IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 13, No. 2, June 2024, pp. 1838~1844
ISSN: 2252-8938, DOI: 10.11591/ijai.v13.i2.pp1838-1844

A lightweight you only look once for real-time dangerous weapons detection

Aicha Khalfaoui[1], Abdelmajid Badri[2], Ilham El Mourabit[3]
Laboratory of Electronics, Energy, Automatic & Information Processing, Faculty of Sciences and Techniques Mohammedia, University Hassan II Casablanca, Morocco

Article Info

Article history:

Received May 23, 2023
Revised Nov 17, 2023
Accepted Dec 14, 2023

Keywords:

Deep learning
Embedded devices
GhostNet
Weapons detection
YOLOv5

ABSTRACT

Deep neural networks are currently employed to detect weapons, and although these techniques provide a high level of accuracy, it still suffers from large weight parameters and a slow inference speed. When considering real-world applications like weapon detection, these methods are frequently unsuitable for deployment on embedded devices due to their large number of parameters and poor efficiency. The most recent object detection technique, which falls under the YOLOv5 (You Only Look Once version 5) family, is commonly used for detecting weapons. However, it faces some difficulties such as high computational parameters and an unfavorable detection rate. to solve these shortcomings. an enhanced lightweight Yolov5s approach is suggested. Which consists of a combination of YOLOv5 and GhostNet modules. To evaluate the efficacy of the suggested technique, a set of experiments was performed on the Sohas weapon dataset., which is commonly used as a reference dataset in the field. Compared to the original YOLOv5, the results indicate a slight increase in the proposed model's mean Average Precision (mAP). Furthermore, there has been a reduction of 2.7 in giga floating point operations per second (GFLOPs) and weights, and the number of model parameters has decreased by 1.42.

Corresponding Author:

Aicha Khalfaoui
Laboratory of Electronics, Energy, Automatic & Information Processing
Faculty of Sciences and Techniques Mohammedia, University Hassan II Casablanca
Mohammedia, Morocco.
Email: aicha96khalfaoui@gmail.com

1. INTRODUCTION

Abnormal object detection is a crucial component of video surveillance systems in smart cities. It involves using advanced algorithms and artificial intelligence to identify and flag any objects or events deemed abnormal or potentially harmful to the surrounding environment or people. This technology is used to enhance public safety and security in various urban areas, such as streets, parking lots, and public transportation systems. The primary goal of abnormal object detection is to quickly and accurately detect any potential threats and alert the authorities or relevant stakeholders, allowing for a swift response to prevent or mitigate any harm.

Detecting dangerous weapons from surveillance video can be challenging. Carrobles *et al.* [1] show that security personnel monitoring CCTV lose focus after 20 minutes. González *et al.* [2], Research has indicated that in continuous video monitoring lasting 12 minutes, a security guard could overlook up to more than 45% of activities. Furthermore, after a period of 22 minutes of continuous surveillance, more than 95% of events could be missed. To overcome this, deep learning can be employed to automate surveillance video feed monitoring, allowing for the detection of critical events.

There are two primary categories of research related to identifying objects in images or videos, as outlined in reference [3]. One group of methods aims to detect guns and knives using traditional algorithms that do not involve deep learning, while another group aims to improve object detection accuracy through the utilization of deep learning methods. Classical algorithms primarily utilize color-based segmentation, corner detection, and appearance features for object recognition. However, the effectiveness of these algorithms is impacted by the quality of frames or images they rely on, which can be considered a limitation, making it challenging to interpret frames with occlusion or noise. Additionally, when the color segments of the foreground and background are similar, it can be difficult to interpret the results when utilizing color-based segmentation, as explained in [4]. Deep learning algorithms primarily rely on neural networks. A major benefit of employing a neural network architecture is its ability to learn feature extraction automatically during training. Additionally, training on a larger dataset enables the neural network model to recognize occluded frames.

The researches [5], [6] the authors presented a system for scanning luggage with X-rays in their respective studies that utilized various object recognition methods like sliding window, You only look once and Faster Region-based convolutional neural network (Faster R-CNN) to classify and detect objects. The system was designed to classify objects into several categories including knives and knife parts, guns and gun parts. However, the accuracy of object detection was limited under different lighting conditions and in the case of high occlusion. A technique for identifying anomalies involves utilizing timed image-based CNN to detect actions [7].

