Volume 11. No 2, May 2024
e-ISSN: 2527-9572 / ISSN: 2354-8924

Speedy Vision-based Human Detection Using Lightweight Deep Learning Network

Gede Erik Aktama
Department of Information System, Parna Raya University, Jl. Sam Ratulangi 1 No.2-3, Wenang, Manado, Indonesia
gedeaktama@microsoft.parnaraya.ac.id

Franky Manoppo
Department of Informatics, Parna Raya University, Jl. Sam Ratulangi 1 No.2-3, Wenang, Manado, Indonesia
cliford@microsoft.parnaraya.ac.id

Rosdiana Simbolon
Department of Information System, Parna Raya University, Jl. Sam Ratulangi 1 No.2-3, Wenang, Manado, Indonesia,
rosdiana@microsoft.parnaraya.ac.id

Adityo Clinton Laloan
Department of Information System, Parna Raya University, Jl. Sam Ratulangi 1 No.2-3, Wenang, Manado, Indonesia,
adityo@microsoft.parnaraya.ac.id

Andreas Sumendap
Department of Computer System, Parna Raya University, Jl. Sam Ratulangi 1 No.2-3, Wenang, Manado, Indonesia,
andmendap@gmail.com

Muhamad Dwisnanto Putro
Department of Electrical Engineering, Faculty of Engineering, Sam Ratulangi University. Jl. Kampus Bahu, Manado, Indonesia,
dwisnantoputro@unsrat.ac.id

***Abstract* – Person detection plays a role as the initial system of video surveillance analysis with various implementations, such as activity analysis, person re-id, behavior analysis, and tracking analysis. The demand for efficient models drives a deep learning architecture with a superficial structure that can operate in real-time. You look only once (YOLO) object detection has been presented as an accurate detector that can operate in real-time. The speed limitation, huge computation cost, and abundant parameters still leave vital issues to improve the efficiency of this architecture. Lightweight human detection is proposed by utilizing the YOLOv5n framework. Modifying layer depth promotes a detection system that can operate fast and without stuttering. As a result, the proposed detector has satisfactory performance and is competitive with existing models. It achieves a mAP of 45.2%, closely competing with other person detectors. Additionally, it can run fast without stumbling at 26 frames per second. The detector's speed offers the advantage of this work that it can be feasibly implemented on a cpu device without a graphics accelerator.**

Keywords: Person detection, efficient YOLO, real-time detector, central processing unit. surveillance system.

I. INTRODUCTION

Human detection is a trend work by determining the location of human presence. In its development, this system can be used for activity analysis [1], person re-id [2], behavior analysis [3], and tracking analysis [4]. Vision systems offer high performance to identify the distinctive features of a person. This approach presents an accurate object detection system that can separate human information from the background. Person contains a specific combination of elements such as head, hands, feet and body. Each of these components has different shape, texture and position information [5]. A vision approach must be able to discover these components precisely and relate the information to each other for a decision. Determining a robust algorithm can decide the success of the detection system, and vice versa. Over the years, the challenge of video surveillance provides small objects with changing illumination. A person detector must be adaptable and able to discriminate between human features and other objects to avoid prediction errors [6]. Moreover, implementation as a surveillance system also emphasizes algorithms that can perform computations with a slight delay.

Modern methods offer accurate object detection systems that can adapt to dynamic challenges. Deep learning encourages a vision system to learn features of interest by employing deep layers. Convolutional Neural Network (CNN) is a deep learning approach that utilizes multiple layers and applies filter operations [7]. This operation allows the network to learn features through the extraction of spatial regions. The kernel weights are updated automatically which can find patterns from objects. Several popular architectures have been presented by works [8], [9], [10] that achieve high performance for recognizing objects from an image. Instead of requiring a small amount of computation and trained weights, these models rely on graphics accelerator devices to operate in real-time. Feature extraction plays an important role in person detection because it is required to separate person elements from complex backgrounds [11]. The challenges of lighting distortion, scale and pose confuse the weak approach. Therefore, a robust person detection system is a priority. In addition, practical applications provide dependence on cheap devices and

force vision systems to operate in real-time [12]. Object detection has become a popular intelligence vision research in recent years. There are two categories of DNN detectors: the two-stage approach and the one-stage approach. The first is to generate a series of candidate boxes and then do the classification and regression later. The other is when the problem is directly transformed into a regression problem without producing candidate boxes. One-stage methods like SSD [13], YOLO series [14], [15], [16], and RetinaNet [17] have a faster detection speed than the two-stage one, in this case, the Faster R-CNN [18].

The YOLO network is well known for detecting multiple objects in real time, and the feature maps of different scales are sorted hierarchically. YOLOv5 [19] is the most usual network model because of its efficiency and better results than previous versions when testing in real-time. Due to the limitation of memory resources and power consumption of the device, running deep neural network models and achieving better object detection results on mobile devices in real-time is still a huge challenge. Based on this motivation, we improve YOLOv5 to localize person objects by increasing efficiency. In this paper, a real-time detection system using a deep learning approach is proposed to identify the presence of a person which is feasible to be implemented on low-cost devices. It utilizes the advantages of YOLOv5 which precisely recognizes the characteristics of target features. The proposed network improves the efficiency of YOLOv5 nano which aims to enable the detector to run smoothly and faster than the standard version. The balance of performance and efficiency leads this detection system to be embedded in a device that can run all day continuously. The contributions of this work can be summarized as follows.

