International Journal of Industrial Engineering & Production Research December 2024 Vol. 35, No. 4: 1-11
DOI: 10.22068/ijiepr.35.4.2141

RESEARCH PAPER

Optimization of Hand Gesture Object Detection Using Fine-Tuning Techniques on an Integrated Service of Smart Robot

Faikul Umam[1*], Hanifudin Sukri[2], Ach Dafid[3], Firman Maolana[4] & Mycel Natalis Stopper Ndruru[5]

Received 10 Aguste 2024; Revised 25 September 2024; Accepted 1 November 2024;

ABSTRACT

Robots are one of the testbeds that can be used as objects for the application of intelligent systems in the current era of Industry 4.0. With such systems, robots can interact with humans through perception (sensors) like cameras. Through this interaction, it is expected that robots can assist humans in providing reliable and efficient service improvements. In this research, the robot collects data from the camera, which is then processed using a Convolutional Neural Network (CNN). This approach is based on the adaptive nature of CNN in recognizing visuals captured by the camera. In its application, the robot used in this research is a humanoid model named Robolater, commonly known as the Integrated Service Robot. The fundamental reason for using a humanoid robot model is to enhance human-robot interaction, aiming to achieve better efficiency, reliability, and quality. The research begins with the implementation of hardware and software so that the robot can recognize human movements through the camera sensor. The robot is trained to recognize hand gestures using the Convolutional Neural Network method, where the deep learning algorithm, as a supervised type, can recognize movements through visual inputs. At this stage, the robot is trained with various weights, backbones, and detectors. The results of this study show that the F-T Last Half technique exhibits more stable performance compared to other techniques, especially with larger input scales (640×644). The model using this technique achieved a mAP of 91.6%, with a precision of 84.6%, and a recall of 80.6%.

KEYWORDS: Optimization; Object detection; Hand gestures; Fine-Tuning; Smart robot.

1. Introduction

In the era of Industry 4.0, technology is rapidly evolving, and the need for intelligent automation is increasing. Higher education institutions face challenges in providing high-quality and efficient services to the academic community and the public at large [1], [2].One way to improve these services is by utilizing robotics technology that can interact with humans more intuitively. One such technology is Hand Gesture Detection. The use of robots has great potential to enhance efficiency in various aspects of service provision in higher education institutions, such as academic services, general services, student services, and other administrative services.

The Convolutional Neural Network (CNN) approach has proven capable of processing and understanding images efficiently, enabling hand gesture recognition with high accuracy [3], [4], [5], [6][7], [8], [9]. Using CNN, the system can be trained to recognize various hand gestures commonly used in human interaction, such as raising a hand to ask a question, giving signals, or pointing directions [10], [11], [12], [13], [14], [15], [16]. The CNN approach has been previously implemented in our research, including body movement detection with 92% accuracy and fish species detection [17], [18], [19], [20], [21], [22], [23]. The relevance to this research lies in developing a system that allows the robot to interact effectively with humans, in line to improve services at higher education institutions.

* **Corresponding author: Faikul Umam**
faikul@trunojoyo.ac.id

1. ***Department of Mechatronics Engineering, Faculty of Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia***.
2. ***Departement of Information System, Faculty of Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia.***
3. ***Department of Mechatronics Engineering, Faculty of Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia.***
4. ***Department of Mechatronics Engineering, Faculty of Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia.***
5. ***Department of Mechatronics Engineering, Faculty of Engineering, Universitas Trunojoyo Madura, Bangkalan 69162, Indonesia.***

Although hand gesture recognition technology has existed before, many systems still use conventional approaches that have limitations in terms of accuracy and response speed. This research aims to address that gap by implementing the CNN approach. In this study, CNN will be implemented on an Integrated Service Robot named Robolater. This robot will recognize and process all gestures conveyed by visitors through a camera sensor. The camera sensor's detection results will be processed by the intelligent system embedded in the robot.

Thus, the integrated smart service robot system can respond appropriately to commands or needs conveyed by users through hand gestures [24], [25], thereby enhancing efficiency and convenience in various activities related to services within the university environment."

2. Methods

The design of the Robolater was carried out using fiber material. This robot is equipped with several main components, including a monitor screen for visual display, a microcontroller to control the system, and speakers for audio output that deliver the provided services. Once the robot was designed and built, as shown in the robot part design in Figure 1, the next step was to calibrate each sensor, such as the servo and camera. This calibration process ensures that every component functions according to its initial purpose. Figure 2 shows the circuit used in this research, where the Raspberry Pi is connected to several servos, sensors, and other modules that support the overall functionality of the system.