In their study, Verma and Dhillon [8] described how they used the internet movie firearm database (IMFDB) to train R-CNN algorithm with a classification head built on the visual geometry group (VGG16) structure to identify handguns. Bhatti *et al.* [9]. created a custom database for YOLOv4 training and evaluated its performance against cutting-edge methods. Their research focused on identifying specific objects such as pistols, revolvers, wallets, metal detectors, and cell phones in videos. The study compared the YOLO model with R-CNN, single shot multibox detector (SSD), and another version of YOLO. While some classification models in static mode displayed encouraging results, they were less accurate and slower in real-time situations when running on devices with limited resources. Although the models achieved high F1 scores on the initial dataset, they were not suitable for scenarios that included background objects. Kanehisa and Neto [10] investigate the implementation of the YOLO algorithm for the purpose of developing a system for detecting firearms, demonstrating its effectiveness in this specific task.

While these studies have demonstrated satisfactory outcomes, their ability to deal with challenging scenarios and resource-constrained environments is somewhat restricted. To address the issues highlighted earlier, we conducted a series of experiments and developed a specialized CNN architecture.

The YOLO detection network has gained popularity since its proposal [11]–[14], mainly due to its fast and accurate performance. Recently, the fifth version of this network has been introduced. Our study utilizes YOLOv5 and introduces a new network that significantly reduces GFLOPs and network weight while maintaining high accuracy. We also provide an explanation for the feasibility of this approach. Our experiments demonstrate that our method has faster convergence and a higher mAP compared to the original YOLOv5.

This article is structured into four key parts. The introductory section serves as the first part, followed by a section dedicated to detailing the proposed approach. The third part is centered on demonstrating the experimental results and their significance. Lastly, the fourth section offers a summary of the primary discoveries of the research and suggests future research directions.

2. METHOD

2.1. The Yolov5 network

Ultralytics proposed YOLOv5 [15], which is an enhanced version of YOLOv4 and serves as a detection model that incorporates the strengths of prior editions and other networks like CSPNet and PANet [16], [17]. As a result, it achieves a satisfactory balance between precision and speed. The structure of YOLOv5 is designed to be more efficient than its previous versions, despite its smaller size. Figure 1 shows a graphic illustration of its fundamental structure.

The YOLOv5 algorithm integrates multiple scale prediction and merges the feature pyramid network (FPN) and path aggregation network (PANet) networks. The FPN network transfers profound semantic characteristics to less profound layers, which in turn enhances semantic expression at various scales. Similarly, PANet network transmits the shallow layer's localization data to the deep layer, thereby improving localization ability at different scales. By combining these two networks, YOLOv5 can boost both semantic representation and localization skills on multiple scales. Its architecture primarily consists of C3, Conv, and spatial pyramid pooling features (SPPF) modules. Moreover, the network width and depth can be regulated through YOLOv5's utilization of depth and width multiples.

In broad terms, YOLOv5 has enhanced its performance by incorporating the following four elements: i) Input improvements such as adaptive image scaling, adaptive anchor box computation, and mosaic data augmentation; ii) Improved backbone architecture utilizing cross stage partial network (CSPNet) and Focus module. iii) Enhanced neck architecture with FPN and PANet networks. iv) Replacing the intersection over union (IoU) with the complete intersection over union (CIoU) for loss function. These improvements have contributed to YOLOv5's overall performance enhancements.