1. A novel lightweight person detector is proposed by efficiently modifying the YOLOv5 framework that reliably localizes important elements of the person. It utilizes C3 (three convolutional blocks) feature extractor, SPPF (spatial pyramid pooling faster), PAN (Path aggregate network), and multi-level detection layers. In its application, this detector is specialized for surveillance video analysis.
2. The performance of the person detector is comprehensively evaluated on benchmark datasets, including MS COCO 2017 [20], PASCAL VOC 2007 [21], and PASCAL VOC 2012 [21]. Besides, the model efficiency analysis shows that the proposed detector can operate feasibly on Central Processing Unit (CPU) devices without a graphics accelerator, without constraints.

II. Related Works

There are many amounts of person detection, and it works excellently, improving later. However, the challenge is to find a deep network that can get perfect results while working efficiently on mobile devices. Ming Xu et al. [22] improved YOLOv3 with two shuffle modules (inspired by ShuffleNet) and included the K-means algorithm. It is to replace the Non-Maximum Suppression (NMS) that is time-consuming when detecting objects. It achieves higher mean average precision (mAP) scores with lower parameters than similar networks (YOLOv3 and tiny-YOLOv3) using the CrowdHuman detection dataset. About the frames per second (FPS), the network can process in real time, and the IoU scores are higher and more accurate. The only drawback is the network is still slower compared to tiny-YOLOv3 in terms of network speed because it is deeper than tiny-YOLO.

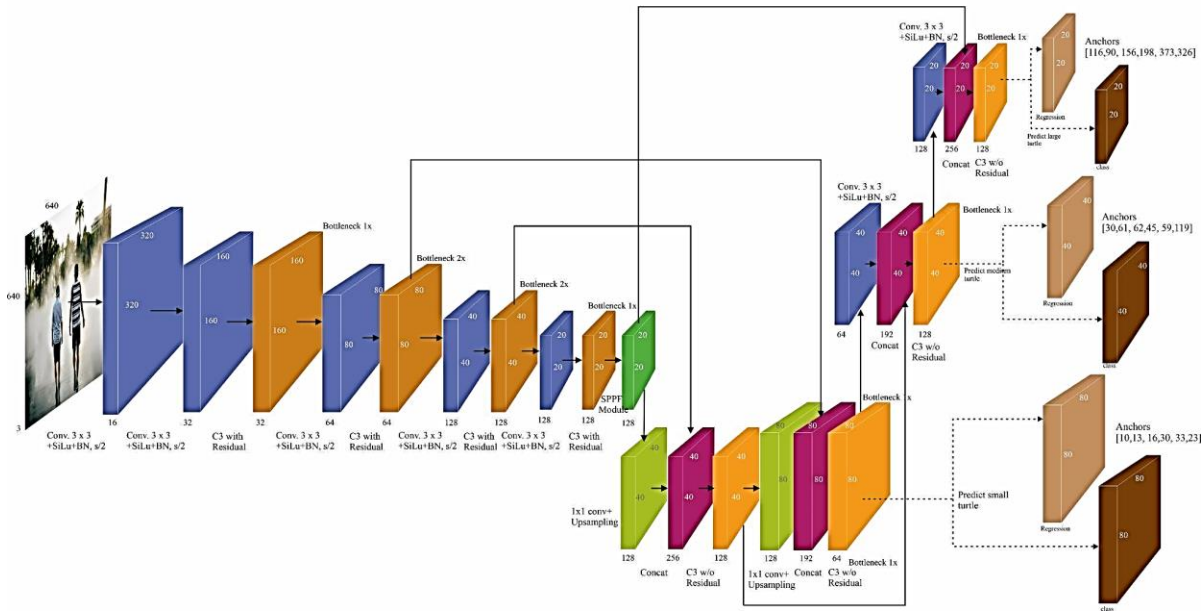

Figure 1. The architecture of human detection improved YOLOv5n. Best in color viewed.