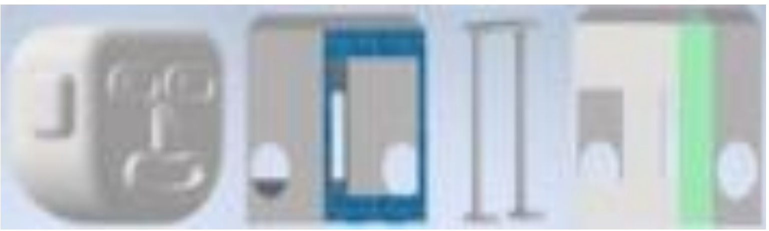

Fig. 1. Design of each part of robolater

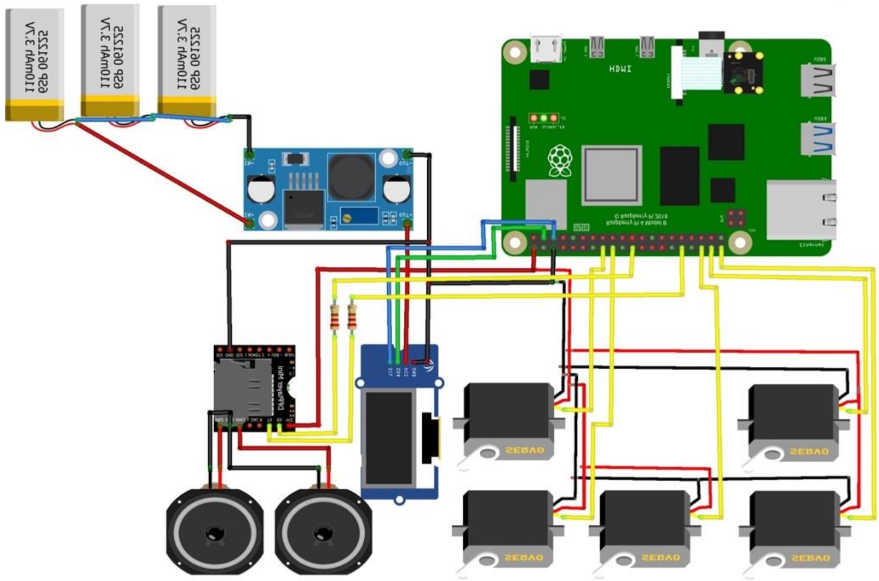

Fig. 2. Robolater electronics circuit

2.1. Convolutional neural network approach

Deep learning is a sub-field of artificial intelligence that involves self-learning from data. This sub-field has seen massive growth [26]. Through the integrated service robot, used as a tool to improve integrated services for the academic community, as illustrated in Figure 1, a camera is used for vision detection to capture visuals from the surrounding environment. The camera also functions as the main perception device in this system, with its visual data being utilized as training data so that the dataset results can meet the values of the confusion matrix and be used as weights in the CNN method. After recognizing various gestures through visuals, Robolater is expected to operate optimally.

In detection using CNN architecture, the process is divided into three primary segments: backbone, neck, and head, which are interpreted in Figure 5. In the head segment, YOLO-v5 is used as the main detection algorithm. In the backbone, Darknet is utilized, which implements the Cross Stage Partial (CSP) strategy, similar to what is applied in YOLOv4 [27]. Furthermore, Spatial Pyramid Pooling (SPP) functions to gather information from the camera and produce outputs with consistent values. To achieve more optimal results, this study uses several backbones, including CSPDarkNet53, CSPResNeXt-50, and EfficientNet-B0. All of the backbones used share the similarity of applying CSP.

Fine-tuning weights on YOLO is a crucial step for hand gesture detection due to its ability to adapt pre-trained weights to recognize specific objects, such as hand detection and classification [28]. By fine-tuning YOLO, researchers can optimize detection accuracy and processing speed, as demonstrated by the results achieved with YOLOv7 and its variations across different datasets [29], [30]. Additionally, the combination of robotic data collection through synthetic methods to train YOLOv5x detectors highlights the importance of efficient fine-tuning techniques in achieving performance, further supporting the usefulness of fine-tuning for accurate detection [31], [32], [33], [34].

