successfully recognized one of the gestures with a confidence of 0.70. Although this confidence level is lower than the training-validation benchmarks, it remains adequate for triggering robot actions, provided an

operational threshold is applied. This demonstrates that the model is capable of real-time gesture recognition, with an average inference speed of 1–2 ms per image, fully supporting the requirements of service robot applications. Additional gesture detection results are shown in Figure 8.



(a) (b)

Figure 7. (a) Confidence value, (b) Combination with HMI

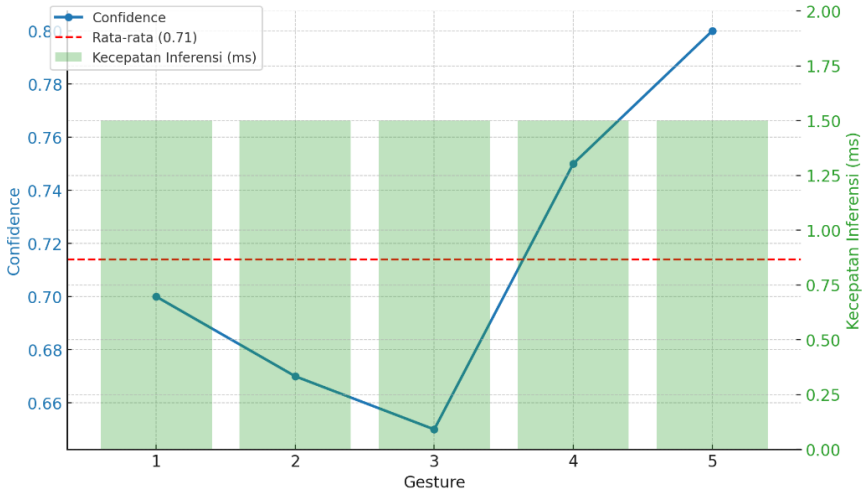


Figure 8. Gesture Detection Results Graph

The YOLOv12 model was trained using transfer learning from YOLOv8 [42], with a gesture dataset of hands with a curated and annotated hand gesture dataset divided into training, validation, and test sets. The training process employed augmentation (flipping, brightness/contrast adjustment, and blurring) to increase robustness under lighting and camera variations. Validation results achieved mAP50 = 0.995 and mAP50-95 = 0.784, confirming that while object detection accuracy is nearly perfect, bounding box localization under extreme conditions (e.g., tilted camera or strong illumination changes) remains a limitation. At the validation stage, precision and recall approached 1.0 across nearly all classes, with class 1 achieving precision = 0.992 and recall = 0.996, and class 4 achieving precision = 0.989 and recall = 1.000. These results confirm stable performance across different confidence levels, with minimal false positives or negatives. Field testing with the robot camera further validated the model's ability to detect gestures accurately, showing bounding boxes and class labels in real-time at around 0.70 confidence. The system maintained an inference speed of 1–2 ms per image, proving its suitability for real-time robotic implementation. In summary, YOLOv12 achieved high accuracy, robust inference speed, and minimal classification errors, though bounding box localization precision still decreases under extreme lighting or angled perspectives.

Future work should focus on expanding the dataset with more environmental variations and exploring advanced fine-tuning strategies (e.g., attention mechanisms, BiFPN integration) to further improve detection precision. Thus, YOLOv12 is a strong candidate for real-time hand gesture recognition in HMI-based robot control systems, with clear potential to evolve into more adaptive solutions in diverse real-world scenarios. To provide a clearer picture of the comprehensive performance of the CNN method, this research also compares three popular architectural models, namely ResNet50, MobileNet, and EfficientNet, with the YOLOv12 model. These three CNN models will be explained in a detailed way, starting from stage training until testing, so that the differences in performance, advantages, and limitations of each can be seen before conducting an analysis comparison with YOLOv12.

3.2.1. ResNet50

In Figure 9 Stage 1, the graph shows a consistent decrease in training loss from around 1.5 to 0.45, accompanied by an increase in training accuracy by ± 0.85. Validation loss also decreased initially, but tends to fluctuate after the 4th epoch, although validation accuracy remains stable and increases until it reaches ±0.75. This indicates a successful model Study with good results, but there are still mild underfitting symptoms because the accuracy validation is not yet as high as the training accuracy. In Figure 10, Stage 2, the performance of ResNet50 improves significantly. Fast training accuracy reached over 0.95 and remained stable, close to 1.0, with a very small loss. However, the validation loss appears to fluctuate sharply (e.g., up to 2.8 in the 4th epoch), although validation accuracy remains relatively high (around 0.85–0.93). Fluctuations indicate the existence of instability in validation data, possibly due to more complex or partial overfitting.