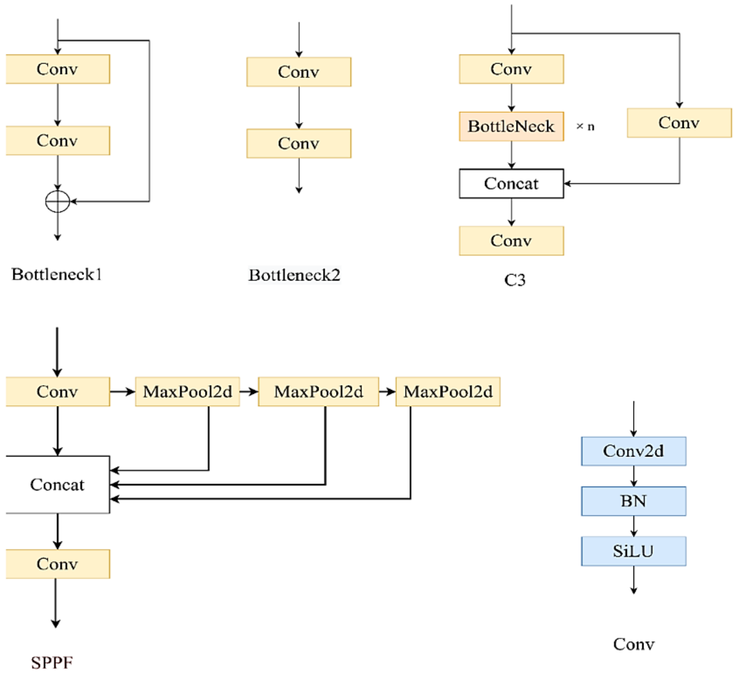

Figure 1. The principal component of YOLOv5s [18]

2.2. GhostNet module

Neural networks usually require a significant and extensive amount of parameters , particularly in initial fully connected layers, to better-fit datasets. The emergence of convolutional neural networks has enabled the application of filters to minimize the number of parameters required. In creating a network designed to complete detection tasks, numerous feature maps containing hundreds of channels are typically required, resulting in a large model. Model compression involves reducing a neural network's total number of parameters, making it easier to deploy on embedded devices.

Some techniques have been developed by researchers to decrease the size of models. For example, ShuffleNet [19], channel shuffle is used to improve the flow of information between channel groups. Xception [20], in contrast, uses a split convolution operation and a more effective feature fusion approach to optimize the parameters of the model. MobileNets [21], leverage a set of depth-wise separable convolutions to achieve superior performance with fewer parameters. Finally, SqueezeNet [22], substitutes a 1x1 convolution kernel for a 3x3 kernel and decreases the number of input channels after compression, the 480 Mb original size of the model is reduced to 4.8 Mb. Han et al. came up with a novel technique called GhostNet to overcome the issue of excessive parameters leading to high resource consumption and to facilitate deploying neural networks on embedded devices more easily. The GhostNet method is designed to produce a larger number of feature maps using more affordable operations [23], [24].

Figure 2 shows that Conv and C3 modules have been replaced by Ghostconv and C3Ghost, respectively. There are an equal number of input channels and output channels in both modules, where C1 stands for the input channels and C2 for the output channels. GhostConv is made up of two Conv modules with the number of hidden channels being half of the input and output channels of the module.

3. RESULTS AND DISCUSSION

3.1. Dataset description

In this work, the Sohas dataset is used [25], The dataset contains 4,014 images featuring weapons, categorized by the type of handheld weapon object used, namely pistols and knives. The model was trained on 70% of the images, while 20% were allocated for validation, and the remaining 10% were utilized to test the model. The two categories were approximately balanced. As the Sohas dataset lacks blurry image samples typically found in surveillance video frames, data augmentation is performed including horizontal flip, rotation, and blur. To create a more diverse and extensive dataset. This can aid the model in generalizing better with images obtained from surveillance videos.

3.2. Training protocol

The experiment was carried out on a Linux operating system, utilizing an NVIDIA Tesla T4 graphic card, and incorporating CUDA 11.1, PyTorch 1.12.1, and Python 3.7 software environment. As the optimizer, we employed the stochastic gradient descent (SGD) algorithm, while keeping the rest of the original YOLOv5 hyperparameters intact. To ensure thorough training, the total number of epochs for the experiment was set to 100, allowing the model to converge and achieve optimal results.

3.3. Results

There exist five primary versions of yolov5, which are YOLOv5x, YOLOv5l, YOLOv5m, YOLOv5s and finally YOLOv5n. The differences between the yolov5's versions relate to their model size, depth, and the number of parameters used. The results of our tests are presented in Table 1, which involved incorporating GhostNet into different locations and comparing model performance using the metrics of weights, GFLOPs, and the number of parameters. The results demonstrated that YOLOv5n and YOLOv5s are the two versions with smaller model sizes than the other official versions. This makes them more memory-efficient, this can offer a benefit in environments with limited resources, such as embedded systems or mobile devices.

