An Efficient Fall Detector Using Improvement of the YOLOv12n Via ONA-Net

Florensce Sumarauw
Master Program of Informatics
Postgraduate Program
Sam Ratulangi University
Jl. Kampus Unsrat, Bahu, Kecamatan Malalayang, Kota Manado, Sulawesi Utara, 95115, Indonesia
florenscesumarauw026@student.unsrat.ac.id

Febriyanti Ludja
Master Program of Informatics
Postgraduate Program
Sam Ratulangi University
Jl. Kampus Unsrat, Bahu, Kecamatan Malalayang, Kota Manado, Sulawesi Utara, 95115, Indonesia
febriyantiludja026@student.unsrat.ac.id

Robby Moody Lintong
Master Program of Informatics
Postgraduate Program
Sam Ratulangi University
Jl. Kampus Unsrat, Bahu, Kecamatan Malalayang, Kota Manado, Sulawesi Utara, 95115, Indonesia
robbymoodylintong026@student.unsrat.ac.id

Alwin Melkie Sambul
Master Program of Informatics
Postgraduate Program
Sam Ratulangi University
Jl. Kampus Unsrat, Bahu, Kecamatan Malalayang, Kota Manado, Sulawesi Utara, 95115, Indonesia,
asambul@unsrat.ac.id

Muhamad Dwisnanto Putro
Master Program of Informatics
Postgraduate Program
Sam Ratulangi University
Jl. Kampus Unsrat, Bahu, Kecamatan Malalayang, Kota Manado, Sulawesi Utara, 95115, Indonesia
dwisnantoputro@unsrat.ac.id

Article history: Received April 4, 2026| Revised April 21, 2026| Accepted April 28, 2026

***Abstract* – The increasing demand for reliable fall detection systems in patient care monitoring while assisting livelihood environments drives the development of computer vision models capable of accurately identifying fall events in real-world conditions. These systems must detect human posture changes across various positions while maintaining robustness against complex backgrounds, which often reduce detection accuracy. Additionally, real-world deployment requires models that operate efficiently on low-cost devices and process live video streams in real time. This study analyses the effectiveness of improving the YOLOv12n version architecture for fall detection by integrating an ONA-Net mechanism. The proposed module enables the network to focus on multiple important responses related to human body posture, allowing the model to better capture spatial cues associated with fall events. The lightweight design reduces computational overhead while maintaining effective feature extraction for accurate detection. Experimental findings exhibit YOLOv12n-ONA-Net as the introduced model obtains strong detection performance, obtaining 92.3% mAP@50 and 59.7% mAP@50:95. Despite its lightweight architecture, the model maintains practical efficiency achieving an inference speed of 13.13 frames per second (FPS). However, the evaluation is primarily limited to CPU-based environments, which may not fully reflect performance across a broader range of hardware architectures. These results demonstrate that integrating ONA-Net into YOLOv12n enhances fall detection accuracy while maintaining the necessary efficiency for real-time deployment on standard CPU devices.**

Keywords: Fall Detection, YOLOv12n, ONA-Net, Real-Time Object Detection, Deep Learning

I. INTRODUCTION

Fall detection become critical component in healthcare monitoring systems that aim to improve safety and living conditions, particularly for geriatric individuals and patients with mobility limitations [1]. Fall is one of the causes of lesion and hospitalization for elderly, making early and reliable detection essential for timely medical response [2]. Vision-based fall detection has emerged as an effective approach because it allows continuous monitoring without requiring users to wear additional sensors or devices [3]. By analyzing visual information captured from cameras, computer vision systems can recognize human posture and movement patterns associated with fall events [4]. However, detecting it remains challenging due to variations in human poses, occlusions, and complex backgrounds [5]. Conventional approaches have difficulties to capture complex spatial patterns and produce limited accuracy in challenging environments [6],[7],[8]. Feature extraction plays a fundamental role in computer vision systems because it determines how effectively the model can discern among common situations and fall incidents [9].