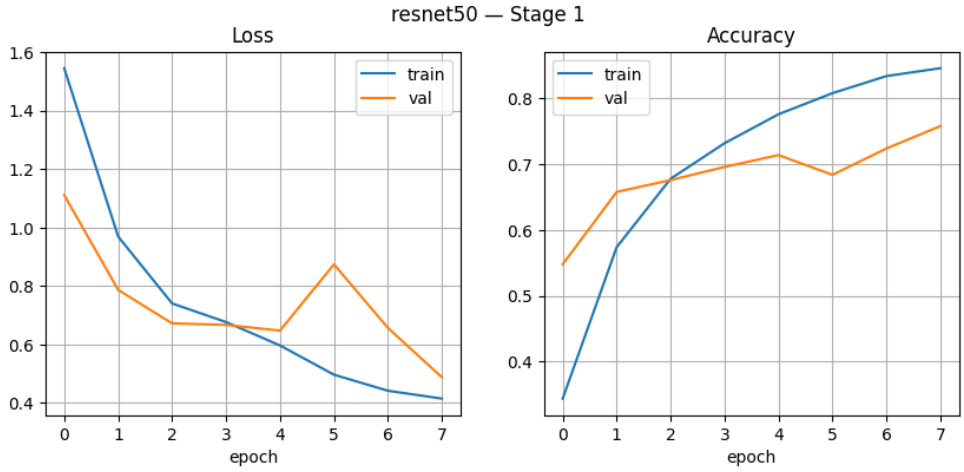


Figure 9. ResNet50 Stage1

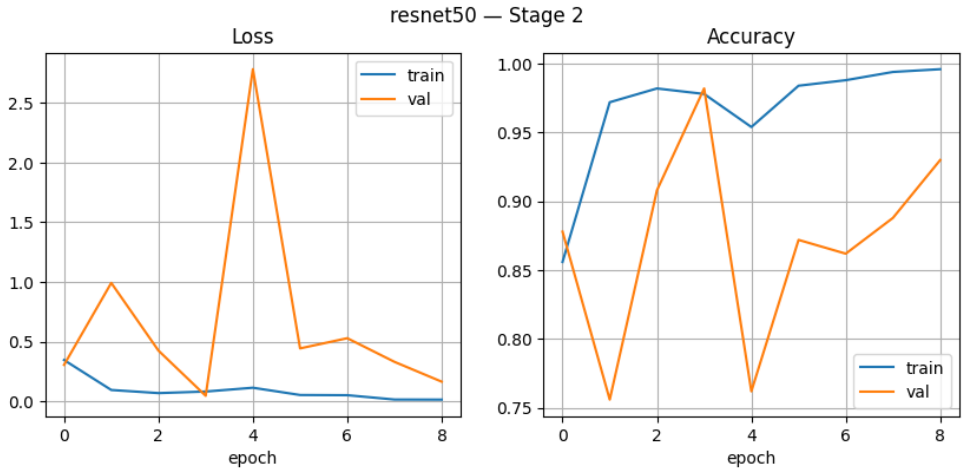


Figure 10. ResNet50 Stage2

3.2.2. MobileNet

In Figure 11, Stage 1, the loss graph shows a consistent decline both on the training data and validation. Training loss decreases from approximately 1.5 to 0.4, while validation loss remains stable at around 0.6. Accuracy increases sharply from the beginning of the epoch, with training accuracy increasing to ±0.85 and validation accuracy reaching around 0.75–0.78. This indicates a learning model with well, though There is still a gap between the training and validation data. In Figure 12, Stage 2, the performance of MobileNetV2 improves significantly. Training accuracy reaches nearly 1.0 with fast reach, while validation accuracy remains stable in the range of 0.92–0.94. Training loss decreases drastically until almost 0, and validation loss remains in the range of 0.15–0.20 with slight fluctuations. This shows a successful model Study in a way that is efficient with good generalization, as well as no severe overfitting symptoms, although greater training accuracy tall from validation.