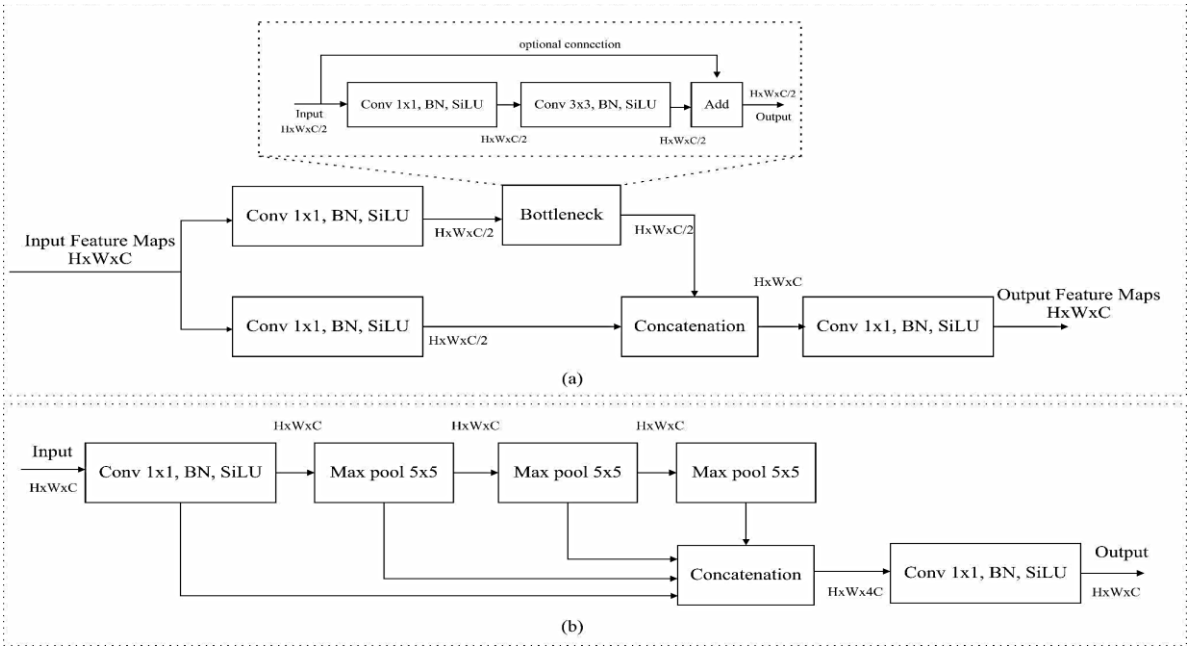

Figure 2. The three convolutional blocks (a) and spatial pyramid pooling faster (b).

Nguyen et al. [23] modified the YOLOv2 network, starting from the residual block by changing the backbone of YOLOv2 from 26 layers to 23 layers, until the Multiple Spatial Pyramid Pooling (MSPP) by adding two SPP blocks. This network obtains high precision, and the memory usage is lower than the models, especially with the YOLO one. However, the memory usage is slightly larger than the SSD-based L-CNN one, which is the disadvantage of these methods.

Another work proposed a pedestrian detection network (PDNet) [24]. This network uses an Efficient Bottleneck Partitioning (EBP) block and a Path Aggregation Network, which helps the network separate the feature map swiftly into two partitions and efficiently mix the different features from low-level features to high-level features. The EBP uses simple convolution to reduce the computation of the network. There are consequences for the higher FPS, which is that the accuracy is slightly lower because of the lack of depth in the deep network. Further development has presented Fast-PdNet [25], which improves the efficiency of the network. This detector focuses on data processing speed and thus applies a shallow convolutional layer. However, the performance produced by this detector was still weak in finding tiny objects.

Although deep learning networks are used widely in object detection, especially human detection, sometimes there needs to be a more particular aspect ratio for detection algorithms. A study [26] solved that problem with the ratio resizing the image without getting deformed or distorted, which leads to a failure in detections. It also uses the segmentation function to split some people in one image that does not overlap into some sub-image. It helps to reduce the occurrence of false positives. These sub-images come in different sizes or ratios. It can run fast on inference time. The hyperparameters are still adjusted manually, and the operation of the segmentation function takes much time to run.

III. PROPOSED NETWORK

Object detection generally employs several blocks that sustain each other to produce high precision. It deploys a backbone as the main feature extractor, an aggregate module transitions and combines features of different levels, and the detection layer to predict bounding box and class probability. YOLO presents a deep CNN structure that contributes to locating multiple objects in a frame. This architecture emphasizes feature learning from ground truth knowledge and several preprocessing techniques that are claimed to improve training performance without compromising data processing speed at the inference stage.

Throughout its development, YOLOv5 has been established as a reliable object detection that can operate on image and video inputs. The efficiency model led to the development of this framework to run fast on low-cost devices. The slim architecture employs a backbone, path aggregate network, and multi-detection layer assisted by nine anchors to initialize bounding box predictions. The details of this efficient development are described in this section. The network of detector is presented in Fig. 1

A. Backbone Module

A backbone is a vital module in determining the success of obtaining essential information from an object. It contains several convolutional blocks that cooperate through a combination of trained filter weights. YOLOv5 applies a 3 x 3 filter operation at the beginning of the stage to reduce the dimension of the feature map. Instead of using pooling to reduce the map size, it applies 3 x 3 convolutional, which is more robust. This operation is also applied at each stage to ensure the target object features are not lost. This filter operation has been tested optimal and widely utilized

by previous studies [9], [10]. This spatial field determines the center point that can balance the information linkage between neighbors. On the other hand, this filter generates fewer parameters than other balanced spatial regions.

Three convolutional blocks (C3) are employed on the backbone part to extract human-specific features efficiently, as shown in Fig 2 (a). It is inspired by the bottleneck CSP structure, which splits the feature map into two and then extracts only one part. It helps the network to save the computational usage of convolution operations. 1 x 1 convolutional is applied to generate two different feature maps, each acquiring half the number of input maps. This reduction ensures that the feature map is split in half fairly to support the network only utilizing sequential convolutional blocks on one part.