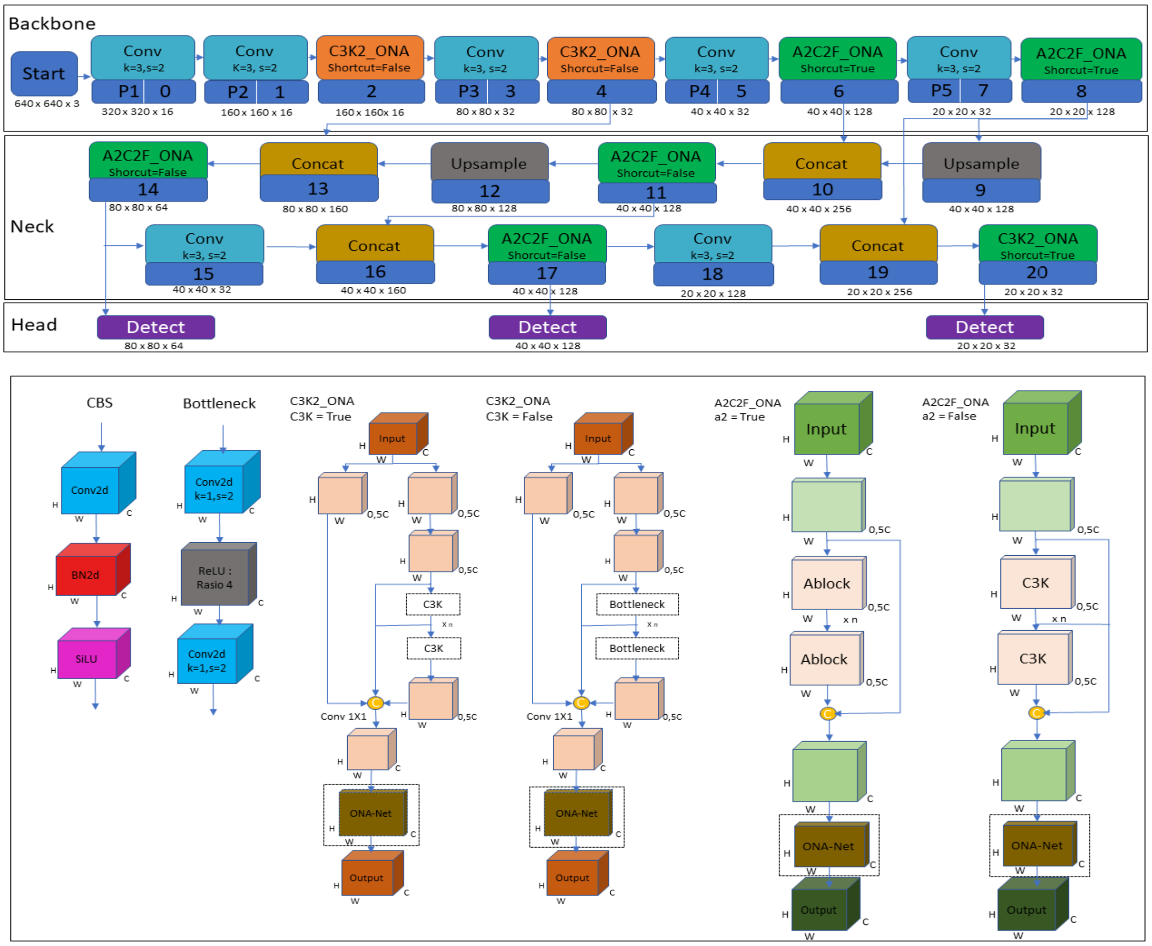

Figure 1. YOLOv12n-ONA-Net, consist of the backbone that have A2C2f and C3K2 both as for attribute extractor. The Neck for fusion of feature representations, and the head for making prediction on three network layers (P3, P4, P5). Optimized Nonlinear Attention (ONA) incorporated as enhancer for feature extraction.

In contemporary times, deep learning has significantly improved object detection and human activity recognition by enabling automatic extraction of discriminative visual features [10], [11]. Among these methods, YOLO is widely recognized for its real-time detection capability with competitive accuracy [12]. Lightweight variants such as YOLOv12n aim to balance performance and computational efficiency [13], however, they often struggle to capture subtle spatial patterns in fall events, especially under body overlap or unusual postures [14]. Existing approaches, such as BMR_YOLO, address complex environments but remain limited in modeling fine-grained spatial dependencies [15]. Compared with attention mechanisms such as SE and CBAM, which emphasize channel or combined channel–spatial attention with additional overhead, ONA more efficiently captures multi-perspective spatial information while remaining lightweight and suitable for real-time CPU deployment [16],[17]. Specifically, ONA leverages multi-frequency feature extraction by combining average pooling and max pooling, followed by a 1 × 1 convolution to enhance object pattern learning.

This highlights the need for a compact attention mechanism that enhances spatial features with minimal cost, while reduced channels enable a lightweight, CPU-efficient model. [18]. This study integrates an Optimized Nonlinear Attention (ONA) module into YOLOv12 to better capture posture-related spatial features while preserving real-time performance, improving fall detection accuracy. The key contributions are as follows:

1) An enhanced lightweight fall detection model based on YOLOv12n is proposed, improving detection performance while minimizing computational cost and model complexity.
2) An Optimized Nonlinear Attention (ONA) module integrates bottleneck convolution and residual learning to enhance efficiency and feature representation, and is incorporated into YOLOv12n to improve feature-level encoding.
3) Extensive experiments show that YOLOv12n-ONA achieves 92.3% mAP@50 and 59.7% mAP@50:95 at 13.13 FPS, demonstrating competitive performance for fall detection.