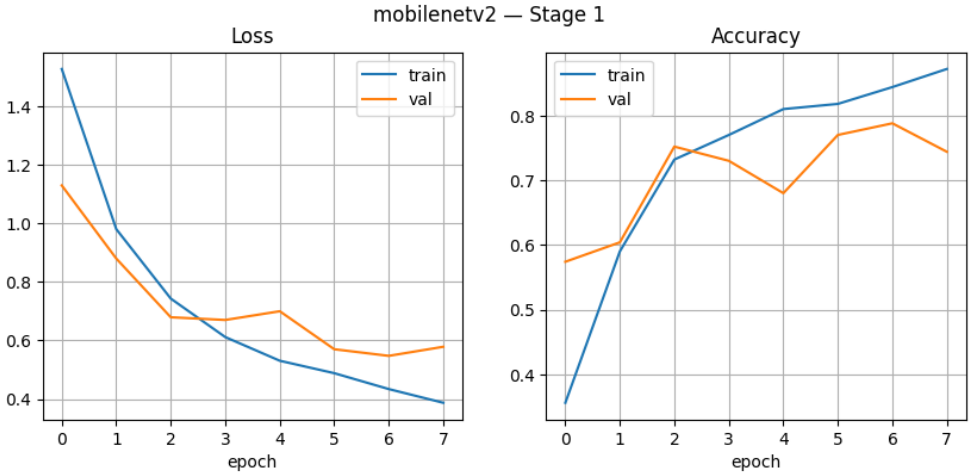


Figure 11. MobileNet Stage1

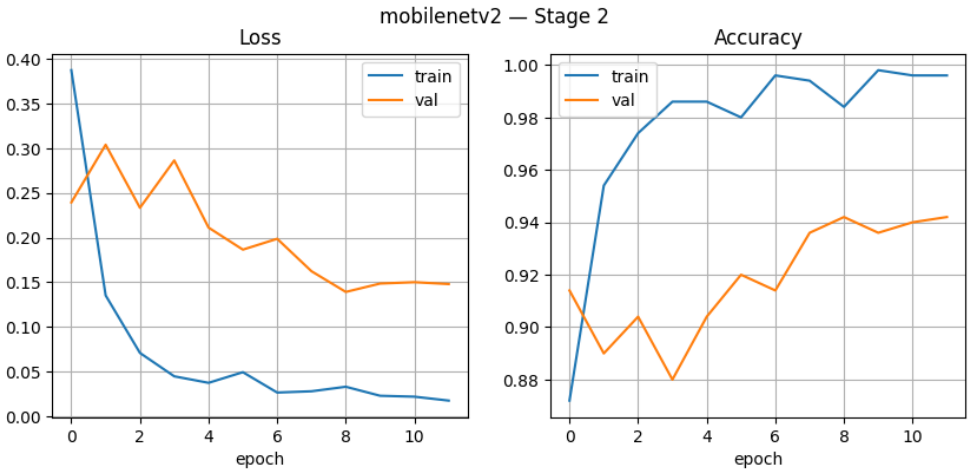


Figure 12. MobileNet Stage2

3.2.3. EfficientNet

In Figure 13, Stage 1, the training loss and validation loss exhibit a consistent decreasing pattern throughout the epoch, with the training loss decreasing from approximately 1.4 to 0.55 and the validation loss decreasing from 1.1 to 0.54. The accuracy graph also shows a steady increase, with the training accuracy reaching around 0.85 and the validation accuracy approaching 0.80. This performance demonstrates that the model can learn well without significant differences between the training and validation data, thus maintaining a relatively high level of generalization. In Figure 14, Stage 2, the performance of EfficientNetB0 significantly improved, with training accuracy approaching 1.0 and validation accuracy stable in the range of 0.97–0.99. Training loss decreased to <0.05, and validation loss remained low at around 0.07–0.10, indicating the model

is highly efficient at learning data patterns. Unlike ResNet50, which tends to fluctuate during validation, EfficientNetB0 is more stable in both loss and accuracy, indicating better generalization capabilities.