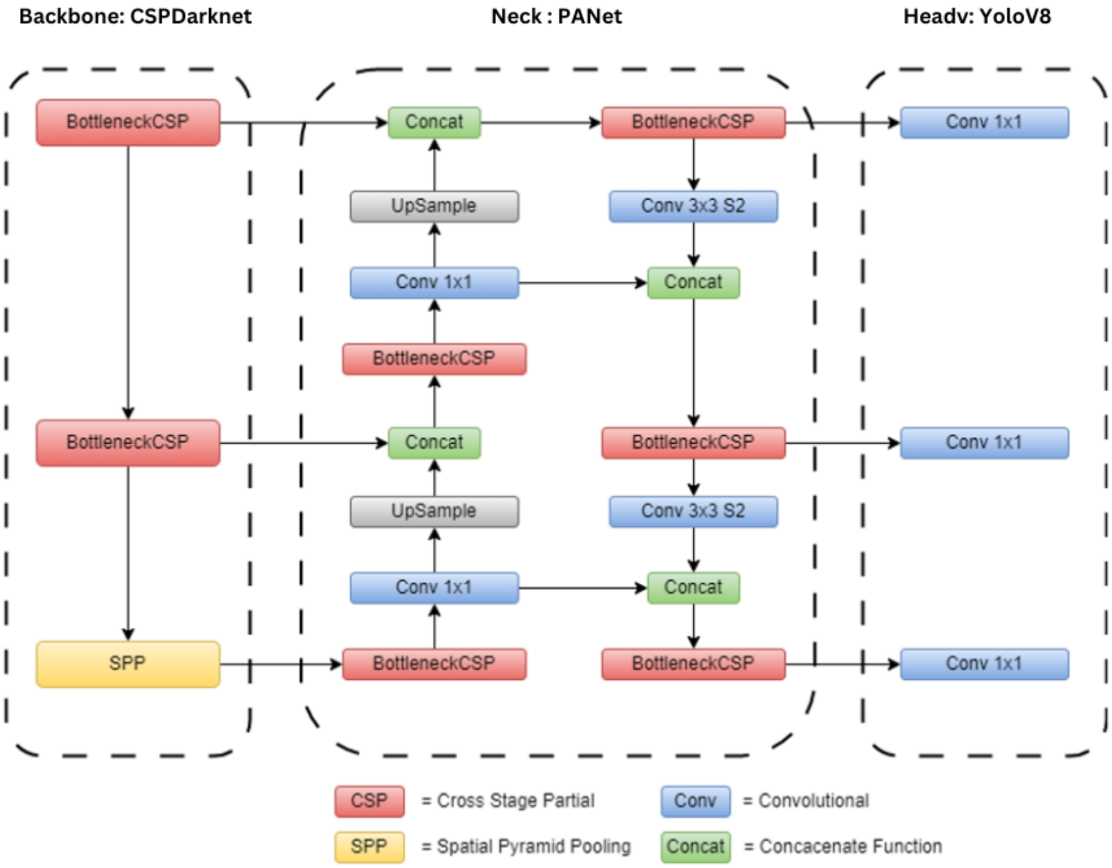

Fig. 3. CNN architecture used

A common limitation of cameras is their insensitivity to changes in light intensity, which can reduce their performance in recognizing movements [35]. As an alternative to address this shortcoming, the environment and light intensity will be set to a fixed value by adjusting the room where the robot will operate, as done in previous research [36]. By setting the light intensity to a static value, the performance of deep learning will not experience significant fluctuations in the confusion matrix values. Another obstacle is the unreliability of cameras in calculating the distance between the sensor and the object. A 2D camera cannot overcome this issue, resulting in a lack of necessary depth information [37]. To minimize this, a dataset will be collected with various gestures and objects of different sizes. A diverse and rich set of objects has an impact on the accuracy level of object detection, which can be measured through the confusion matrix [38]. The deep learning concept in this research uses CNN architecture, which is closely related to the processing of grid-like data, such as images [39]. The complexity of an image can be decomposed into data, allowing the visuals within it to be recognized. CNN is also a subfield of machine learning using supervised learning techniques. Multiclass detection requires training data rich in features. By fulfilling this aspect, the accuracy of visual detection can be reliably achieved, as seen in previous research [40].

As with any system, output is required as the final result of the deep learning process. This study proposes using speakers as output to provide information to users interacting with the Integrated Service Robot. The sound produced by the speaker consists of information regarding services available at the university, which have been arranged similarly to typical university services. This robot is expected to improve services at the university through the intelligent media embedded within it. The schematic diagram of Robolater consists of input, process, and output, which will be fundamentally illustrated in Figure 4.

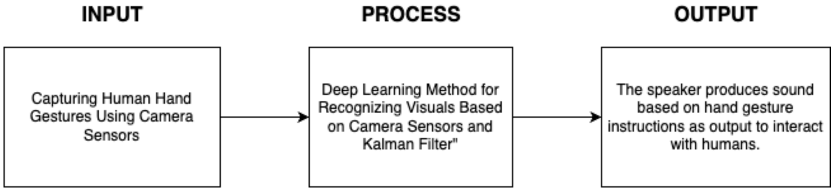
INPUT
PROCESS
OUTPUT
Capturing Human Hand Gestures Using Camera Sensors
Deep Learning Method for Recognizing Visuals Based on Camera Sensors and Kalman Filter"
The speaker produces sound based on hand gesture instructions as output to interact with humans.

Fig. 4. Integrated service robot process flowfine-tuning

The following section introduces several fine-tuning procedures along with partial F-T (Fine-Tuning) strategies to transfer a set of learned features from the base-task and apply them to the target-task, as shown in Figure 5. This process involves three main concepts: base-task, target-task, and transfer or fine-tuning. In the base-task illustrated in Figure 5, the base-dataset ($Bdataset$) is trained using the base-network ($network$B) to complete the base-task ($task$B). Meanwhile, in the target-task depicted in Figure 5, the target-dataset ($dataset$T) is trained with the target-network ($network$T) to complete the target-task ($task$T). In the fine-tuning process, the features in the form of weight parameters from ($network$B) are transferred from the pre-trained model to improve the performance of $network$T on the new task $task$T, under the assumption that $dataset$B ≠ $dataset$T or $task$B ≠ $task$T.