The bottleneck module extracts one part of the map by applying two 1 x 1 convolutional layers. This network allows for an increasing number of bottleneck modules. Furthermore, a concatenation operation is used to fuse two separate feature maps. Thus, it ensures that no essential features are reduced. This operation combines two maps by arranging the first and second maps so that this technique can double the number of channels. The final part of the C3 block is mixing each unit pixel from a row of channels containing different information. Applying a 1 x 1 kernel operation can strengthen the feature representation. For efficiency purposes of the proposed detector, we limit the maximum number of channels to 256 in all convolution layers. It clearly reduces the computational complexity and parameters, which directly impacts improving the speed of the detector.

An SPPF is utilized after the C3 module to find high-value features from various spatial regions, as presented in Fig 2 (b). Valuable information appears as a high score on a feature map. This module helps the network to summarize such valuable features without significantly reducing the data processing speed. SPPF employs three consecutive max-pooling with a window size of 5 x 5. This structure allows the second and third pools to acquire larger receptive fields than the first layer, which are 9 x 9 and 13 x 13, respectively. At the end of this module, a 1 x 1 convolution is used to mix features with different pooling levels and construct the number of channels from the module output.

B. Path Aggregate Network

Path Aggregation Network (PANet) is an improved version of FPN (Features Pyramid Network) that uses two pyramids with the addition of the bottom-up pyramid. The reasons for using these pyramids include shortening the information path using lateral connections, recovering some information lost between each level using adaptive feature pooling, and gathering the information on each level, from low to high level. The difference between PAN and FPN is an addition of the bottom-up pyramid that is similar to the FPN pyramid, but it starts from a lower to a higher level. The bottom-up path has enhanced the capability of localization that low-level and high-level are crucial to an indicator of predicting localized instances by using a lateral connection. This lateral connection becomes a shortcut for information being passed from low-level to high-level that only takes less than ten layers, instead of an FPN that takes a long path (passing more layers from low to high). The C3 module plays a vital role in extracting features in the PAN architecture. It divides the feature map into two parts and merges it using a cross-stage hierarchy. It is similar to an extractor in the backbone. However, the residual branch of C3 in PAN is not applicable because it avoids differences in combining feature dimensions and increases accuracy in the prediction process. The advantage of applying this module in PANet is to improve the performance of detectors that fuse different frequency information. In addition, it enriches the variety of features that encourage the network to derive element options from different objects.

C. Multiple Detection Layers

This module plays a vital role in predicting the bounding boxes and class of objects. We apply three feature maps with different sizes from PAN to be utilized as detection layers. These layers are 20 x 20, 40 x 40, and 80 x 80. The smallest feature map is utilized to predict large humans. The medium feature map is used to predict medium humans, and the large map is used for small objects. A convolutional layer produces each detection layer containing 1 x 1 filters. As a result, it generates a feature map for each detection layer responsible for bounding boxes regression, objectness prediction, and object probability prediction (human and none). To effectively predict the size and location of the human bounding box, it applies three anchors for each of the three detection layers. The anchor assignments involve different dimensions to represent human size and pose variations. Furthermore, a Multi-boxes loss function [19] is utilized to calculate the prediction error that focuses on finding the difference of the bounding box dimensions and locations, object presence, and the difference in probability of each class.

D. Preprocessing in Training Phase

Object detection applies preprocessing to the input image to generate various data. Moreover, it also helps improve the detector's performance when implemented in real applications. The proposed network employs color distortion, lighting variation, rotate, crop, flip, scale, and affine transformation to manipulate each image's color, brightness, and geometry transformation from the dataset. Instead of inserting images one by one in a batch to be trained, it applies a frame mosaic that puts several augmented images on a frame randomly. This process aims to generate a variety of scales from the instances, which enables the network to get diverse scaling information from an object. In addition, some partially truncated humans will enrich the variety of occlusion data and directly improve the detector's ability in real-case environments.

E. Training and evaluation configuration and datasets

In order to prevent saturation and vanishing gradient training results, several configurations are set

in this work. The proposed network was simulated on Python code with Pytorch deep learning frameworks. It uses GTX1080Ti as a graphic accelerator only in the training phase. We apply complete IoU and cross-entropy loss to calculate the difference between prediction and target, which refers to the study [19] with Adam optimizer. A learning rate of 0.001 is employed during the training stage with 200 epochs. It divides the dataset into 64 batches to speed up the training process. In the testing stage, we use CPU devices to measure the model speed. It utilizes Intel core i7 6600T with RAM of 32, which represents a low-cost device.

Furthermore, the proposed network was trained and evaluated on three datasets, a subset of object detection, including MS COCO, PASCAL VOC 2007, and PASCAL VOC 2012. We trained the detector knowledge on the MS COCO dataset only, which contains a variety of instances. To fairly evaluate the detector, we used the same preprocessing and augmentation configuration with the work [19].