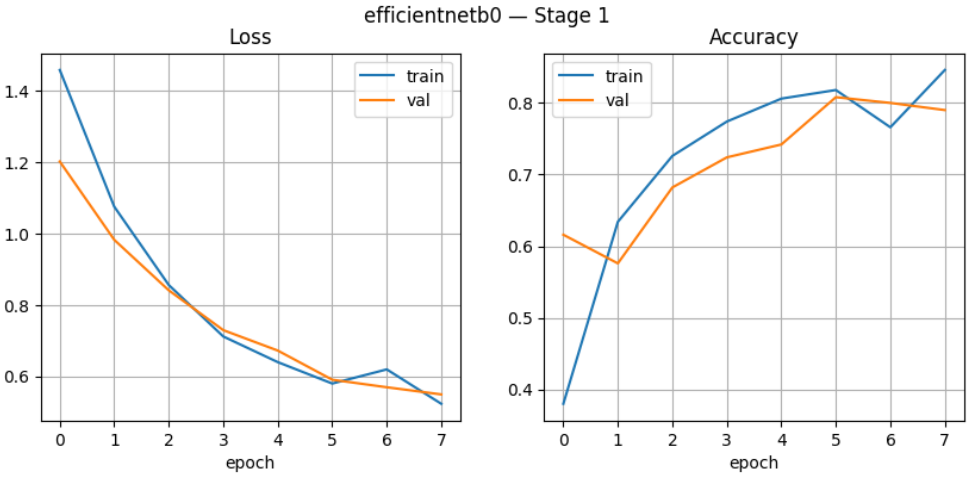


Figure 13. EfficientNet Stage 1

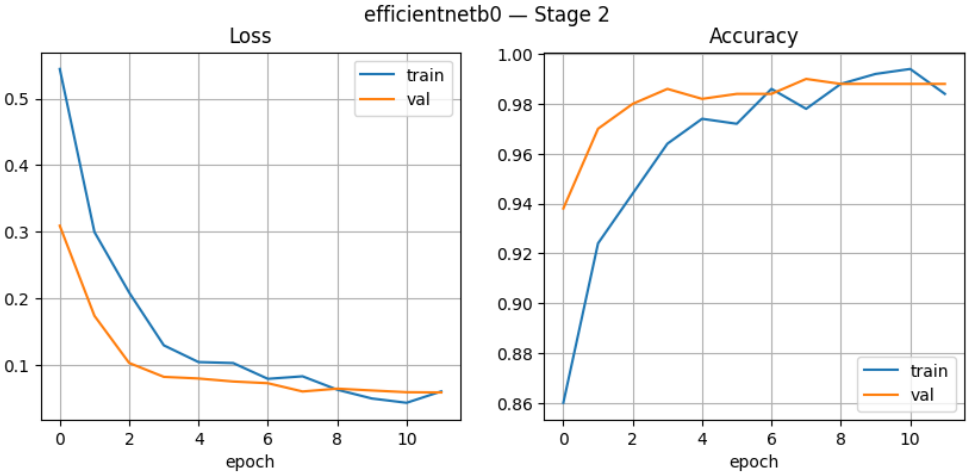


Figure 14. EfficientNet Stage2

3.3. Analysis Comparison of the YoloV12 Model with Three Other Models

Test results in Figure 15 show that the three CNN models, namely ResNet50, MobileNetV2, and EfficientNetB0, have varying performance in detecting five classes of objects. The ResNet50 model consistently yields the highest confidence value compared to the other two models. For example, in class 1, it reaches around 0.80 and remains above 0.60 in most cases, particularly for big classes, although it decreases to 0.44 in class 5. The EfficientNetB0 model shows relatively stable performance with a confidence value in the range of 0.41–0.59. Although not as high as ResNet50, this model is capable of striking a balance between accuracy and efficiency. While that, MobileNetV2 tends to generate more confidence low, namely in the range of 0.37–0.55, but show quite an improvement good in grade 4.

Testing four CNN models reveals that each has distinct advantages and limitations in detecting hand gestures in real-time. **YOLOv12** achieves its best performance with very high accuracy (mAP50 = 0.995), an average inference speed of 1–2 ms/image, and stable detection performance even under varying lighting conditions. **ResNet50** achieves the best accuracy on the test data, with the highest confidence value (reaching 0.80 in certain classes), but its validation tends to fluctuate, thereby risking overfitting. EfficientNetB0 exhibits an optimal balance between accuracy and generalization, with stable validation accuracy values ranging from 0.97 to 0.99 and an average confidence of 0.41 to 0.59 on the test data, making it a reliable choice with good stability. Meanwhile, **MobileNetV2** shows less computationally demanding performance and is suitable for

devices with limited resources, although its detection confidence is relatively lower (0.37–0.55) compared to the other three models can be seen in Table 1.

Thus, **YOLOv12** is the ideal model for implementing a service robot-based hand gesture recognition system because it combines high accuracy with real-time speed. **ResNet50** is more suitable for scenarios requiring high classification accuracy, and **EfficientNetB0** excels for applications that require a balance between performance and efficiency. At the same time, **MobileNetV2** can be optimized for portable applications or devices with computing constraints. Thus, YOLOv12 is the most ideal model for implementing a service robot-based hand gesture recognition system because it combines high accuracy with real-time speed. ResNet50 is more suitable for scenarios requiring high classification accuracy, EfficientNetB0 excels for applications that require a balance between performance and efficiency, while MobileNetV2 can be optimized for portable applications or devices with computing constraints.

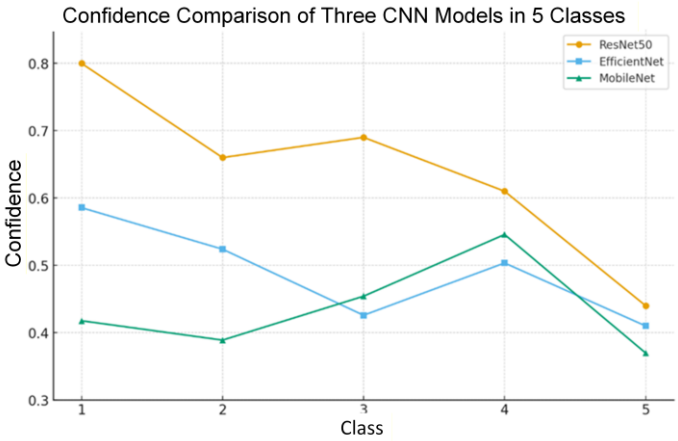


Figure 15. Test Results of 3 CNN Models

Table 1. Comparison of Data of 4 CNN Models

Gesture	Confidence			
	YOLOv12	ResNet50	EfficientNetB0	MobileNetV2
1	0.70	0.8	0.59	0.41
2	0.67	0.65	0.51	0.39
3	0.65	0.69	0.43	0.45
4	0.75	0.61	0.5	0.54
5	0.8	0.65	0.41	0.38

4. CONCLUSIONS

This study demonstrates that applying the transfer learning method to the CNN architecture, particularly with the development of the YOLOv12 model, significantly improves the effectiveness of real-time hand gesture recognition in service robots. Unlike previous works that mainly applied YOLOv5 or YOLOv7, this study introduces a modified YOLOv12 framework based on YOLOv8 weights, specifically optimized for small-object recognition and robust inference in dynamic environments. The experimental results confirm that the proposed model achieves a very high level of accuracy (mAP50 = 0.995, mAP50-95 = 0.784) with an average inference speed of 1–2 ms per image, enabling real-time deployment. Compared with benchmark CNN architectures such as ResNet50, MobileNetV2, and EfficientNetB0, YOLOv12 consistently demonstrated greater stability under varying lighting conditions and achieved lower error rates, highlighting its superiority for service-robot applications. The main novelty of this research lies in demonstrating, for the first time, that a YOLOv12 modification can be effectively adapted for human–robot interaction tasks in higher education environments. This contribution provides a strong academic foundation for integrating advanced transfer learning models into intelligent robotic systems. However, the study also identifies limitations in bounding box localization under extreme environmental variations, which emphasizes the need for further improvements. Future work should focus on expanding the dataset with more diverse conditions, exploring attention-based feature enhancement and BiFPN-based multi-scale fusion, and investigating edge-device deployment for on-board processing. Moreover, integrating multimodal inputs (e.g., RGB-D sensors, wearable signals) and applying federated learning approaches could enhance robustness, privacy, and scalability. Overall, this