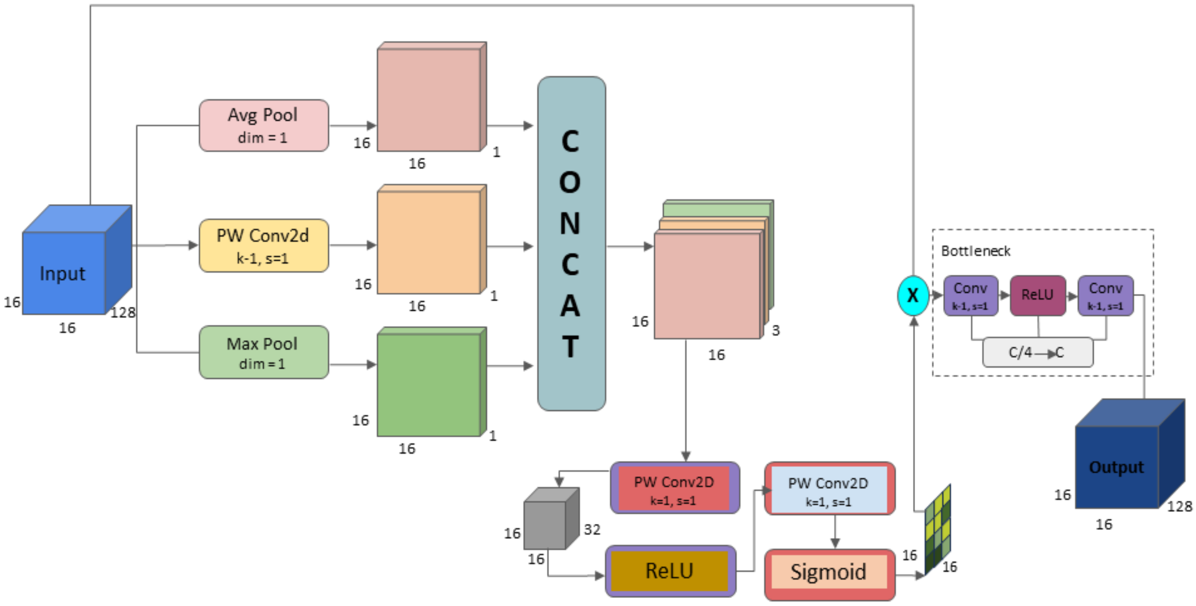

Figure 2 Optimized Nonlinear Attention (ONA) Module. Upgraded version of MRA that adding bottleneck after residual to enhance feature extraction.

II. METHOD

The YOLO framework is widely used in object detection due to its balance in the middle of accuracy and efficiency. As evidenced in Figure 1, YOLOv12n incorporates attention-based modules, such as A2C2f and C3K2, to enhance feature extraction and better handle multi-scale features, improving detection performance within a single-stage framework. Additionally, its polished backbone and neck frameworks improve feature aggregation and localization meanwhile keeping computation at low cost. In this study, the YOLOv12-n variant is adopted, with approximately 2.5 M parameters and 6.0 GFLOPs.

A. Backbone

YOLOv12n have backbone that acts as the primary attribute extractor, learning important visual patterns from the input image. Both A2C2f and C3K2 sections elevate feature value while maintaining efficacy, enabling detection of objects at different scales. The A2C2f module enhances feature extraction by combining attention and efficient convolution. It first reduces channel dimensions, then splits the feature map into two routes: one as a bypass and the other processed using either area attention or a C3K block. The attention mechanism encodes global context, while the C3K block improves spatial feature learning [19]. The outputs are then refined using lightweight convolution layers. The C3K2 module is a lightweight feature extractor based on a CSP-style design. It splits the feature map into two sections, where one is applied through convolutional operation and the other serves as a shortcut to improve efficiency. It supports both C3K and C2f structures for flexible feature extraction [20], and the outputs are merged to enhance channel interaction.

B. Neck

The neck acts as a junction between the backbone and the head by combining feature maps from different scales. This process helps produce more informative and discriminative features. YOLOv12 uses a Path Aggregation Network (PAN) network, which allows both top-down and bottom-up component fusion to improve context understanding spanning level networks. In addition, convolution layers and C3K2 parts are used to enhance feature merging whereas keeping the computation efficient [21].

C. Head

YOLOv12n employs an anchor-free detection head established before on YOLOv8, which separates objectness, classification, and bounding-box regression analysis to elevate precision and steadiness. The head uses two concurrent sub-networks with 3×3 and 1×1 convolutions and generates forecasts at three different scales to handle objects of different sizes. For optimization, Complete Intersection over Union (CIoU) loss is applied for bounding-box regression [22], Binary Cross-Entropy (BCE) for classification where the model predicts one of two classes, and Distribution Focal Loss (DFL) to elevate localization stability [23], [24].

