Empowering Elderly Care: Innovative Fall Detection with OpenPose and YOLO

Ridho Adha Hardiyanto[1], Casi Setianingsih[2], Tito Waluyo Purboyo[3], Aura Syafa Aprillia Radim[4], Alvin Anandra Brilliandy[5], M. 'Irfan Chanif Rusydi[6], Muhammad Rakan Fawwaz[7], Nurul Amelia[8]

School of Electrical Engineering, Telkom University, Bandung, Indonesia

[1]ridhoadha117@gmail.com, [2]setiacasie@telkomuniversity.ac.id, [3]titowaluyo@telkomuniversity.ac.id, [4]syafaaura@student.telkomuniversity.ac.id, [5]alvinanandra@student.telkomuniversity.ac.id, [6]chanifrusydi@student.telkomuniversity.ac.id, [7]mrakanf@student.telkomuniversity.ac.id, [8]nramelia@student.telkomuniversity.ac.id

2023 6th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI) | 979-8-3503-5834-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISRITI60336.2023.10468014

***Abstract*—**In the elderly population, the physiological and psychological functions undergo a decline, rendering them susceptible to various risks. Families seek solutions to assist elderly individuals in times of distress. A devised system employs a webcam and utilizes the You Only Look Once (YOLO) algorithm to detect elderly movements and positions within the household. YOLO employs a single neural network, segmenting the image into regions and predicting bounding boxes and probabilities to classify objects. The highest probability bounding box becomes the object separator. The camera is positioned in a room, capturing a 3m x 3m area, and inference is made. The project compares performance between the OpenPose and non-OpenPose systems. The optimal model under OpenPose demonstrates exceptional results, boasting 100% precision, recall, F1 score, and mAP, with an accuracy of 100%. This success is achieved with a ratio of 70%:30%, a batch size of 64, and a learning rate of 0.01.

Keywords—*elderly, fall detection, yolo, object detection.*

I. INTRODUCTION

When humans enter old age, or can be referred to as elderly people (elderly), various body systems in the elderly experience a decline, both in terms of psychology and physically, which makes them not work optimally[1]. This condition carries serious risks, especially when the elderly experience falls, which can cause damage to the elderly body or trigger the development of other diseases due to age[2]. The consequences of these events can be fatal or bring unwanted impacts. The decline in the body functions of the elderly which is very worrying requires a solution to help the elderly family to supervise or take action if something happens to the elderly. In facing this challenge, a holistic and integrated approach needs to be applied, which not only pays attention to the psychological and social aspects of elderly well-being but also the physical aspects.

A system that works in real-time and intelligently becomes one of the solutions[3] where one of the systems, namely artificial intelligence, is the answer to the problem called. Realtime systems are also needed so that the elderly can be monitored as well as being able to see what is happening to the elderly[4]. One of the artificial intelligence that will be used in this final project is OpenPose as a working device that will perform object detection and Artificial Neural Network, and the variation of the Artifical Neural Network that will be used is YOLO (You Only Look Once) to detect whether the elderly fall or not. The YOLO method is one of the fastest and most accurate object detection algorithms, which can detect objects up to twice as fast as other algorithms[16].

II. RELATED WORK

There are several studies that discuss a fall detection framework using OpenPose and artificial neural networks (RNN) to detect falls based on skeletal data. The framework achieves a high fall detection accuracy of 98.2% and eliminates the need for wearable sensors. The authors discuss the dataset, data pre-processing techniques, and the use of linear interpolation and different models (RNN, LSTM, GRU) to analyze joint point data. The results show that the relative position normalization method has the best performance in terms of accuracy[5].

In the research conducted, it was concluded that the proposed fall detection algorithm, which integrates feature enhancement using OpenPose and fall detection using MobileNetV2, achieved higher accuracy in detecting falls compared to other existing methods. The use of OpenPose helps to improve the accuracy of human key point labeling, especially in low-light environments, by highlighting dark frames. The algorithm accurately detected the fall behavior in every frame of the test dataset and achieved 99.75% accuracy, surpassing the accuracy of other listed algorithms. The proposed method combines human key point information and human pose information to enhance features without increasing image complexity, resulting in better detection accuracy[6].