research not only validates the feasibility of YOLOv12 for real-time gesture recognition in robotic systems but also opens opportunities for future applications in adaptive, intelligent services aligned with SDG 4 (Quality Education), SDG 9 (Industry, Innovation, and Infrastructure), and SDG 11 (Sustainable Cities and Communities).

DECLARATION

Supplementary Materials

The writer wants to express gratitude and sincere love to the Ministry of Higher Education, Science and Technology of the Republic of Indonesia for the funding and support provided for this study. The author also thanks Universitas Trunojoyo Madura for the facilities, resources, power, and support that are invaluable during the survey.

Author Contribution

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Faikul Umam	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Ach. David	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	
Hanifudin Sukri	✓	✓	✓	✓			✓			✓	✓		✓	✓
Firman Maolana			✓	✓	✓		✓	✓			✓			
Adi Andriansyah				✓	✓		✓			✓		✓		✓
Ahmad Yusuf					✓		✓		✓		✓	✓		✓

C : **Conceptualization** I : **Investigation** Vi : **Visualization**
M : **Methodology** R : **Resources** Su : **Su** pervision
So : **So** ftware D : **Data** Curation P : **Project** administration
Va : **Validation** O : Writing - **Original** Draft Fu : **Funding** acquisition
Fo : **Formal** analysis E : Writing - Review & **E** diting

Funding

The writer expresses love to the Directorate of Research, Technology, and Community Service to the Community (DRTPM) Ministry of Higher Education, Science, and Technology, which has given financial support through the 2025 Fundamental Research Scheme contract number B/033/UN46.1/PT.01.03/BIMA/PL/2025.

Data Availability

The data available is relevant to this paper because the data used is a development of previous research entitled Optimization of Hand Gesture Object Detection Using Fine-Tuning Techniques on an Integrated Service of Smart Robot.

REFERENCES

[1] S. Padmakala, S. O. Husain, E. Poornima, P. Dutta, and M. Soni, "Hyperparameter Tuning of Deep Convolutional Neural Network for Hand Gesture Recognition," in *2024 Second International Conference on Networks, Multimedia and Information Technology (NMITCON)*, pp. 1–4, 2024, https://doi.org/10.1109/NMITCON62075.2024.10698984.

[2] K. L. Manikanta, N. Shyam, and S. S, "Real-Time Hand Gesture Recognition and Sentence Generation for Deaf and Non-Verbal Communication," in *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, pp. 1–6, 2024, https://doi.org/10.1109/ADICS58448.2024.10533502.

[3] J. W. Smith, S. Thiagarajan, R. Willis, Y. Makris, and M. Torlak, "Improved Static Hand Gesture Classification on Deep Convolutional Neural Networks Using Novel Sterile Training Technique," *IEEE Access*, vol. 9, pp. 10893–10902, 2021, https://doi.org/10.1109/ACCESS.2021.3051454.

[4] S. Meshram, R. Singh, P. Pal, and S. K. Singh, "Convolution Neural Network based Hand Gesture Recognition System," in *2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, pp. 1–5, 2023, https://doi.org/10.1109/ICAECT57570.2023.10118267.

[5] R. Bekiri, S. Babahenini, and M. C. Babahenini, "Transfer Learning for Improved Hand Gesture Recognition with Neural Networks," in *2024 8th International Conference on Image and Signal Processing and their Applications (ISPA)*, pp. 1–6, 2024, https://doi.org/10.1109/ISPA59904.2024.10536712.

[6] U. Kulkarni, S. Agasimani, P. P. Kulkarni, S. Kabadi, P. S. Aditya, and R. Ujawane, "Vision based Roughness Average Value Detection using YOLOv5 and EasyOCR," in *2023 IEEE 8th International Conference for Convergence in Technology (I2CT)*, pp. 1–7, 2023, https://doi.org/10.1109/I2CT57861.2023.10126305.