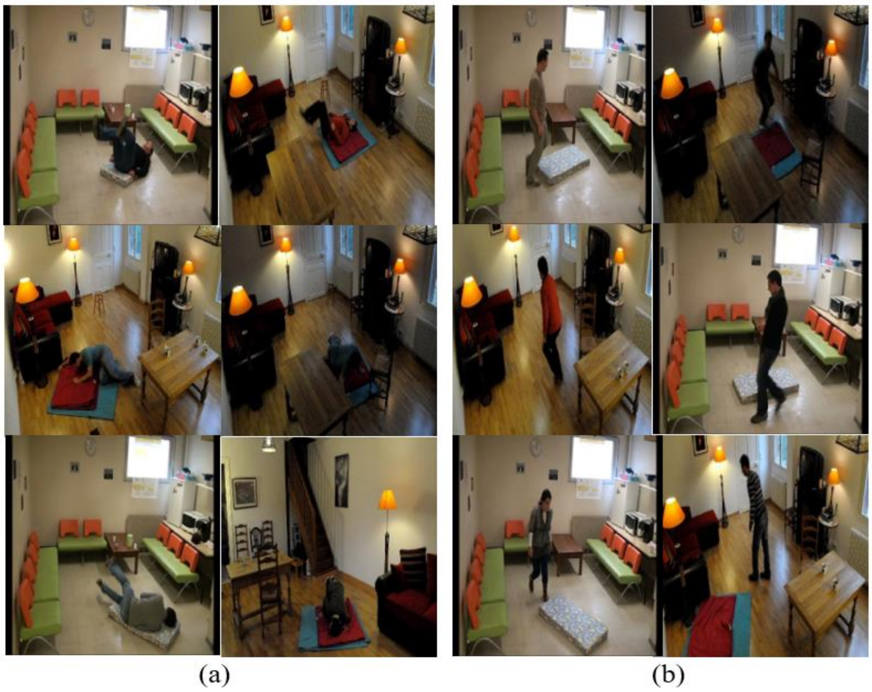

Figure 3 Dataset Le2i Samples. (a) Fallen Position, (b) Standing Position

Figure 4 Fallen Dataset Samples.

Table 1. Training and Testing Environment Settings

Properties	Deployment
Platform	Kaggle
GPU	P100
Image Size	640 x 640
Epochs	300
Batch Size	32
Optimizer	Stochastic Gradient Descent
Learning Rate	0.01

Table 2. Datasets Properties

Dataset	Training Data	Validation Data
Le2i-Roboflow	2.557 images	2.445 images
Fallen-Roboflow	303 images	504 images

D. ONA (Optimized Nonlinear Attention)

Improving feature representation in YOLOv12, the Optimized Nonlinear Attention (ONA) module is introduced. Inspired by Multi-Response Attention (MRA), it highlights important spatial information, distinguish distinctive features and avoid redundancy [25]. Unlike conventional approaches, the attention is applied within the LFP to better integrate high-frequency features at the end of the block. This design uses lightweight convolutional blocks to extract and enhance representative features from partial feature maps by leveraging diverse viewpoints subsequently enhances the relevant components, as illustrated in Figure 2. Notably, multiple summaries can be interpreted as

$$M_s(x_{dw}) = W_u(x_{dw}) \oplus Pool_{avg}(x_{dw}) \oplus Pool_{max}(x_{dw}) \quad (1)$$

where the features of depthwise convolution as

$$x_{dw} = Conv_{dw}(x_i) \quad (2)$$

ONA The Optimized Nonlinear Attention (ONA) module is an enhanced attention mechanism derived from MRA, designed to efficiently capture multi-perspective spatial information. It leverages lightweight operations, such as 1 × 1 convolution along with max and average pooling, to extract diverse feature representations from partial feature maps. These features are then refined through sequential convolutional processing to generate adaptive attention weights, which selectively emphasize important information illustrated as

$$\mathrm{ONA}(x_i) = B(x_{dw}) \otimes \sigma\left(W_z\left(W_v\, M_s(x_{dw})\right)\right) \quad (3)$$

Through MRA, ONA is optimized to remain lightweight and efficient for real-time applications. It focuses on critical patterns (e.g., posture transitions) to enhance feature representation while reducing background noise, improving both localization and classification accuracy. As a result, it achieves higher mAP. ONA consists of a residual bottleneck described as

$$B = x + C_2\left(\mathrm{ReLU}\left(C_1(x)\right)\right) \quad (4)$$

This represents a feature transformation module where the input $x$is processed by a convolution layer C_1, followed by a ReLU activation [26] to introduce nonlinearity, and then refined by a second convolution C_2. A residual pathway implements the groundwork input through transformed output, helping preserve information, improve gradient flow, and enhance feature learning efficiency.