A novel semi-supervised framework for image categorization purposes utilizing Deep Neural Networks was developed by Mohammadmin Salimi, Jose J.M. Machado and Joao Manuel R.S. Taveres (2022). The proposed fall detection solution based on the Fast-Pose estimation method achieves high accuracy and low computational requirements, making it suitable for use on edge devices. This model outperforms existing state-of-the-art methods for fall detection. This study highlights the need for more diverse datasets and exploration of augmentation techniques, such as Generative Adversarial Networks (GAN), for future research. Human fall detection in images is complex due to the similarity of fall poses to other activities such as sitting or lying down, but fall detection has distinguishable characteristics, such as rapid changes in body dimensions and inactivity on the ground. The development of lightweight and optimized solutions for pose estimation is important in this field[7].

III. RESEARCH METHOD

The objective of this study is to find out if the person (LANSIA) falls or not and compare the best performance between the system falls using OpenPose or not using OpenPose.

A. Artificial Neural Network

Artificial Neural Network is a machine learning technique that mimics a human neural network system. The process of creating this network involves connected points that form layers, such as input, output, and hidden node layers. These nodes are activated using activation functions such as sigmoid, tanh, or ReLU[3]. Examples of artificial neural network implementations include sparse encoders, convolutional neural networks (CNNs), restricted Boltzmann machines (RBMs), and Long short-term memory (LSTM).

B. You Only Look Once (YOLO)

You Only Look Once (YOLO) is an algorithm that uses DarkNet's Convolutional Neural Network (CNN)[8] as its base. In YOLO CNN, there are several evaluation matrix performance parameters that are used when the model performs detection:

1. IoU (Intersection over Union)

This metric measures the extent of overlap between predicted and actual bounding boxes[9]. IoU is used to evaluate the accuracy of object detection.

2. Precision

Precision measures the ability of a model to identify only relevant objects. Precision indicates how many objects are predicted to be correct out of the total predicted objects[9]

3. Recall

Recall measures the model's ability to detect all actual objects. It calculates how many objects were successfully detected out of the total objects present[9]

4. Average Precision (AP)

AP measures the quality of the model in representing the area under the Precision-Recall curve. The larger the area under the curve, the better the model will perform[9]

5. Mean Average Precision (mAP)

mAP is the average of the average precision (AP) values for all classes of objects. This gives an overall picture of the model's performance[9]

6. Mean Average Precision .95 (mAP@.95)

This metric extends AP@.5 and AP@.75 by calculating AP@ with a range of different IoU values (e.g., 0.5, 0.55, ..., 0.95) and averaging the results. These metrics provide more detailed information about model performance at different levels of overlap[9]

The higher the IoU, precision, recall, AP, mAP, and mAP@.95 values, the better the performance of CNN's YOLO detection model.

C. Pose Estimation

Pose estimation is a Machine Learning technique that identifies the position and relationship of joint points and parts of the human body[10]. It uses Deep Learning to classify body configurations, allowing computers to recognize body movements such as hands, head, and feet[11]. Examples of Pose estimation forms include OpenPose, AlphaPose, TF Pose estimation, openPifPaf, and others. This technique allows computers to recognize and understand the movements of the human body.

Fig 1. Pose Estimation

D. Openpose

OpenPose is a real-time detection system capable of identifying more than 135 points of the human body, including hands, face, and feet, in a single image[12]. The system was developed by researchers at Carnegie Mellon University in 2017. OpenPose has various implementations such as Python code, C++ implementation, and Unity Plugin. It allows detection and mapping of points of the human body in real time.