[7] S. Dhyani and V. Kumar, "Real-Time License Plate Detection and Recognition System using YOLOv7x and EasyOCR," in *2023 Global Conference on Information Technologies and Communications (GCITC)*, pp. 1–5, 2023, https://doi.org/10.1109/GCITC60406.2023.10425814.
[8] W. Fang, L. Wang, and P. Ren, "Tinier-YOLO: A Real-Time Object Detection Method for Constrained Environments," *IEEE Access*, vol. 8, pp. 1935–1944, 2020, https://doi.org/10.1109/ACCESS.2019.2961959.
[9] P. Samothai, P. Sanguansat, A. Kheaksong, K. Srisomboon, and W. Lee, "The Evaluation of Bone Fracture Detection of YOLO Series," in *2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, pp. 1054–1057, 2022, https://doi.org/10.1109/ITC-CSCC55581.2022.9895016.
[10] T.-H. Nguyen, R. Scherer, and V.-H. Le, "YOLO Series for Human Hand Action Detection and Classification from Egocentric Videos," *Sensors*, vol. 23, p. 3255, 2023, https://doi.org/10.3390/s23063255.
[11] P. Shourie, V. Anand, D. Upadhyay, S. Devliyal, and S. Gupta, "YogaPoseVision: MobileNetV3-Powered CNN for Yoga Pose Identification," in *2024 7th International Conference on Circuit Power and Computing Technologies (ICCPCT)*, pp. 659–663, 2024, https://doi.org/10.1109/ICCPCT61902.2024.10673072.
[12] S. Guanglei and W. Wen, "Recognizing Crop Diseases and Pests Using an Improved MobileNetV3 Model," in *2023 13th International Conference on Information Technology in Medicine and Education (ITME)*, pp. 302–308, 2023, https://doi.org/10.1109/ITME60234.2023.00069.
[13] X. Zhang, N. Li, and R. Zhang, "An Improved Lightweight Network MobileNetv3 Based YOLOv3 for Pedestrian Detection," in *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pp. 114–118, 2021, https://doi.org/10.1109/ICCECE51280.2021.9342416.
[14] G. Singh, K. Guleria, and S. Sharma, "A Fine-Tuned MobileNetV3 Model for Real and Fake Image Classification," in *2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*, pp. 1–5, 2024, https://doi.org/10.1109/ICoICI62503.2024.10696448.
[15] N. Anggraini, S. H. Ramadhani, L. K. Wardhani, N. Hakiem, I. M. Shofi, and M. T. Rosyadi, "Development of Face Mask Detection using SSDLite MobilenetV3 Small on Raspberry Pi 4," in *2022 5th International Conference of Computer and Informatics Engineering (IC2IE)*, pp. 209–214, 2022, https://doi.org/10.1109/IC2IE56416.2022.9970078.
[16] R. Pillai, N. Sharma, and R. Gupta, "Detection & Classification of Abnormalities in GI Tract through MobileNetV3 Transfer Learning Model," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–6, 2023, https://doi.org/10.1109/ICCCNT56998.2023.10307732.
[17] S. Saifullah, R. dreżewski, A. Yudhana, and A. P. Suryotomo, "Automatic Brain Tumor Segmentation: Advancing U-Net With ResNet50 Encoder for Precise Medical Image Analysis," *IEEE Access*, vol. 13, pp. 43473–43489, 2025, https://doi.org/10.1109/ACCESS.2025.3547430.
[18] A. F. A. Alshamrani and F. Saleh Zuhair Alshomrani, "Optimizing Breast Cancer Mammogram Classification Through a Dual Approach: A Deep Learning Framework Combining ResNet50, SMOTE, and Fully Connected Layers for Balanced and Imbalanced Data," *IEEE Access*, vol. 13, pp. 4815–4826, 2025, https://doi.org/10.1109/ACCESS.2024.3524633.
[19] R. Singh, S. Gupta, S. Bharany, A. Almogren, A. Altameem, and A. Ur Rehman, "Ensemble Deep Learning Models for Enhanced Brain Tumor Classification by Leveraging ResNet50 and EfficientNet-B7 on High-Resolution MRI Images," *IEEE Access*, vol. 12, pp. 178623–178641, 2024, https://doi.org/10.1109/ACCESS.2024.3494232.
[20] S. Tamilselvi, M. Suchetha, and R. Raman, "Leveraging ResNet50 With Swin Attention for Accurate Detection of OCT Biomarkers Using Fundus Images," *IEEE Access*, vol. 13, pp. 35203–35218, 2025, https://doi.org/10.1109/ACCESS.2025.3544332.
[21] N. Ansori, S. S. Putro, S. Rifka, E. M. S. Rochman, Y. P. Asmara, and A. Rachmad, "The Effect of Color and Augmentation on Corn Stalk Disease Classification Using ResNet-101," *Mathematical Modelling of Engineering Problems*, vol. 12, no. 3, pp. 1007–1012, 2025, https://doi.org/10.18280/mmep.120327.
[22] A. Younis *et al.*, "Abnormal Brain Tumors Classification Using ResNet50 and Its Comprehensive Evaluation," *IEEE Access*, vol. 12, pp. 78843–78853, 2024, https://doi.org/10.1109/ACCESS.2024.3403902.
[23] Z. Liu *et al.*, "Research on Arthroscopic Images Bleeding Detection Algorithm Based on ViT-ResNet50 Integrated Model and Transfer Learning," *IEEE Access*, vol. 12, pp. 181436–181453, 2024, https://doi.org/10.1109/ACCESS.2024.3508797.
[24] N. Saba *et al.*, "A Synergistic Approach to Colon Cancer Detection: Leveraging EfficientNet and NSGA-II for Enhanced Diagnostic Performance," *IEEE Access*, pp. 1–1, 2024, https://doi.org/10.1109/ACCESS.2024.3519216.
[25] Z. Liu, J. John, and E. Agu, "Diabetic Foot Ulcer Ischemia and Infection Classification Using EfficientNet Deep Learning Models," *IEEE Open J Eng Med Biol*, vol. 3, pp. 189–201, 2022, https://doi.org/10.1109/OJEMB.2022.3219725.
[26] Z. Liu, J. John, and E. Agu, "Diabetic Foot Ulcer Ischemia and Infection Classification Using EfficientNet Deep Learning Models," *IEEE Open J Eng Med Biol*, vol. 3, pp. 189–201, 2022, https://doi.org/10.1109/OJEMB.2022.3219725.
[27] A. U. Nabi, J. Shi, Kamlesh, A. K. Jumani, and J. Ahmed Bhutto, "Hybrid Transformer-EfficientNet Model for Robust Human Activity Recognition: The BiTransAct Approach," *IEEE Access*, vol. 12, pp. 184517–184528, 2024, https://doi.org/10.1109/ACCESS.2024.3506598.