E. Dataset

This research method comprises several main stages. Firstly, data was collected from two sources: the Le2i dataset [27],[28]. It contains 3,010 images as shown on Figure 3, and the Fallen dataset on Figure 4, comprising 3,290 images, used for comparison [29]. Both datasets cover both falling and standing conditions; the images were then annotated using the standard YOLO format to ensure consistency and accuracy of the labels. Next, the data was partitioned into three sections: 75% for training, 10% for validation, and 15% for testing.

F. Implementation Setup

This Network was deployed and trained on the Kaggle environment utilizing a P100 GPU by NVIDIA. The input image dimension was established to 640 × 640 pixels. Training was conducted for 300 epochs with a batch size of 32. The optimization process employed Stochastic Gradient Descent (SGD) with a learning rate of 0.01[30].

Table 3. Comparison of YOLO variants, trade-off between accuracy and model complexity (parameters)

Model	Parameter	mAP50	mAP50;95
YOLOv12n	2,520,054	0.866	0.555
YOLOv11n	2,590,230	0.887	0.560
YOLOv10n	2,707,820	0.859	0.565
YOLOv9t	2,005,798	0.871	0.572
YOLOv8n	3,011,238	**0.927**	0.590
YOLOv12n-ONA-Net	**1,271,258**	0.923	**0.597**

Table 4. Comparison of YOLO variants in terms of model complexity (parameters), computational cost (GFLOPs), and inference speed (FPS), complementing the accuracy analysis in Table 3.

Model	Parameter	GFLOPs	FPS
YOLOv12n	2,520,054	6.0	11.84
YOLOv11n	2,590,230	6.4	**15.94**
YOLOv10n	2,707,820	8.4	12.70
YOLOv9t	2,005,798	7.8	12.56
YOLOv8n	3,011,238	8.2	13.64
YOLOv12n-ONA-Net	**1,271,258**	**4.9**	13.13