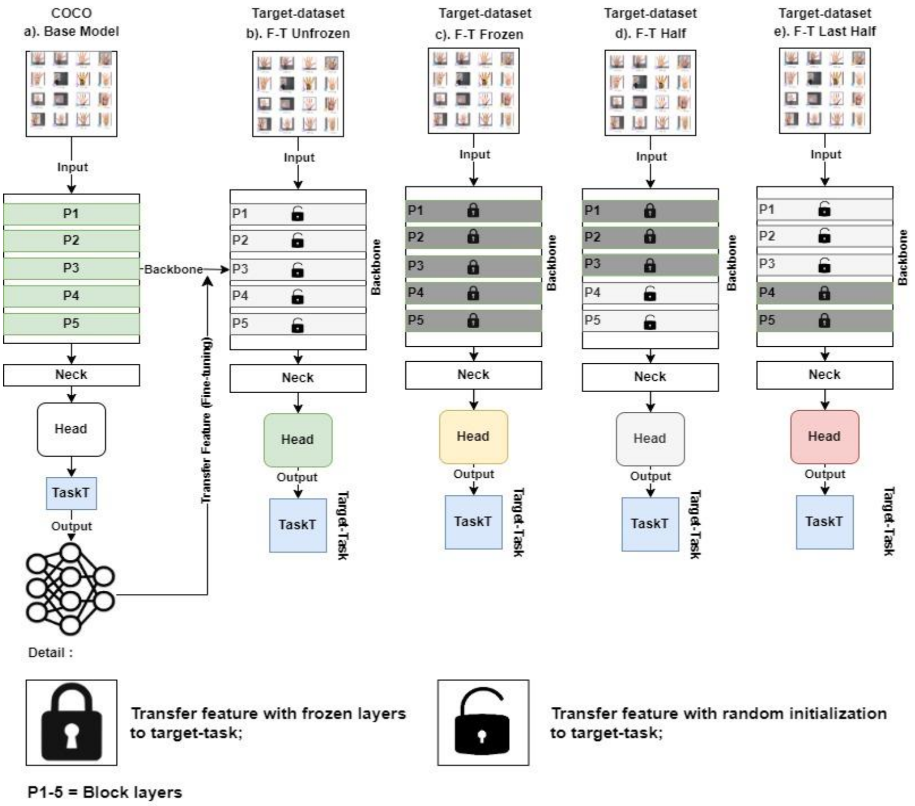

Fig. 5. Fine-tuning strategy architecture

In $network$B, we used the YOLOv8 model pre-trained on the COCO base-dataset [41] for the base-task $task$B. YOLOv8 consists of 5 separate blocks labeled $B1-5$. After that, fine-tuning was performed on $network$T by training it with the custom target-dataset to complete the target-task of hand gesture detection. The details of each fine-tuning strategy, including the spatial fine-tuning strategies (F-T Half and F-T Last Half) we proposed, are shown in Figure 5. Here's the explanation:

- Unfrozen fine-tuning (F-T Unfrozen): transfers features from the previously trained $network$B to $network$T, where the parameters in $B1-5$ are randomly initialized when trained with $dataset$T for a specific task. More detailed explanations can be found in Figure 5.
- Frozen fine-tuning (F-T Frozen): transfers features from $network$B to $network$T with $B1-5$ frozen, meaning no parameter changes occur in $network$T during training with $dataset$T. Detailed explanations are available in Figure 5.
- Half fine-tuning (F-T Half): transfers features from $network$B to $network$T with $B1-3$ frozen, while $B4-5$ are randomly initialized and updated during training. Details can be found in Figure 5.
- Last half fine-tuning (F-T Last Half): the opposite of F-T Half, where $B4-5$ are frozen, and $B1-3$ are randomly initialized, with parameter updates only occurring in $B1-3$ during training. More detailed explanations are provided in Figure 5.

3. Result and Discussion

3.1. Experimental result

A. Dataset

To validate the effectiveness of the fine-tuning strategy, we used a self-collected dataset [41]. This self-created dataset consists of six class categories: (one finger, two fingers, three fingers, four fingers, five fingers, and zero fingers) with a total of 250 images, 150 for training, 25 for validation, and 75 for testing. For all fine-tuning techniques in this research, we trained them on the self-collected dataset and evaluated them using the validation set.