[28] C. Huang, W. Wang, X. Zhang, S.-H. Wang, and Y.-D. Zhang, "Tuberculosis Diagnosis Using Deep Transferred EfficientNet," *IEEE/ACM Trans Comput Biol Bioinform*, vol. 20, no. 5, pp. 2639–2646, 2023, https://doi.org/10.1109/TCBB.2022.3199572.
[29] L. Li, Q. Yin, X. Wang, and H. Wang, "Defects Localization and Classification Method of Power Transmission Line Insulators Aerial Images Based on YOLOv5 EfficientNet and SVM," *IEEE Access*, vol. 13, pp. 74833–74843, 2025, https://doi.org/10.1109/ACCESS.2025.3559657.
[30] J. Padhi, L. Korada, A. Dash, P. K. Sethy, S. K. Behera, and A. Nanthaamornphong, "Paddy Leaf Disease Classification Using EfficientNet B4 With Compound Scaling and Swish Activation: A Deep Learning Approach," *IEEE Access*, vol. 12, pp. 126426–126437, 2024, https://doi.org/10.1109/ACCESS.2024.3451557.
[31] P. Kumar Tiwary, P. Johri, A. Katiyar, and M. K. Chhipa, "Deep Learning-Based MRI Brain Tumor Segmentation With EfficientNet-Enhanced UNet," *IEEE Access*, vol. 13, pp. 54920–54937, 2025, https://doi.org/10.1109/ACCESS.2025.3554405.
[32] H. Han, H. Liu, C. Yang, and J. Qiao, "Transfer Learning Algorithm With Knowledge Division Level," *IEEE Trans Neural Netw Learn Syst*, vol. 34, no. 11, pp. 8602–8616, 2023, https://doi.org/10.1109/TNNLS.2022.3151646.
[33] Y. Ma, S. Chen, S. Ermon, and D. B. Lobell, "Transfer learning in environmental remote sensing," *Remote Sens Environ*, vol. 301, p. 113924, 2024, https://doi.org/10.1016/j.rse.2023.113924.
[34] Z. Zhao, L. Alzubaidi, J. Zhang, Y. Duan, and Y. Gu, "A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations," *Expert Syst Appl*, vol. 242, p. 122807, 2024, https://doi.org/10.1016/j.eswa.2023.122807.
[35] S. A. Serrano, J. Martinez-Carranza, and L. E. Sucar, "Knowledge Transfer for Cross-Domain Reinforcement Learning: A Systematic Review," *IEEE Access*, vol. 12, pp. 114552–114572, 2024, https://doi.org/10.1109/ACCESS.2024.3435558.
[36] J. Pordoy *et al.*, "Multi-Frame Transfer Learning Framework for Facial Emotion Recognition in e-Learning Contexts," *IEEE Access*, vol. 12, pp. 151360–151381, 2024, https://doi.org/10.1109/ACCESS.2024.3478072.
[37] Z. Zhu, K. Lin, A. K. Jain, and J. Zhou, "Transfer Learning in Deep Reinforcement Learning: A Survey," *IEEE Trans Pattern Anal Mach Intell*, vol. 45, no. 11, pp. 13344–13362, 2023, https://doi.org/10.1109/TPAMI.2023.3292075.
[38] H. Han *et al.*, "SWIPTNet: A Unified Deep Learning Framework for SWIPT Based on GNN and Transfer Learning," *IEEE Trans Mob Comput*, vol. 24, no. 10, pp. 9477–9488, 2025, https://doi.org/10.1109/TMC.2025.3563892.
[39] Q. Song, Y.-J. Zheng, J. Yang, Y.-J. Huang, W.-G. Sheng, and S.-Y. Chen, "Predicting Demands of COVID-19 Prevention and Control Materials via Co-Evolutionary Transfer Learning," *IEEE Trans Cybern*, vol. 53, no. 6, pp. 3859–3872, 2023, https://doi.org/10.1109/TCYB.2022.3164412.
[40] S. H. Waters and G. D. Clifford, "Physics-Informed Transfer Learning to Enhance Sleep Staging," *IEEE Trans Biomed Eng*, vol. 71, no. 5, pp. 1599–1606, 2024, https://doi.org/10.1109/TBME.2023.3345888.
[41] L. Cheng, P. Singh, and F. Ferranti, "Transfer Learning-Assisted Inverse Modeling in Nanophotonics Based on Mixture Density Networks," *IEEE Access*, vol. 12, pp. 55218–55224, 2024, https://doi.org/10.1109/ACCESS.2024.3383790.
[42] F. Umam, H. Sukri, A. Dafid, F. Maolana, M. Natalis, and S. Ndruru, "Optimization of Hand Gesture Object Detection Using Fine-Tuning Techniques on an Integrated Service of Smart Robot," *International Journal of Industrial Engineering & Production Research*, vol. 35, no. 4, pp. 1–11, 2024, https://doi.org/10.22068/ijiepr.35.4.2141.
[43] B. La and H. Wu, "The Few-Shot YOLOv12 Object Detection Method Based on Dual-Path Cosine Feature Alignment," in *2025 6th International Conference on Computer Vision, Image and Deep Learning (CVIDL)*, pp. 318–324, 2025, https://doi.org/10.1109/CVIDL65390.2025.11085796.
[44] P. B. Rana and S. Thapa, "Comparative Study of Object Detection Models for Fresh and Rotten Apples and Tomatoes: Faster R-CNN, DETR, YOLOv8, and YOLOv12S," in *2025 International Conference on Inventive Computation Technologies (ICICT)*, pp. 660–667, 2025, https://doi.org/10.1109/ICICT64420.2025.11005373.
[45] S. H. Thai, H. N. Viet, H. N. Thu, N. Nguyen Van, and H. Dang Van, "AI-Driven Real-Time Traffic Management Using YOLOv12 for Dynamic Signal Control and Intelligent Surveillance," in *2025 4th International Conference on Electronics Representation and Algorithm (ICERA)*, pp. 92–97, 2025, https://doi.org/10.1109/ICERA66156.2025.11087320.
[46] A. B. Shaik, A. K. Kandula, G. K. Tirumalasetti, B. Yendluri, and H. K. Kalluri, "Comparative Analysis of YOLOv11 and YOLOv12 for Automated Weed Detection in Precision Agriculture," in *2025 5th International Conference on Pervasive Computing and Social Networking (ICPCSN)*, pp. 787–793, 2025, https://doi.org/10.1109/ICPCSN65854.2025.11036078.
[47] A. M. El-Kafrawy and E. H. Seddik, "Personal Protective Equipment (PPE) Monitoring for Construction Site Safety using YOLOv12," in *2025 International Conference on Machine Intelligence and Smart Innovation (ICMISI)*, pp. 456–459, 2025, https://doi.org/10.1109/ICMISI65108.2025.11115450.
[48] N. Simic and A. Gavrovska, "Comparative Analysis of YOLOv11 and YOLOv12 for AI-Powered Aerial People Detection," in *2025 12th International Conference on Electrical, Electronic and Computing Engineering (IcETRAN)*, pp. 1–4, 2025, https://doi.org/10.1109/IcETRAN66854.2025.11114306.
[49] H. S. Patel, D. K. Patel, D. K. Patel, M. K. Patel, and M. Darji, "Enhancing Brain Tumor Detection Using YOLOv12," in *2025 3rd International Conference on Inventive Computing and Informatics (ICICI)*, pp. 1–7, 2025, https://doi.org/10.1109/ICICI65870.2025.11069471.