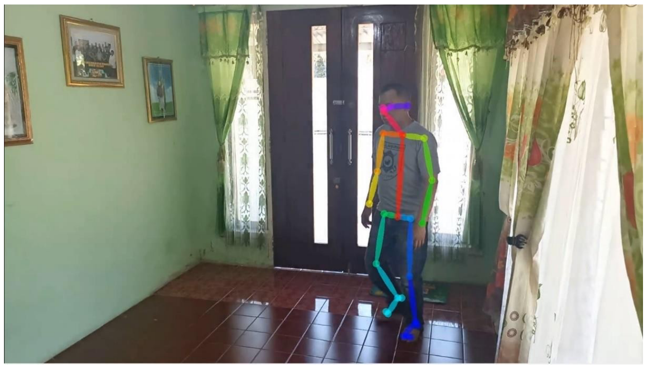

Fig 2. OpenPose

E. Inference in Machine Learning

Inference in the context of machine learning is the step of generating predictions as well as outputting prediction results towards the actual outcome direction of the data fed into the machine learning model, and also performing calculations against a single numerical score. This process is also known as "machine learning model building"[13]. On the other hand, according to[14], inference in machine learning refers to evaluating the extent to which current inputs match each rule and decision making according to those inputs. Next, rules of lesser relevance are combined to form control actions in machine learning. View [15], defines inference as a system component that applies logical rules to a knowledge base to generate new information. The first component of the inference engine is the expert system. The standard expert system consists of a knowledge base as well as an inference engine. The knowledge base stores information about the world. The inference engine applies logical rules to the knowledge base and draws new conclusions. This step is repeated because any additional information in the knowledge base might result in the application of additional rules on the inference engine. The inference engine operates primarily in two modes: forward chaining and backward chaining. Forward chaining starts with known facts and generates new facts. Backward chaining starts from the goal, and goes backwards to determine what facts need to be confirmed in order for the goal to be achieved.

IV. SYSTEM DESIGN & OVERVIEW

A. System Overview

In this research You Only Look Once (YOLO) and OpenPose will be used to detect people who have reached the age of elderly and fallen. The data will be in the form of video sourced from a webcam laptop and will be processed by YOLO and OpenPose to determine the elderly who have fallen. The system will run as follows:

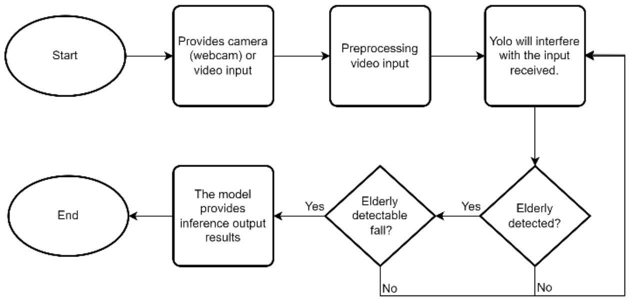

Fig 3. Flowchart System

B. Elderly Fall Data

Data is collected manually by taking direct photos of 430 images. the data that has been collected is divided into 190 images without OpenPose and 240 images with OpenPose. image data is divided into images of the elderly falling, the elderly not falling, not the elderly falling, and not the elderly not falling.

Fig. 4. (a) Datasets without OpenPose to Process

Fig. 4. (b) Datasets with OpenPose to Process

C. Yolo Algorithm Process

The first process in the YOLO algorithm is to input a dataset in the form of an image or video of the desired object such as the elderly and unknown for training. After that, the dataset is resized to 416x416 pixels and then performs the two most important processes in this algorithm, namely Convolution and Max Pooling. Then the results of the two processes produce bounding boxes and each bounding box has a confidence score to predict the highest value of the desired class classification. Finally, non-max suppression will eliminate bounding boxes with low confidence scores.