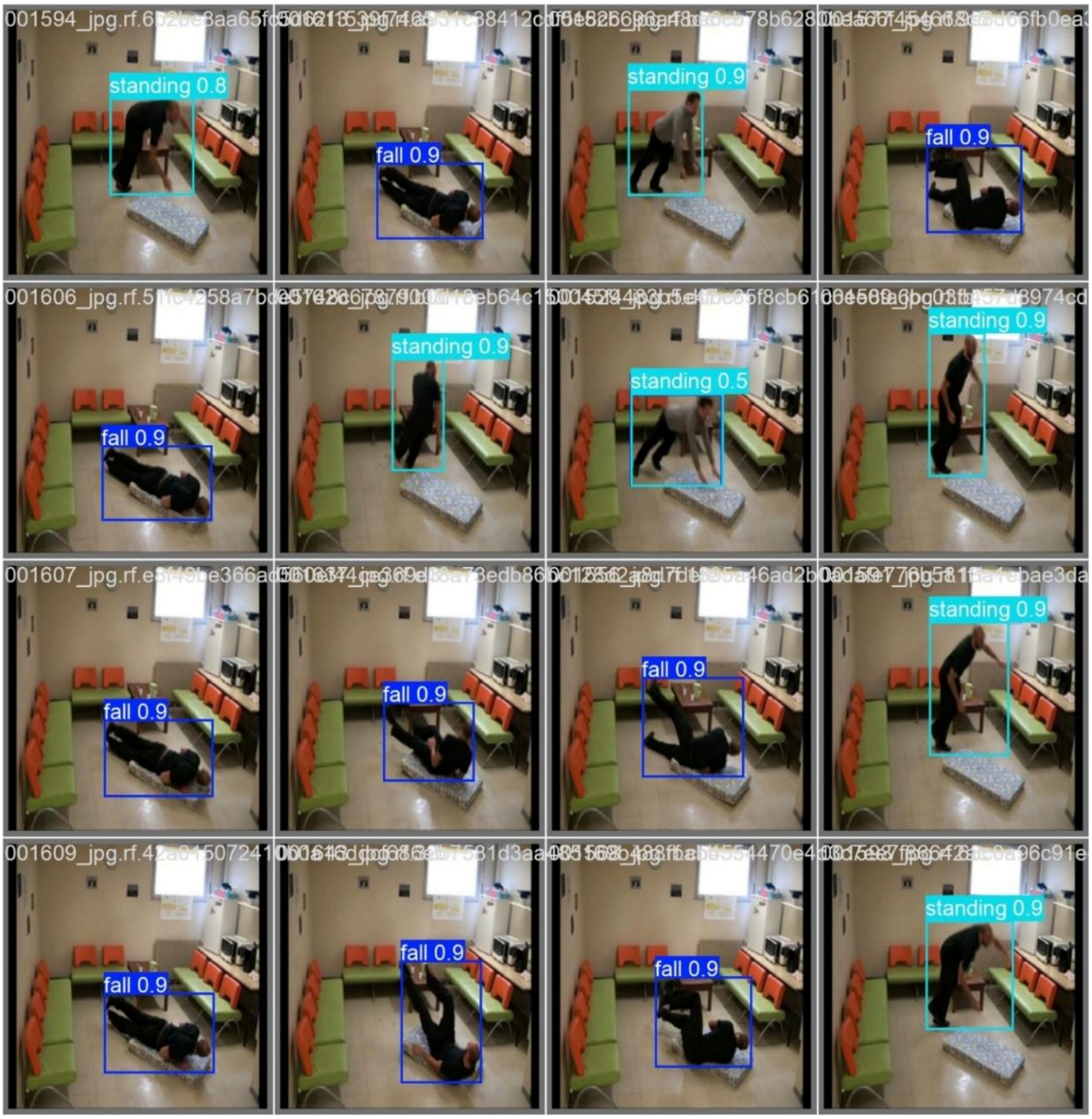

Figure 5 Training Result of YOLOv12-ONA-Net

III. RESULTS AND DISCUSSION

This section illustrates the results of this work with the comprehensive discussion. Focused on three main aspects: dataset evaluation to measure performance and generalization, runtime efficiency to assess computational cost and an ablation study to understand the role of each model part. Both quantitative results and qualitative insights are presented to give a clear view of the model's competence. In addition, comparisons with baseline methods are provided to demonstrate the effectiveness and robustness of the proposed network.

A. Evaluation of Datasets

The model is evaluated using mAP (mean average precision) for accuracy, parameters and GFLOPs for complexity, and FPS for inference speed, providing a comprehensive performance assessment. Training uses the original dataset with default YOLOv12n settings, including mosaic augmentation in the final 10 epochs, without additional modifications; class imbalance is preserved to ensure fair comparison and robustness evaluation if compared with another model. Figure 5 shows stable performance, detecting fall and standing positions with about 0.9 mAP accuracy.

As shown in Table 3, YOLOv12n-ONA-Net outperforms YOLOv12n and other YOLO variants, achieving 0.923 mAP50 and 0.597 mAP50:95, which indicates improved precision with a modest trade-off. This can be observed by comparison with YOLOv8n, which attains the highest mAP@50 of 0.927. In terms of efficiency, the proposed model operates with only 1,271,258 parameters, approximately half the number required by other YOLO-based models, such as YOLOv12n, which uses 2.52 million parameters.

Table 5. Model Analysis of YOLOv12-ONA-Net

Model	Parameter	GFLOPs	mAP50	mAP50;95
YOLOv12n	2,520,054	6.0	0.866	0.555
YOLOv12 tuning channel	1,025,178	3.6	0,915	0,577
YOLOv12n-MRA	1,235,290	4.7	0,898	0,589
YOLOv12n-ONA-Net	**1,271,258**	**4.9**	**0.923**	**0.597**

B. Runtime Efficiency

Table 4 shows that, in addition to having fewer parameters, YOLOv12n-ONA-Net also incurs lower computational cost, requiring 4.9 GFLOPs compared to approximately 6.0 GFLOPs for YOLOv12n. However, in terms of inference performance, the model achieves 13.13 FPS. This speed is higher than that of YOLOv12n but remains lower than other YOLO variants, such as YOLOv11n with faster inference at 15.94 FPS. Although the proposed network exhibits slightly lower inference speed, these results indicate that it remains computationally efficient. Certain trade-offs in the network design may affect inference speed; however, the performance is still acceptable and remains suitable for real-time detection applications [31].

C. Model Analysis

The proposed YOLOv12-ONA-Net model is developed based on YOLOv12n through a two-stage modification process. First, a channel tuning strategy is applied by pruning the number of channels, which significantly reduces the model parameters (from 2.52M to 1.03M) and computational cost (from 6.0 to 3.6 GFLOPs). This step improves efficiency but also leads to a slight trade-off in representation capacity. Addressing this limitation, the Optimized Nonlinear Attention (ONA) module is then integrated into the YOLOV12n pruned model. The addition of ONA-Net on A2C2f and C3k enhances feature representation by introducing nonlinear attention and residual learning, allowing the network to capture more complex patterns. As a result, the model parameters and GFLOPs increase to 1.27M and 4.9 GFLOPs, respectively. Despite this increase, the model remains lightweight compared to the original YOLOv12n. This modification leads to a consistent improvement in detection performance, where YOLOv12-ONA-Net obtain the utmost precision with an mAP50 of 0.923 and mAP50:95 of 0.597 as shown in Table 5. This demonstrates that the integration of ONA effectively balances efficiency and accuracy, lead to the proposed network suitable for CPU devices.