[50] C. Gong and M. Fang, "An Automated Detection System for Magnetic Bead Residues in Cell Therapy Products Based on YOLOv12," in *2025 7th International Conference on Artificial Intelligence Technologies and Applications (ICAITA)*, pp. 479–483, 2025, https://doi.org/10.1109/ICAITA67588.2025.11137958.

AUTHOR BIOGRAPHY

Faikul Umam, Department of Mechatronics Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia.
Email: faikul@trunojoyo.ac.id
Scopus ID: 57189687586
Orcid ID: 0000-0001-8077-2825

Ach. Dafid, Department of Mechatronics Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia.
Email: ach.dafid@trunojoyo.ac.id
Scopus ID: 57218395867
Orcid ID: 0000-0002-6711-8367

Hanifudin Sukri, Department of Information System, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia.
Email: hanifudinsukri@trunojoyo.ac.id
Scopus ID: 57201856188
Orcid ID: 0000-0002-5017-8611

Yuli Panca Asmara, INTI International University, FEQS, Nilai, Malaysia.
Email: ypanca@hotmail.com
Scopus ID: 37114158200
Orcid ID: 0000-0001-6930-0771

Md Monzur Morshed, University of Dhaka, Dhaka-1000, Bangladesh.
Email: mmmorshed@yahoo.com
Scopus ID: 58782159400
Orcid ID: 0009-0005-1717-778X

Firman Maolana, Student of Department of Mechatronics Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia
Email: 210491100001@student.trunojoyo.ac.id
Scopus ID: -
Orcid ID: -

Ahcmad Yusuf, Student of Department of Mechatronics Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia
Email: 210491100009@student.trunojoyo.ac.id
Scopus ID: -
Orcid ID: -