D. OpenPose process

The first process that runing out in OpenPose is that OpenPose will accept the input of an image, then the image will pass through the CNN architecture here using CNN VGG-19, then the result of feature extraction will go through the next 2 stages, the first is OpenPose will provide map confidence predictions, namely point -the point of interest of each pixel in the image that is believed to be a part of the human body, and after that, along with the map confidence prediction, a part affinity field prediction process occurs, where in the location of the human body in the image is determined. After completion, the next step is to combine the results of the 2 predictions with the bipartite machine learning algorithm, then OpenPose will give the output rendered.

E. Image Annotation

Labeling or annotation is done on the image to provide a bounding box or boundary box along with the class name on the image object to be trained. This image annotation is done using the Roboflow web. To determine the elderly and detected falls, the dataset that has been collected is used. After that upload the collected images to the Roboflow website, after that provide a boundary box and label according to the object.

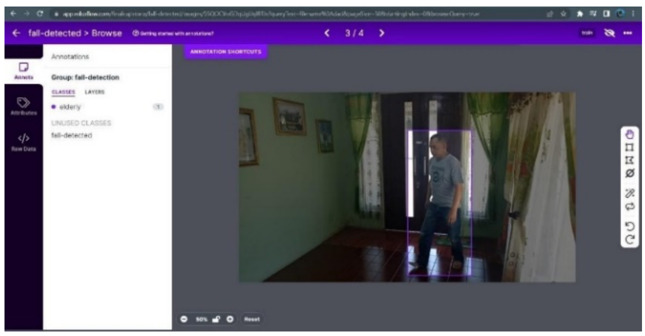

Fig 5. Image Annotation

After performing the labeling process on the dataset, the result of the labeling is a coordinate point stored in the form of .txt with a value of 0 in the form of elderly and 1 in the form of fall detected. The following is a view of the coordinates after labeling the dataset.

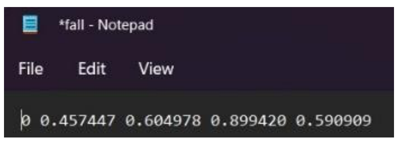

Fig 6. Image Annotation Result

V. THE RESULT

A. Testing

The testing scenario planned for this culminating project involves conducting training tests using datasets that have been manually labeled. The tests will focus on evaluating different parameters, including dataset ratios, learning rates, and batch sizes. Concerning ratio testing, the dataset will be split into three distinct ratios: 70%:30%, 80%:20%, and 90%:10%. For testing the learning rate is divided into 0.01, 0.03, and 0.05. For testing batch size is divided into 16, 32, and 64. The training process was carried out using Google Colab Notebook.

1. Ratio Test Results

Following the segmentation of the dataset into three proportions, which are 70% allocated for training and 30% for testing, 80% for training and 20% for testing, as well as 90% for training and 10% for testing, it becomes evident that the ratio with 90% training data and 10% testing data results in the highest performance of 99%.

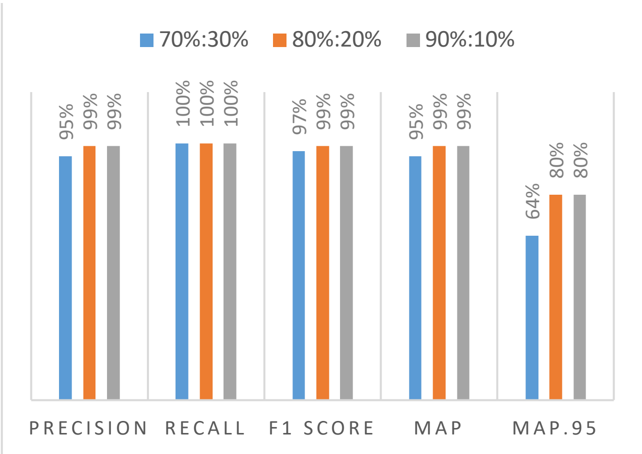

Fig 7. Ratio Test without OpenPose

The results of the training performed on all ratio datasets with OpenPose achieved a mAP of 99%. However, at 90%:10% ratio the model shows a relatively stable trend model performance in training.