B. Evaluation matrices

To evaluate each fine-tuning strategy, we used several relevant parameters [51], including precision (P), recall (R), average precision (AP), and mean average precision (mAP). The mAP measurement was conducted by setting a threshold of 0.5 for intersection over union (IoU). The details of each parameter are explained in points (1)-(4), where (1) explains P, (2) explains R, (3) explains AP, and (4) explains mAP.

$$\text{Precision (P)} = \frac{\text{TP}}{\text{TP+FP}} \tag{1}$$

$$\text{Recall (R)} = \frac{\text{TP}}{\text{TP+FN}} \tag{2}$$

$$\text{AP} = \textstyle\sum_n (R_{n+1} - R_n) \underset{\tilde{R}:\tilde{R}\geq R_{n+1}}{mamaxsax} P(\tilde{R}) \tag{3}$$

$$\text{mAP} = \frac{1}{\text{N}} \textstyle\sum_{\text{i}=1}^{\text{N}} AP_i \tag{4}$$

Based on points (1) and (2), true positive (TP) refers to correct detections based on the bounding box of the ground truth, false positive (FP) refers to detected objects that do not match, and false negative (FN) refers to instances where the ground truth bounding box is not detected. AP is the average value of P and R, as explained in (3), where $P(\tilde{R})$ represents P calculated at R. mAP is the average of AP for all class categories in the dataset, serving as a metric to assess object detection accuracy. In (4), APi refers to the AP for class i, and N is the total number of evaluated classes.

3.2.Experimental details

In this study, a pre-trained model based on YOLOv8 [41] was used as the backbone, and processing was done using a custom dataset designed to detect several gestures divided into six classes. Feature transfer from the pre-trained model to each technique, such as fine-tuning: F-T, Unfrozen F-T, Frozen F-T Half, and F-T Last Half, was performed. After determining the fine-tuning technique, each of these techniques was applied to the specified dataset [41] as the object detection task on the device. During the training phase, we used stochastic gradient descent for optimization with a momentum of 0.8, a batch size of 16, a learning rate of 0.015, and 250 training epochs with an input size of 640×640 pixels. The framework for training and validation was PyTorch using a Tesla T4 GPU, while real-time detection was carried out using hardware with the following specifications: Intel Core i5 10300H CPU @ 2.50GHz, 16 GB RAM, and NVIDIA GeForce GTX 1650.

3.3.Result

Tab. 1. Differences in fine-tuning techniques

Training	Input Size	Precision	Recall	mAP@50
F-T Traditional	640x640	67.4	65.4	65.4
F-T Unfrozen	640x641	91.8	83.6	83.2
F-T Frozen	640x642	74	75.4	72.4
F-T Half	640x643	80.4	75	73
F-T Last Half	640x644	84.6	80.6	91.6

A. Analysis of different scales

To obtain more in-depth and robust analysis results, each fine-tuning technique was evaluated with different input scales, namely 640×640, 640×641, 640×642, 640×643, and 640×644, as explained in the previous table. Unfrozen F-T showed inferior results compared to several other techniques such as F-T Traditional, F-T Frozen, F-T Half, and especially the F-T Last Half technique, with a 0.1% difference at scales 640x642 and 640×640.

However, at scales 640×644 and 640x643, the F-T Last Half technique demonstrated superiority when compared to F-T Traditional, F-T Frozen, F-T Half, and especially F-T Unfrozen, with a 0.5% difference at those two scales. These results show that F-T Unfrozen had a significantly higher value compared to F-T Last Half at smaller input scales. However, based on our experiments, the results from F-T Last Half were more stable with larger input pixels compared to F-T Unfrozen, F-T Traditional, F-T Common, and F-T Half.

Analysis F-T last half

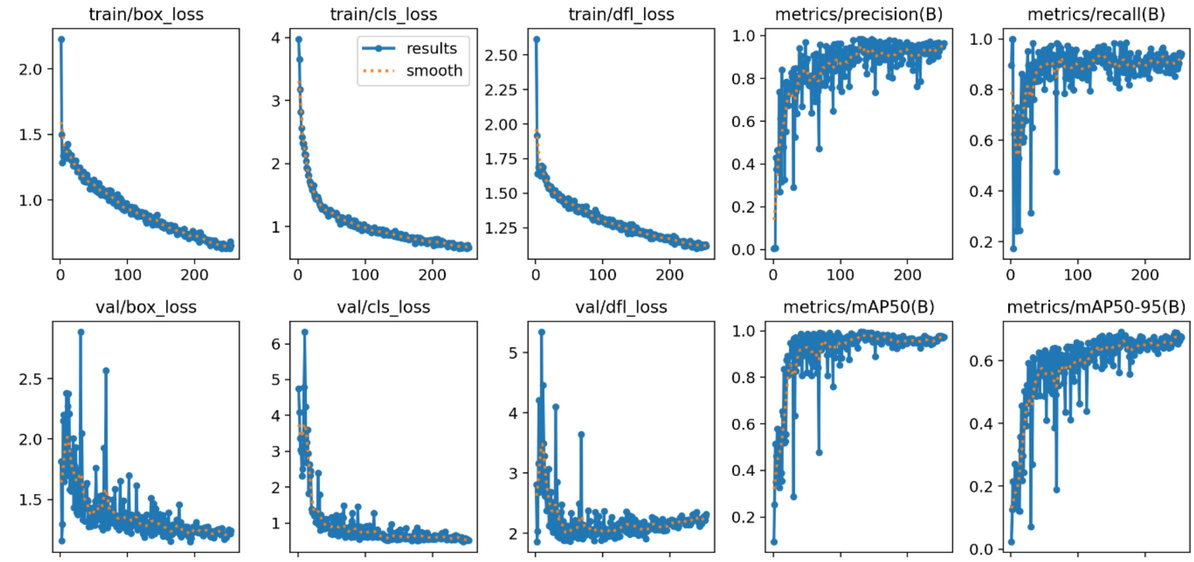

Fig. 6. Results of training on the final round of FT fine-tuning values.