IV. Results and Discussion

In order to measure the effectiveness and efficiency of the proposed model, we evaluate it in this section by observing the detection performance and calculating the number of parameters, computation, and data processing speed of the model. The application of video surveillance leads this evaluation testing on a device without graphic accelerators. In addition, we also compare these metrics with some existing work as well as the lightweight YOLO method.

A. Qualitative Results

This subsection observes the prediction result of the proposed detector by investigating the bounding boxes drawing. It illustrates the box prediction through the green area, as illustrated in Fig. 3. This area indicates the detected human region, making it easier to observe the success and failure of the system. It also provides the system prediction's object (person) class information on the top of a box. In addition, the confidence score is also shown beside the prediction class, which indicates the prediction value of the model with a range of 0 - 1. As a result, the proposed detector can locate humans with various object poses.

In order to evaluate the reliability of the system, we tested the proposed detector system under different lighting and environmental conditions. Fig 3 (a) shows the proposed detector applied to an outdoor environment with direct lighting from sunlight. The testing video is taken during the daytime, which contains complex textures and colors. With this challenge, the proposed detector can easily localize the human. It even performs satisfying results when locating objects that are far away from the view camera. This object has the challenge of small size with limited information on human body elements.

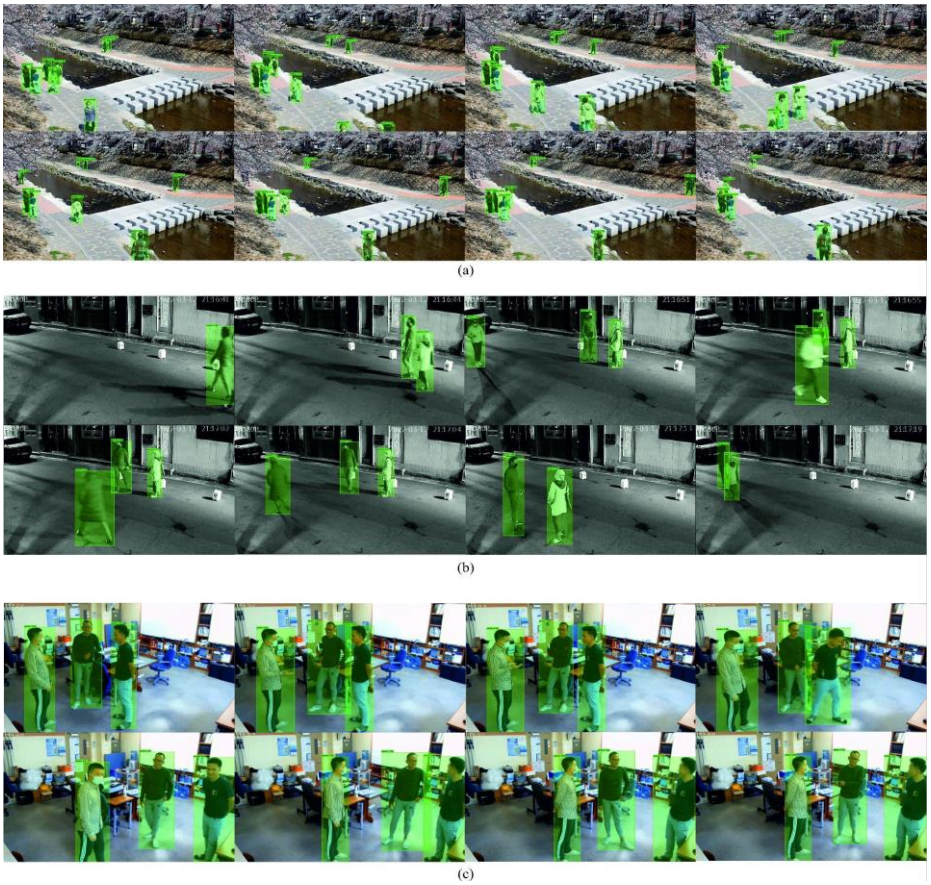

Figure 3. The qualitative result of the proposed human detection. The detector is tested outdoors (a), the detector is tested outdoors at nighttime using infrared video (b), and the detector is tested indoors with a university environment (c). Best in color viewed.

Table 1 Comparisons of human detection results on MS COCO, VOC 2007, and VOC 2012