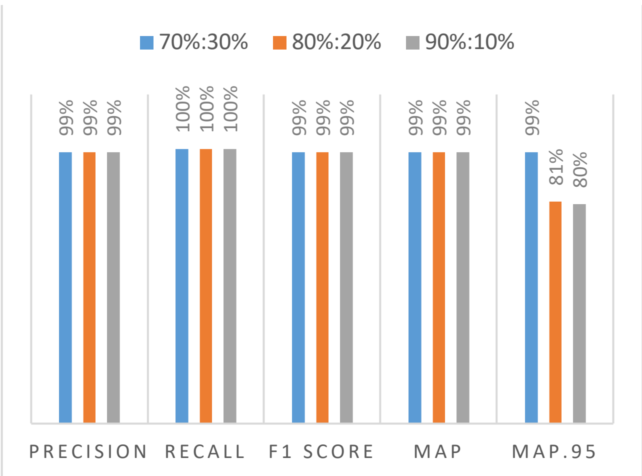

Fig 8. Ratio Test with OpenPose

Fig 9 is a graph of training results with a 90%:10% dataset ratio.

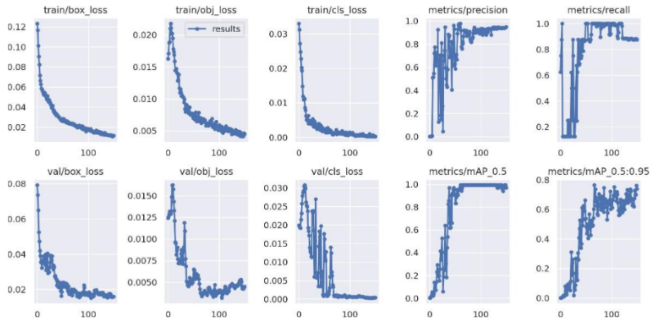

Fig 9. Training results with a 90%:10% dataset ratio

2. Learning Rate Test Results

In the experiments conducted on the learning rate, the dataset ratio used is 70%:30% ratio where the learning rate varies (0.01, 0.03, and 0.05). In experiments without using OpenPose, learning rate 0.01 gets the highest mAP results reaching 99%.

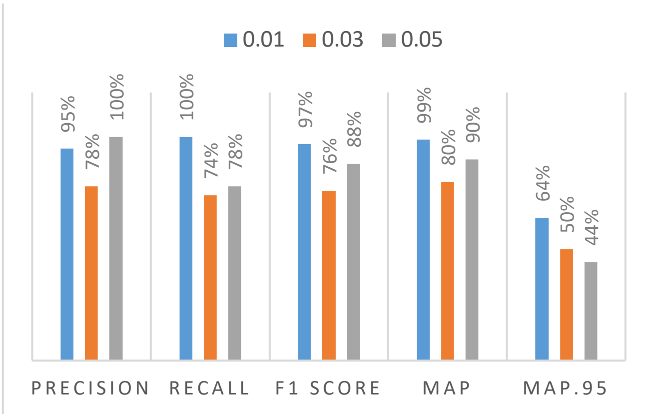

Fig 10. Learning Rate Test without OpenPose

Fig 11 this is the result using OpenPose.

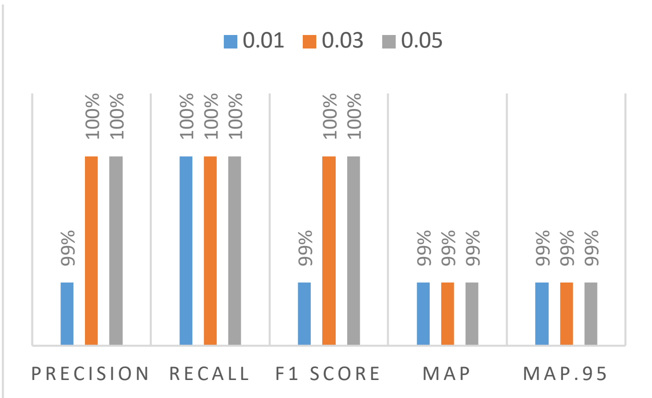

Fig 11. Learning Rate Test with OpenPose

Training with a learning rate of 0.05 with a ratio of 70% train data and 30% test data produces a stable recall performance and is the same as learning rate 0.03.