In the results of the object detection model training, we observed a comparison between the validation and training values for three key metrics: box_loss, cls_loss, and dfl_loss. The box_loss on the validation data is 2.8, slightly higher than the training value of 2.3. This indicates that the model performed well in detecting bounding boxes during training but slightly declined on the validation data. This could be a sign of slight overfitting, where the model fits the training data better than the validation data.

A larger gap is seen in cls_loss and dfl_loss. The validation cls_loss is 6.3, while the training value is lower at 4. This suggests that the model had more difficulty correctly classifying objects in the validation data. A similar pattern is observed with dfl_loss, where the validation value is 5.5, significantly higher than the training value of 2.6. These two metrics indicate that the model might not generalize well enough to new data, pointing to potential overfitting.

Despite the gap in the validation metrics, the model's overall performance remains very strong. With a precision of 84.6, recall of 80.6, and mAP of 91.6, the model demonstrates strong capabilities in detecting and classifying objects accurately. The high precision means the model rarely makes false predictions on non-existent objects, and the high recall indicates that the model successfully detects most of the objects present. The mAP of 91.6 also serves as an indicator that the model consistently produces high-quality predictions.

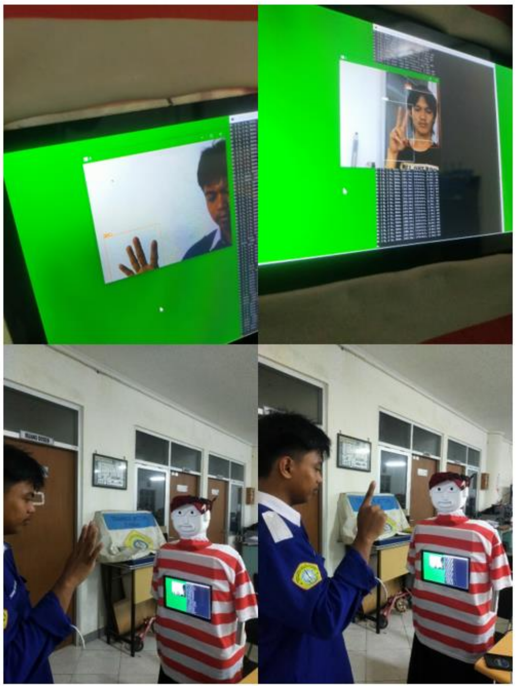

Fig. 7. Detection visualization

3.4. Comparison with the state of novelty

The discussion in this last section is to compare the proposed method, namely fine-tuning that we bring with the type of F-T Last Half along with several methods as a comparison, namely transformer-based [42] and YOLOv8-base[43]. The choice of using a comparison method in previous studies is because the technique detects the same thing and function to detect hand gestures from images, this makes the comparison possible considering that the detection of the same thing is done in our method and the previous method. The explanation in Table 2 below explains that the results of the fine-tuning carried out have a mAP value of 91.6% better than the value of the transformer-based method which has a value of 73.8% and this also has a better value than the precision value of YOLOv8-base which has a value of 78.7%. In comparison, the method proposed by the researcher has a value of 84.6%. With these results, it can be said that the fine-tuning method with the type of FT Last-Half proposed has advantages when compared to several results from previous studies [42], [43].

Tab. 2. Performance comparison with previous research

No	Model	mAP@50	Precision	Recall
1	YOLOv8-base	73.1%	78.7%	75.3%
2	Transformer-based	73.8%	86.2%	77.3%
3	Ours	91.6%	84.6%	80.6%

4. Conclusion

The main conclusion of this study is that the F-T Last Half technique demonstrates more stable performance compared to other techniques, especially with larger input scales (640×644). The model using this technique achieved an mAP of 91.6%, with a precision of 84.6% and a recall of 80.6%. Although the model experienced slight overfitting on the validation data, its overall performance remained strong in detecting and classifying objects. This experiment shows that fine-tuning with the F-T Last Half technique provides the most optimal results in object detection using the YOLOv8 model on the hand gesture dataset employed. From this analysis, it is evident that input size affects the performance of each fine-tuning technique, with F-T Last Half delivering the most optimal results overall. This technique strikes a good balance between precision, recall, and mAP, particularly at larger input scales.

5. Acknowledgments

The author would like to thank the Directorate of Research, Technology and Community Service (DRTPM), Ministry of Education, Culture, Research, and Technology for providing financial support through the Fundamental Research Schemes 2024 contract number 019/UN46.4.1/PT.01.03/2024

References

[1] D. Rustiono, “Unit Layanan Terpadu, Wajah Baru Inovasi Layanan Publik,[*Integrated Service Unit, New Face of Public Service Innovation*]” Gagasan Unnes.