MS COCO Dataset		
Model	**AP**	**Backbone**
ResNet18	40,10	ResNet18
ShuffleNetv2 (0.5)	33,80	ShuffleNetv2
MobileNetv2 (0.33)	39,10	MobileNetV2
PeleeNet (0.5)	41,90	PeleeNet
Tiny model - bai et al.	42,90	Manually Designed
Tiny-YOLOv2	44,97	DarkNet19
Fast-PdNet [25]	41,60	Manually Designed
YOLOv5n [19]	46.78	C3 Bottleneck
The Proposed	**45.20**	**C3 Bottleneck**
PASCAL VOC 2007 Dataset		
Model	**AP**	**Backbone**
Improved Faster-RCNN	75,65	VGG16
Tiny-YOLOv2	63,88	DarkNet19
Tiny-YOLOv3	68,54	DarkNet19
Improved Tiny-YOLOv3	73,98	DarkNet19
Enhanced Tiny-YOLOv3	78,64	DarkNet19
YOLOv5-nano [19]	86,20	C3 Bottleneck
YOLOv5-small [19]	88,80	C3 Bottleneck
Fast-PdNet [25]	82,70	Manually Designed
The Proposed	**86.50**	**C3 Bottleneck**
PASCAL VOC 2012 Dataset		
Model	**AP**	**Backbone**
Faster R-CNN	62.90	VGG16
SSD512	39.40	VGG16
RefinedDet320	58.50	VGG16
RefinedDet320+	61.60	VGG16
RefinedDet512	63.60	VGG16
RefinedDet512+	66.00	VGG16
RFBNet300	29.00	VGG16
RFBNet512-E	32.60	VGG16
RFBMobileNet	23.80	MobileNet
RetinaNet	60.70	ResNet-50
YOLOv5-nano [19]	86.60	C3 Bottleneck
YOLOv5-small [19]	88.90	C3 Bottleneck
Fast-PdNet [25]	83.60	Manually Designed
The Proposed	**86.90**	**C3 Bottleneck**

On the other hand, we also tested the detector in an indoor environment. The test video also contains a complex background, as shown in Fig. 3 (c). This video was recorded in a laboratory environment that has varied background colors. Cabinets, chairs, blackboards, and books present a challenge for the proposed network to discriminate the features of these objects. However, these challenges do not make detecting a full-body human complicated. The results shown by the visualization illustrate that this detection system can accurately predict humans.

Furthermore, we also tested the reliability of the proposed system that can operate at night. It uses a video from an infrared camera operated at night with limited lighting. Fig. 3 (b) shows that each video frame contains a grayscale. Despite the limitation of color information, this does not prevent the proposed detector from precisely predicting the full-body human. These results indicate that the proposed detector system is reliable for video surveillance systems operating simultaneously with different lighting conditions.

B. Quantitative Results

The proposed detector is comprehensively evaluated for its performance by measuring recall and prediction precision. It combines these two metrics in the mean average precision (mAP). Based on MS COCO as the dataset used, it considers two criteria of intersection over union scores. We use a default IoU of 0.5. Meanwhile, the primary performance calculates the number of true positives of all predictions, which sets the average of the IoU threshold from 0.5 to 0.95. The experiments demonstrate that the proposed model compares with other competitors and previous studies on MS COCO, PASCAL VOC 2007, and PASCAL VOC 2012, as shown in Table 1. When evaluated in the MS COCO dataset, our module outperforms Fast-PdNet [24]. The competitor uses a shallow network with a hierarchical feature pyramid structure to focus on time processing speed. However, the proposed model is still under the performance of YOLOv5n [19], which differs by 1.5%.

Furthermore, we evaluate the precision of the proposed detector on the PASCAL VOC 2007 dataset. It shows that our detector outperforms YOLOv5n and FastPdNet are the closest competitors. While other works such as Improved Faster-RCNN, Tiny-YOLOv2, Tiny-YOLOv3, Improved Tiny-YOLOv3 [14], and Enhanced Tiny-YOLOv3 [14] yield lower performance than the proposed detector. On the other hand, the proposed detector is under the performance of YOLOv5s. It presents a difference of 2.35 from this competitor. This dataset has fewer instances than MS COCO and sets the IoU threshold at 0.5. Therefore, the performance of the detectors evaluated in this dataset is higher than the MS COCO dataset. In this study, we also compare the performance of the proposed human network on the PASCAL VOC 2012 dataset. It contains more instances than the previous version. As a result, the proposed detector achieves an mAP of 86.9%, which outperforms RetinaNet, YOLOV5-nano [13], and Fast-PdNet. Although YOLOV5-small [13] outperforms the proposed human detector, this competitor operates slower in the inference stage. It implies that our network also focuses on efficiency, highlighting the sector implementation's advantages.