3. Batch Testing

In the training stage using various batch size (16, 32, and 64), the highest achievement reached 100%. This result was observed on models with batch sizes of 32 and 64 using OpenPose. This success may be due to the performance improvement that tends to occur when the batch size gets larger. In this case, increasing the batch size tends to contribute to the improvement of the model performance. In this scenario, the dataset used is with a ratio of 70%:30%, a learning rate of 0.01, and 150 epochs.

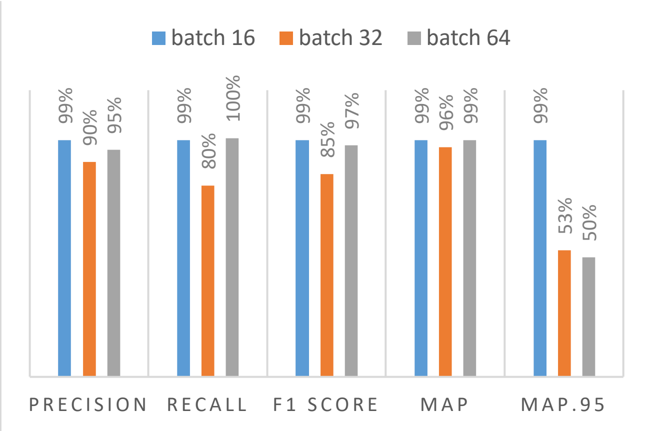

Fig 12. Batch Size Test without OpenPose

Fig 13 this is the result using OpenPose.

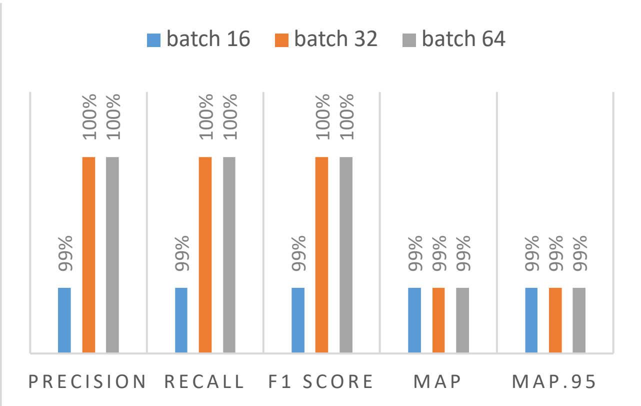

Fig 13. Batch Size Test without OpenPose

B. System Scenario

The fall detected System will use a program that is connected to a camera in 1 room in the area for the experiment with width 3x3 meters and a input tools will given to system will placed in corner of the room.

C. Detection Direction Testing

The test regarding the detection direction run in this research is expected to result in the identification of a successfully detected fall event, with the following results: "fall-detected". This fall detection system can detect the object from several angles. This is useful for improving system detection in detecting the elderly. It can be seen from the following figure.

Fig 14. Examples of fall Detection Tests from Several Angles with OpenPose

Fig 15. Examples of fall Detection Tests from Several Angles

VI. CONCLUSION

Based on the results of testing and analysis that has been carried out in this research, it can be concluded that the elderly fall detection system has succeeded in 100% detecting the elderly and determining falls or not using the YOLO (You Only Look Once) algorithm. By conducting hyperparameter testing on the yolo algorithm, the best model performance was obtained at a ratio of 70%: 30%, Batch size 64, Learning rate 0.01 with results of 100% Precision, 100% Recall, 100% F1 Score, 100% mAP and the resulting accuracy reached 100%. During testing the shape and direction of the object detected must be correct because if you make an error from the video / camera recorder, the model will cause incorrect detection. The suggestion that can be proposed for future research is to use Google Colab Pro + to conduct training so that the length of training time becomes faster and not subject to limits. Using high-spec hardware such as NVIDIA GPUs with multiple cores can use a better version of YOLO so that the detection process is better and get stable fps and add more diverse datasets.