D. Ablation Study

Table 5 shows about the ablation study processed in modification of YOLOv12n network. It examines contribution effect of bottleneck that incorporated into MRA block that achieve 2.5 % increase on mAP50 and 0.8 % increase on mAP50:95 if compared to YOLOv12-MRA with 0.2 more GFLOPs used on operation. Compared with YOLOv12n however there is a significant increase around 5.7 % on mAP50 and 4.2 %.

IV. CONCLUSION

The proposed YOLOv12n-ONA-Net demonstrates strong performance in fall and standing position detection while maintaining a compact and efficient architecture. With reduced parameters and computational cost compared to other YOLO variants, the model provides a favourable balance between detection accuracy and efficiency. The incorporation of the bottleneck in the MRA block contributes significantly to performance improvement, as confirmed by the ablation study, indicating that enhanced spatial feature representation plays a key role in accurate fall detection. From a deployment perspective, the model is well-suited for CPU devices due to its lightweight design and reduced computational demand. Although the achieved inference speed of 13.13 FPS is not the highest, it remains sufficient for fall detection scenarios, where events occur over short but observable durations rather than requiring ultra-high frame rates. This makes the model practical for real-time monitoring CPU-based systems.

V. ACKNOWLEDGMENTS

The authors sincerely give appreciation and thanks to AIVISION studio research team for their support and advice, discussion and proficiency in deep learning and vision-based computing also with resources that become pillar that supported this work.

REFERENCES

[1] N. T. Newaz and E. Hanada, “The methods of fall detection: A literature review,” Sensors, vol. 23, no. 11, p. 5212, 2023. doi: 10.3390/s23115212.

[2] J. Gutiérrez, V. Rodríguez, and S. Martin, “Comprehensive review of vision-based fall detection systems,” *Sensors,* vol. 21, no. 3, p. 947, 2021. doi: 10.3390/s21030947.

[3] S. Kumar et al, “Falls among older adults: An exploration of trends, clinical outcomes, predisposing risk factors, and intervention strategies,” The American Journal of Surgery, vol. 245, p. 116385, 2025. doi: 10.1016/j.amjsurg.2025.116385

[4] B. Chen et al., “Computer vision and machine learning-based gait pattern recognition for flat fall prediction,” Sensors, vol. 22, no. 20, p. 7960, 2022. doi: 10.3390/s22207960.

[5] S. Juraev, A. Ghimire, J. Alikhanov, V. Kakani and H. Kim, "Exploring Human Pose Estimation and the Usage of Synthetic Data for Elderly Fall Detection in Real-World Surveillance," in IEEE Access, vol. 10, pp. 94249-94261, 2022, doi: 10.1109/ACCESS.2022.3203174.

[6] N. T. Newaz and E. Hanada, “The methods of fall detection: A literature review,” *Sensors*, vol. 23, no. 11, p. 5212, 2023. doi: 10.3390/s23115212.

[7] P. Yozgatli *et al.*, “Fall detection in passenger elevators using intelligent surveillance camera systems: An application with YOLOv8 Nano model,” *arXiv preprint arXiv:2501.01985*, 2024. doi: 10.48550/arXiv.2501.01985.

[8] G. A. Pereira, “Fall detection for industrial setups using YOLOv8 variants,” *arXiv preprint arXiv:2408.04605*, 2024. doi: 10.48550/arXiv.2408.04605.

[9] F. X. Gaya-Morey, C. Manresa-Yee, and J. M. Buades-Rubio, “Deep learning for computer vision-based activity recognition and fall detection of the elderly: A systematic review,” *Applied Intelligence*, vol. 54, no. 19, pp. 8982–9007, 2024. doi: 10.1007/s10489-024-05434-9.

[10] A. Priadana, D. -L. Nguyen, X. -T. Vo, J. Choi, R. Ashraf and K. Jo, "HFD-YOLO: Improved YOLO Network Using Efficient Attention Modules for Real-Time One-Stage Human Fall Detection," in *IEEE Access*, vol. 13, pp. 41248-41258, 2025, doi: 10.1109/ACCESS.2025.3547360.

[11] A. Raza, M. H. Yousaf and S. A. Velastin, "Human Fall Detection using YOLO: A Real-Time and AI-on-the-Edge Perspective," *2022 12th International Conference on Pattern Recognition Systems (ICPRS)*, Saint-Etienne, France, 2022, pp. 1-6, doi: 10.1109/ICPRS54038.2022.9854070.

[12] M. L. Ali and Z. Zhang, “The YOLO framework: A comprehensive review of evolution, applications, and benchmarks in object detection,” *Computers*, vol. 13, no. 12, p. 336, 2024. doi: 10.3390/computers13120336.