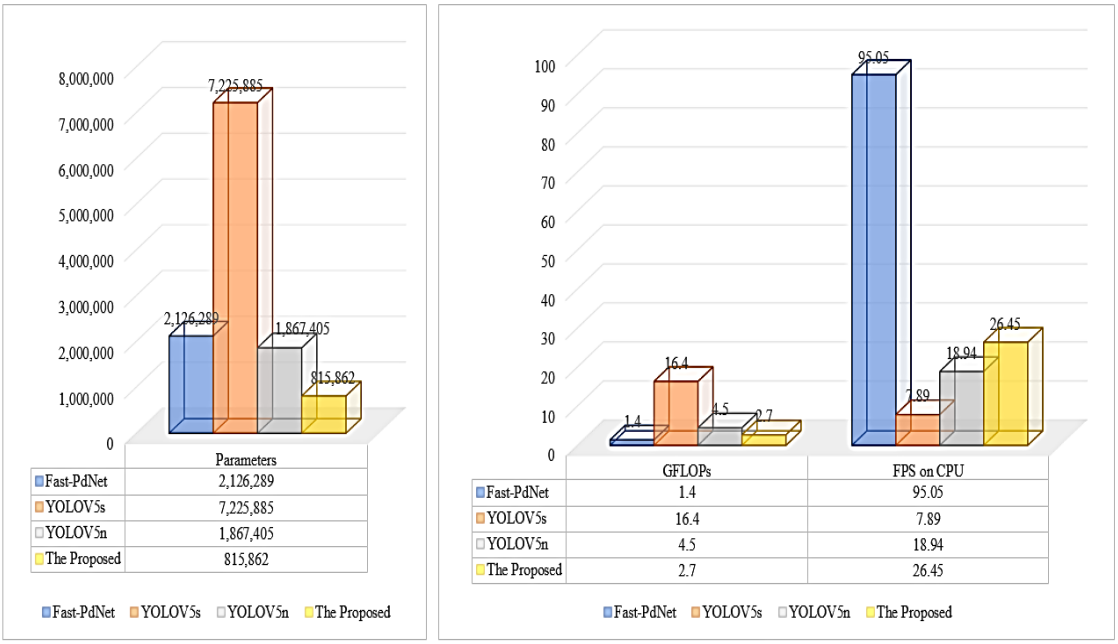

Figure 4. Model efficiency evaluation. Number of parameter comparison (a). Comparison of model computational complexity and data processing speed (b). Best in color viewed.

C. Efficiency comparison

Deep learning networks bring the efficiency issue as a vital discussion to be analyzed. At the same time, these algorithms tend to require huge trainable variables and produce expensive computations to predict objects in complex challenges. In addition, object detectors tend to utilize inefficient networks that employ deep convolution layers with a very large number of channels. On the other hand, practical applications also require object detectors to run in real time, especially in surveillance cameras.

The system has to process an input frame quickly by minimizing the delay time to encourage rapid prediction and further analysis. The proposed model produces small parameters of 815.862. It is lighter than other detectors, as illustrated in Fig 4(a). In addition, our network only requires a computational complexity of 2.7 GFLOPs to extract features and predict a full-body human using a bounding box. It differs by 1.3 GFLOPs from Fast-PdNet [25]. Nevertheless, our detector is more accurate than this competitor. We employ fewer layers than YOLOv5n and YOLOv5s, making it more efficient without significantly degrading performance.

Furthermore, this study also investigates the model's speed implemented on low-cost devices such as central processing units (CPUs). It uses a core i7 6600T CPU with 12GB RAM without a graphic accelerator. This device is a representation of the hardware of the surveillance system, which is indicative of an inexpensive device. This test ran the detector on a pre-recorded video and then calculated the frames per second (FPS) converted from the time processing model. The proposed model achieved 26.46 FPS, which is faster than YOLOv5n. Even our human detector is three times faster than YOLOv5s. The speed of our model reaches real-time speed on low-cost devices and is feasible to implement in video surveillance. Although Fast-PdNet is faster than the proposed detector, this competitor is inaccurate and generates many false positives when assigned to find small-scale humans.

V. Conclusion

In this paper, a lightweight human detector using a superficial network is offered to localize the full-body person without compromising on execution speed. The need for a person detector for video surveillance that can run in real-time on low-cost devices delivers a lightweight architecture design. The proposed network develops YOLOv5n by increasing its efficiency. Limiting the maximum number of channels on the backbone decreases the usage of trained parameters of the deep learning network, significantly increasing the data processing speed. The path aggregate network helps to interconnect features of different frequencies. This module enhances the various levels of information relations. On the other hand, multi-layer detection employs anchors with distinct dimensions to predict humans of various sizes based on the assignment scale. As a result, the evaluation detector on MS COCO, VOC 2007, and VOC 2012 shows that the proposed network achieves an average precision of 45.20%, 86.50, and 86.90, respectively, which is competitive with previous methods and comparable detectors. Inference time testing demonstrates the proposed model can operate in real-time at 26.45 FPS, which is faster than YOLOv5s and YOLOv5n. Future work may develop the detection of human elements, such as the hand, foot, body, and head.

References

[1] S. Zhu, R. G. Guendel, A. Yarovoy, and F. Fioranelli, "Continuous Human Activity Recognition with Distributed Radar Sensor Networks and CNN-RNN Architectures," *IEEE Transactions on Geoscience and*

Remote Sensing, vol. 60, 2022, doi: 10.1109/TGRS.2022.3189746.

[2] Y. Tang, X. Yang, X. Jiang, N. Wang, and X. Gao, "Dually Distribution Pulling Network for Cross-Resolution Person Reidentification," *IEEE Trans Cybern*, vol. 52, no. 11, pp. 12016–12027, Nov. 2022, doi: 10.1109/TCYB.2021.3077500.

[3] C. Cui and R. Xu, "Multiple Machine Learning Algorithms for Human Smoking Behavior Detection," in *Proceedings - 2022 International Conference on Machine Learning and Intelligent Systems Engineering, MLISE 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 240–244. doi: 10.1109/MLISE57402.2022.00054.

[4] T. Zhou and Y. Liu, "Long-Term Person Tracking for Unmanned Aerial Vehicle Based on Human-Machine Collaboration," *IEEE Access*, vol. 9, pp. 161181–161193, 2021, doi: 10.1109/ACCESS.2021.3132077.