[2] T. Iwan, “Peningkatan Layanan Prima Kunci Bangun Kampus Unggul dan Kompetitif, Menuju Reputasi Internasional, [*Improving Excellent Service is the Key to Building a Superior and Competitive Campus, Towards an International Reputation*]” https://uinsgd.ac.id/peningkatan-layanan-prima-kunci-bangun-kampus-unggul-dan-kompetitif-menuju-reputasi-internasional/.

[3] G. Park, V. K. Chandrasegar, and J. Koh, “Accuracy Enhancement of Hand Gesture Recognition Using CNN,” *IEEE Access*, Vol. 11, (2023).

[4] Y. L. Chung, H. Y. Chung, and W. F. Tsai, “Hand gesture recognition via image processing techniques and deep CNN,” *Journal of Intelligent and Fuzzy Systems*, Vol. 39, No. 3, (2020).

[5] A. TUTAK ERÖZEN, “A New CNN Approach for Hand Gesture Classification using sEMG Data,” *Journal of Innovative Science and Engineering (JISE)*, (2020).

[6] C.-Y. Jang and T.-Y. Kim, “Hand Feature Enhancement and User Decision Making for CNN Hand Gesture Recognition Algorithm,” *Journal of the Institute of Electronics and Information Engineers*, Vol. 57, No. 2, (2020).

[7] T. R. Gadekallu *et al.*, “Hand gesture classification using a novel CNN-crow search algorithm,” *Complex and Intelligent Systems*, Vol. 7, No. 4, (2021).

[8] Z. Wu, C. Fang, G. Wu, Z. Lin, and W. Chen, “A CNN-Regression-Based Contact Erosion Measurement Method for AC Contactors,” *IEEE Trans Instrum Meas*, Vol. 71, (2022), pp. 1-10.

[9] A. Kaluthantrige, J. Feng, J. Gíl-Fernández, and A. Pellacani, “Centroid regression using CNN-based Image Processing Algorithm with application to a binary asteroid system,” in *2022 IEEE Congress on Evolutionary Computation (CEC)*, (2022), pp. 1-7.

[10] R. Saravanan, S. Retnaswamy, and S. Selvan, “A Method of Hand Gestures Recognition using Convolutional Neural Network,” in *Lecture Notes in Electrical Engineering*, (2022).

[11] J. Qi, L. Ma, Z. Cui, and Y. Yu, “Computer vision-based hand gesture recognition for human-robot interaction: a review,” *Complex and Intelligent Systems*, (2023).

[12] M. A. A. Razak, F. Y. A. Rahman, R. Mohamad, S. Shahbuddin, Y. W. M. Yusof, and S. I. Suliman, “Hand Gesture Recognition based on Convolution Neural Network (CNN) and Support Vector Machine (SVM),” in *2023 IEEE 14th Control and System Graduate Research Colloquium, ICSGRC 2023 - Conference Proceeding*, (2023).

[13] N. Petranon and S. Umchid, “Design and Development of Hand Movement Detection Device for Sign Language using IMU and sEMG Sensors,” in *2023 Research, Invention, and Innovation Congress: Innovative Electricals and Electronics (RI2C)*, (2023), pp. 1-4.

[14] Q. Gao, J. Liu, and Z. Ju, “Hand gesture recognition using multimodal data fusion and multiscale parallel convolutional neural network for human–robot interaction,” in *Expert Systems*, (2021).

[15] M. H. Korayem, M. A. Madihi, and V. Vahidifar, “Controlling surgical robot arm using leap motion controller with Kalman filter,” *Measurement*, Vol. 178, (2021), p. 109372.

[16] M. H. Korayem and V. Vahidifar, “Detecting hand’s tremor using leap motion controller in guiding surgical robot arms and laparoscopic scissors,” *Measurement*, Vol. 204, (2022), p. 112133.

[17] Faikul Umam and Ach. Dafid, “Implementation Of Android-Based Fish Detection & Recognition System Using Convolutional Neural Network Method,” *Technium: Romanian Journal of Applied Sciences and Technology*, Vol. 16, No. 1, (2023), pp. 183-192.

[18] M. Fuad *et al.*, “Towards Controlling Mobile Robot Using Upper Human Body Gesture Based on Convolutional Neural Network,” *Journal of Robotics and Control (JRC)*, Vol. 4, No. 6, (2023), pp. 856-867.

[19] E. M. Budi, A. A. Rochim, H. K. Dipojono, A. Handojo, and J. Sarwono, “Musical gesture recognition for interactive angklung robot,” in *2013 3rd International Conference on Instrumentation Control and Automation (ICA)*, (2013), pp. 149-154.

[20] G. Park, V. K. Chandrasegar, and J. Koh, “Accuracy Enhancement of Hand Gesture Recognition Using CNN,” *IEEE Access*, Vol. 11, (2023), pp. 26496-26501.

[21] G. Park, V. K. Chandrasegar, and J. Koh, “Accuracy Enhancement of Hand Gesture Recognition Using CNN,” *IEEE Access*, Vol. 11, (2023).