REFERENCES

[1] L. Mayhew, "Health and Elderly Care Expenditure in an Aging World," Laxenburg, Sep. 2000.

[2] D. Park, H. Jo, C. H. Yoon, E. S. Lee, M.-K. Oh, and C. H. Lee, "Fall Risk Assessment of Rural Elderly Population in Korea," *Ann Rehabil Med*, vol. 43, no. 3, pp. 269–278, Jun. 2019, doi: 10.5535/arm.2019.43.3.269.

[3] A. Shrestha and A. Mahmood, "Review of Deep Learning Algorithms and Architectures," *IEEE Access*, vol. 7, pp. 53040–53065, 2019, doi: 10.1109/ACCESS.2019.2912200.

[4] Z. He, D. Lu, Y. Yang, and M. Gao, "An Elderly Care System Based on Multiple Information Fusion," *J Healthc Eng*, vol. 2018, pp. 1–13, 2018, doi: 10.1155/2018/4098237.

[5] C.-B. Lin, Z. Dong, W.-K. Kuan, and Y.-F. Huang, "A Framework for Fall Detection Based on OpenPose Skeleton and LSTM/GRU Models," *Applied Sciences*, vol. 11, no. 1, p. 329, Dec. 2020, doi: 10.3390/app11010329.

[6] M. Gao, J. Li, D. Zhou, Y. Zhi, M. Zhang, and B. Li, "Fall detection based on OpenPose and MobileNetV2 network," *IET Image Process*, vol. 17, no. 3, pp. 722–732, Feb. 2023, doi: 10.1049/ipr2.12667.

[7] M. Salimi, J. J. M. Machado, and J. M. R. S. Tavares, "Using Deep Neural Networks for Human Fall Detection Based on Pose Estimation," *Sensors*, vol. 22, no. 12, p. 4544, Jun. 2022, doi: 10.3390/s22124544.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," Jun. 2015.

[9] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, IEEE, Jul. 2020, pp. 237–242. doi: 10.1109/IWSSIP48289.2020.9145130.

[10] M. Hofmann and D. M. Gavrila, "Multi-view 3D Human Pose Estimation in Complex Environment," *Int J Comput Vis*, vol. 96, no. 1, pp. 103–124, Jan. 2012, doi: 10.1007/s11263-011-0451-1.

[11] A. Toshev and C. Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2014, pp. 1653–1660. doi: 10.1109/CVPR.2014.214.

[12] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional Multi-person Pose Estimation," Nov. 2016.

[13] M. H. Kazeminezhad, A. Etemad-Shahidi, and S. J. Mousavi, "Application of fuzzy inference system in the prediction of wave parameters," *Ocean Engineering*, vol. 32, no. 14–15, pp. 1709–1725, Oct. 2005, doi: 10.1016/j.oceaneng.2005.02.001.

[14] I. H. Sarker, "AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems," *SN Comput Sci*, vol. 3, no. 2, p. 158, Mar. 2022, doi: 10.1007/s42979-022-01043-x.

[15] J. Self, "Building expert systems, edited by F. Hayes-Roth D.A. Waterman and D.B. Lenat Addison-Wesley, London, 1983 (£27.50)," *Robotica*, vol. 2, no. 2, pp. 119–119, Apr. 1984, doi: 10.1017/S0263574700002022.

[16] A. Amwin , *DETEKSI DAN KLASIFIKASI KENDARAAN BERBASIS ALGORITMA YOU ONLY LOOK ONCE (YOLO)* . 2021.