[5] Q. Bai, J. Xin, M. Yan, Y. Wang, E. Li, and S. Zhao, "An optimized mask-guided mobile pedestrian detection network with millisecond scale," in *Proceedings - 2020 Chinese Automation Congress, CAC 2020*, Institute of Electrical and Electronics Engineers Inc., Nov. 2020, pp. 4975–4980. doi: 10.1109/CAC51589.2020.9326617.

[6] X. Li, X. Luo, H. Hao, F. Chen, and M. Li, "Pedestrian detection method based on multi-scale fusion inception-SSD model," 2020, pp. 1549–1553. doi: 10.1109/ITAIC49862.2020.9338909.

[7] F. B. Setiawan, C. B. Adipradana, and L. H. Pratomo, "Fruit Ripeness Classification System Using Convolutional Neural Network (CNN) Method," *PROtek : Jurnal Ilmiah Teknik Elektro*, vol. 10, no. 1, p. 46, Jan. 2023, doi: 10.33387/protk.v10i1.5549.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Dec. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[9] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Sep. 2014, [Online]. Available: http://arxiv.org/abs/1409.1556

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks." [Online]. Available: http://code.google.com/p/cuda-convnet/

[11] X. Zhang, S. Cao, and C. Chen, "Scale-Aware Hierarchical Detection Network for Pedestrian Detection," *IEEE Access*, vol. 8, pp. 94429–94439, 2020, doi: 10.1109/ACCESS.2020.2995321.

[12] M. D. Putro, L. Kurnianggoro, and K. H. Jo, "High Performance and Efficient Real-Time Face Detector on Central Processing Unit Based on Convolutional Neural Network," *IEEE Trans Industr Inform*, vol. 17, no. 7, pp. 4449–4457, Jul. 2021, doi: 10.1109/TII.2020.3022501.

[13] D. Chen, S. Xia, and Y. Zhou, "Pedestrian detection via contour fragments," in *Chinese Control Conference, CCC*, IEEE Computer Society, Aug. 2016, pp. 4054–4060. doi: 10.1109/ChiCC.2016.7553986.

[14] C. B. Murthy and M. Farukh Hashmi, "Real Time Pedestrian Detection Using Robust Enhanced Tiny-YOLOv3," in *2020 IEEE 17th India Council International Conference, INDICON 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020. doi: 10.1109/INDICON49873.2020.9342082.

[15] J. An, M. D. Putro, A. Priadana, Y. Lee, J. Kim, and K. Jo, "YOLOv5 with Dual Attention Network for Object Detection on Drone," in *2023 International Workshop on Intelligent Systems (IWIS)*, IEEE, Aug. 2023, pp. 1–6. doi: 10.1109/IWIS58789.2023.10284592.

[16] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," Jul. 2022, [Online]. Available: http://arxiv.org/abs/2207.02696

[17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection."

[18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," Jun. 2015, [Online]. Available: http://arxiv.org/abs/1506.01497

[19] G. Jocher *et al.*, "ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements." Zenodo, Oct. 2020. doi: 10.5281/zenodo.4154370.

[20] M. and B. S. and H. J. and P. P. and R. D. and D. P. and Z. C. L. Lin Tsung-Yi and Maire, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*, T. and S. B. and T. T. Fleet David and Pajdla, Ed., Cham: Springer International Publishing, 2014, pp. 740–755.

[21] M. Everingham, L. Gool, C. K. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *Int. J. Comput. Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.

[22] M. Xu, Z. Wang, X. Liu, L. Ma, and A. Shehzad, "An Efficient Pedestrian Detection for Realtime Surveillance Systems Based on Modified YOLOv3," *IEEE Journal of Radio Frequency Identification*, vol. 6, pp. 972–976, 2022, doi: 10.1109/JRFID.2022.3212907.

[23] H. H. Nguyen, T. N. Ta, N. C. Nguyen, V. T. Bui, H. M. Pham, and D. M. Nguyen, "YOLO Based Real-Time Human Detection for Smart Video Surveillance at the Edge," in *ICCE 2020 - 2020 IEEE 8th International Conference on Communications and Electronics*, Institute of Electrical and Electronics Engineers Inc., Jan. 2021, pp. 439–444. doi: 10.1109/ICCE48956.2021.9352144.

[24] M. D. Putro, D. L. Nguyen, and K. H. Jo, "A CPU-based Pedestrian Detector using Deep Learning for Intelligent Surveillance Systems," in *Proceedings of the IEEE International Conference on Industrial Technology*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICIT48603.2022.10002735.

[25] M. D. Putro, D. L. Nguyen, A. Priadana, and K. H. Jo, "Fast Person Detector with Efficient Multi-level Contextual Block for Supporting Assistive Robot," in *Proceedings - 2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems, ICPS 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICPS51978.2022.9816965.

[26] W. Y. Hsu and W. Y. Lin, "Adaptive Fusion of Multi-Scale YOLO for Pedestrian Detection," *IEEE Access*, vol. 9, pp. 110063–110073, 2021, doi: 10.1109/ACCESS.2021.3102600.