[22] H. Y. Chung, Y. L. Chung, and W. F. Tsai, “An efficient hand gesture recognition system based on deep CNN,” in *Proceedings of the IEEE International Conference on Industrial Technology*, (2019).

[23] 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, (2022).

[24] M. H. Korayem, R. Vosoughi, and V. Vahidifar, “Design, manufacture, and control of a laparoscopic robot via Leap Motion sensors,” *Measurement*, Vol. 205, (2022), p. 112186.

[25] A. Najafinejad and M. H. Korayem, “Detection and minimizing the error caused by hand tremors using a leap motion sensor in operating a surgeon robot,” *Measurement*, Vol. 221, (2023), p. 113544.

[26] Z. Tan and M. Karaköse, “Optimized Reward Function Based Deep Reinforcement Learning Approach for Object Detection Applications,” in *2022 International Conference on Decision Aid Sciences and Applications (DASA)*, (2022), pp. 1367-1370.

[27] M. Mahasin and I. A. Dewi, “Comparison of CSPDarkNet53, CSPResNeXt-50, and EfficientNet-B0 Backbones on YOLO V4 as Object Detector,” *International Journal of Engineering, Science and Information Technology*, (2022), pp. 64-72.

[28] T.-H. Nguyen, R. Scherer, and V.-H. Le, “YOLO Series for Human Hand Action Detection and Classification from Egocentric Videos,” *Sensors*, Vol. 23, (2023), p. 3255.

[29] H. Chen, W. Wan, M. Matsushita, T. Kotaka, and K. Harada, “Automatically Prepare Training Data for YOLO Using Robotic In-Hand Observation and Synthesis,” *IEEE Transactions on Automation Science and Engineering*, (2023), pp. 1-17.

[30] 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), (2023), pp. 1-5.

[31] Y. Dai and P. Chen, “YOLO lightweight contraband detection network using attention mechanism,” in *International Conference on Mechatronics Engineering and Artificial Intelligence (MEAI 2022)*, C. Zhao, Ed., SPIE, (2023).

[32] 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)*, (2023), pp. 1-7.

[33] C. Bao, T. Xie, W. Feng, L. Chang, and C. Yu, “A Power-Efficient Optimizing Framework FPGA Accelerator Based on Winograd for YOLO,” *IEEE Access*, Vol. 8, (2020), pp. 94307–94317.

[34] W. Fang, L. Wang, and P. Ren, “Tinier-YOLO: A Real-Time Object Detection Method for Constrained Environments,” *IEEE Access*, Vol. 8, (2020), pp. 1935-1944.

[35] Y. Zhang, “A Rapid Automatic Exposure Technology for Micro-Nano Remote Sensing Camera,” in *2021 4th International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE)*, (2021), pp. 744-748.

[36] M. Wischow, G. Gallego, I. Ernst, and A. Börner, “Monitoring and Adapting the Physical State of a Camera for Autonomous Vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, Vol. 25, No. 5, (2024), pp. 4290-4303.

[37] J. Kolowski, J. Oley, and W. McShea, “High-density camera trap grid reveals lack of consistency in detection and capture rates across space and time,” *Ecosphere*, Vol. 12, Jun. (2021).

[38] L.-E. Pommé, R. Bourqui, R. Giot, and D. Auber, “Relative Confusion Matrix: Efficient Comparison of Decision Models,” in *2022 26th International Conference Information Visualisation (IV)*, (2022), pp. 98-103.

[39] G. Calderon, A. Perez, M. Nakano, K. Toscano, H. Quiroz, and H. Perez, “CNN-Based Quality Assessment for Retinal Image Captured by Wide Field of View Non-Mydriatic Fundus Camera,” in *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, (2019), pp. 282-285.

[40] P. A. Dkengne Sielenou and S. Girard, “Detection of Traffic Scene Objects using YOLO Algorithm: Theory and Practical Guide,” Jun. (2023).

[41] J. H. Lee, G. Mikriukov, G. Schwalbe, S. Wermter, and D. Wolter, “Concept-Based Explanations in Computer Vision: Where Are We and Where Could We Go?,” (2024).

[42] G. Pei and C. Xu, “Hand Detection and Gesture Classification based on Transformer,” in *2022 China Automation Congress (CAC)*, (2022), pp. 276-281.

[43] S. R. M, S. M. M. Roomi, B. Sathyabama, and M. Senthilarasi, “Hand Gesture Recognition System Using Transfer Learning,” in *2023 International Conference on Energy, Materials and Communication Engineering (ICEMCE)*, (2023), pp. 1-5.

Follow this article at the following site:

Faikul Umam, Hanifudin Sukri, Ach Dafid, Firman Maolana & Mycel Natalis Stopper Ndruru. Optimization of Hand Gesture Object Detection Using Fine-Tuning Techniques on an Integrated Service of Smart Robot. IJIEPR 2024; 35 (4) :1-11
URL: http://ijiepr.iust.ac.ir/article-1-2141-en.html