[13] Y. Tian, Q. Ye, and D. Doermann, “YOLOv12: Attention-centric real-time object detectors,” arXiv preprint, arXiv:2502.12524, 2025, doi: 10.48550/arXiv.2502.12524.

[14] P. D. K. Manembu, J. I. Litouw, F. D. Kambey, A. H. J. Ontowirjo, V. C. Poekoel, and M. D. Putro, “Efficient fall detection using lightweight network to enhance smart internet of things,” International Journal of Electrical and Computer Engineering (IJECE), vol. 15, no. 5, pp. 5031–5044, 2025, doi: 10.11591/ijece.v15i5.pp5031-5044.

[15] H. Ren and P. Lan, “BMR-YOLO: A deep learning approach for fall detection in complex environments,” PLoS One, vol. 20, no. 11, p. e0335992, 2025, doi: 10.1371/journal.pone.0335992.

[16] J. Hu, L. Shen, and G. Sun, “Squeeze-and-Excitation Networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018, pp. 7132–7141, doi: 10.1109/CVPR.2018.00745.

[17] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional Block Attention Module,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018, pp. 3–19, doi: 10.1007/978-3-030-01234-2_1

[18] H. Prasetya, R. R. Balo, T. Tumbal, A. M. Sambul, and M. D. Putro, “Comparative Study of Lightweight YOLOv12 Models for Real-Time Underwater Object Detection,” *Jurnal Telematika*, vol. 20, no. 2, pp. 127–137, 2025, doi: 10.61769/telematika.v20i2.799

[19] J. Song, J. Xie, Q. Wang, and T. Shen, “An improved YOLO-based method with lightweight C3 modules for object detection in resource-constrained environments,” Journal of Supercomputing, vol. 81, p. 702, 2025, doi: 10.1007/s11227-025-07187-w.

[20] X. Wu and S. Yu, “CSP-YOLO: An efficient and lightweight real-time algorithm for internal rail defect detection,” *The Journal of Supercomputing*, vol. 81, no. 15, p. 1450, 2025. doi: 10.1007/s11227-025-07949-6.

[21] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, “Path aggregation network for instance segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 8759–8768. doi: 10.1109/CVPR.2018.00913.

[22] X. Wang and J. Song, "ICIoU: Improved Loss Based on Complete Intersection Over Union for Bounding Box Regression," in *IEEE Access*, vol. 9, pp. 105686-105695, 2021, doi: 10.1109/ACCESS.2021.3100414

[23] U. R. Dr. A., “Binary cross entropy with deep learning technique for image classification,” Int. J. Adv. Trends Comput. Sci. Eng., vol. 9, no. 4, pp. 5393–5397, Aug. 2020, doi: 10.30534/ijatcse/2020/175942020.

[24] X. Li et al., “Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection,” Jun. 2020. [Online]. Available: http://arxiv.org/abs/2006.04388.

[25] M. Dwisnanto Putro, A. Priadana, D. -L. Nguyen and K. -H. Jo, "EMOTIZER: A Multipose Facial Emotion Recognizer Using RGB Camera Sensor on Low-Cost Devices," in *IEEE Sensors Journal*, vol. 25, no. 2, pp. 3708-3718, 15 Jan.15, 2025, doi: 10.1109/JSEN.2024.3493947.

[26] A. Mujhid, S. Surono, N. Irsalinda, and A. Thobirin, “Comparison and combination of Leaky ReLU and ReLU activation function and three optimizers on deep CNN for COVID-19 detection,” in *Fuzzy Systems and Data Mining VIII: Proc. FSDM 2022*, pp. 50–57, 2022. doi:10.3233/FAIA220369.

[27] X. Kan, S. Zhu, Y. Zhang, and C. Qian, “A lightweight human fall detection network,” Sensors, vol. 23, no. 22, p. 9069, 2023. doi: 10.3390/s23229069.

[28] D. Schneider, Z. Marinov, R. Baur, Z. Zhong, R. Düger, and R. Stiefelhagen, “OmniFall: A unified staged-to-wild benchmark for human fall detection,” *arXiv preprint*, arXiv:2505.19889, 2025. doi: 10.48550/arXiv.2505.19889.

[29] B. ByoungWook, “fallen_new_version Dataset,” Roboflow Universe, Roboflow, Jan. 2023. [Online]. Available: https://universe.roboflow.com.

[30] I. Dagal, K. Tanrıöven, A. Nayir, and B. Akın, “Adaptive stochastic gradient descent (SGD) for erratic datasets,” *Future Generation Computer Systems*, vol. 166, p. 107682, 2025. doi: 10.1016/j.future.2024.107682.

[31] J. Lee and K. Hwang, “YOLO with adaptive frame control for real-time object detection applications,” *Multimedia Tools and Applications*, vol. 81, pp. 36375–36396, 2022, doi: 10.1007/s11042-021-11480-0