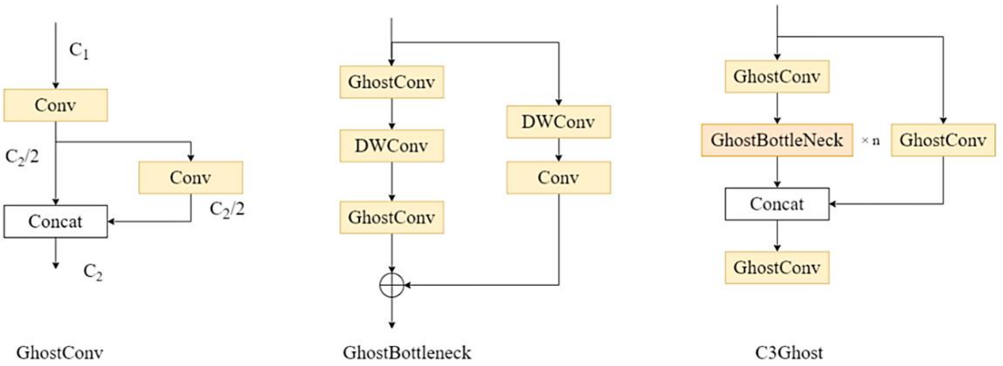

Figure 2. Primary architecture of yolov5 incorporating the Ghostnet module [18]

Table 1. Model’s parameters

Models	Params (Millions)	Weights (Mb)	GFLOPs
YOLOv5n	1.76	3.9	4.1
YOLOv5n-GHOST-ALL	**0.94**	**2.3**	**2.3**
YOLOv5n-GHOST-BACKBONE	1.29	3	3
YOLOv5n-GHOST-HEAD	1.42	3.2	3.6
YOLOv5s	7.02	14.4	15.9
YOLOv5s-GHOST-ALL	**3.7**	**7.8**	**8.2**
YOLOv5s-GHOST-BACKBONE	5.1	10.6	10.5
YOLOv5s-GHOST-HEAD	5.6	11.7	13.2
YOLOv5m	20.89	40.3	48.1
YOLOv5m-GHOST-ALL	**8.55**	**16.8**	**18.4**
YOLOv5m-GHOST- BACKBONE	15.5	30.1	38
YOLOv5m-GHOST-HEAD	13.89	27	28.5
YOLOv5l	46.17	88.6	108
YOLOv5l-GHOST-ALL	**15.62**	**30.5**	**33.3**
YOLOv5l-GHOST-BACKBONE	32.7	63.1	82
YOLOv5l-GHOST-HEAD	29.02	56	59.3
YOLOv5x	86.25	165	204.3
YOLOv5x-GHOST-ALL	**25.09**	**48.7**	**53.3**
YOLOv5x-GHOST-BACKBONE	59.2	113	151.3
YOLOv5x-GHOST-HEAD	52.1	100	106.4

Table 2 provides a comparison of the latest detection models based on the (mAP) and detection time. yolov5s with GhostNet integrated achieve better results in terms of mAP. While yolov5n with GhostNet integrated represent the smallest detection time. Yolov5s-Head-Ghost shows suitable trade-off between precision and speed. which makes it an effective choice for embedded and real time applications

The training loss curve shows that the inclusion of GhostNet in the head part of the model leads to a faster rate of convergence, which is likely due to the model's compact design and reduced number of parameters. Figure 3 demonstrate, that the model's loss is high in the beginning, as it makes inaccurate predictions during training. However, through iterative optimization, the loss decreases as the model improves its predictions. As the loss curve converges towards zero, it indicates the model is effectively capturing data patterns and generalizing well to unseen data. The convergence also suggests the model is not overfitting, performing well on new samples. Both training and validation loss decrease together, which confirms the model is learning meaningful representations and not overfitting. and, Table 2 reveals a higher mAp with GhostNet. This indicates that GhostNet is an efficient addition to the model. Figure 4 represents some test example detection.

Table 2. Evaluation metrics of each model

Models	mAP@0.5	Detection-Time(ms)
YOLOv3	0.98	14.6
Tiny-YOLOv3	0.959	7.69
YOLOv5n	0.976	**5.9**
YOLOv5n-GHOST-ALL	0.981	6.8
YOLOv5n-GHOST-BACKBONE	0.98	6.5
YOLOv5n-GHOST-HEAD	0.984	6.4
YOLOv5s	0.991	11.5
YOLOv5s-GHOST-ALL	0.993	9.9
YOLOv5s-GHOST-BACKBONE	0.992	10.5
YOLOv5s-GHOST-HEAD	**0.994**	10.9

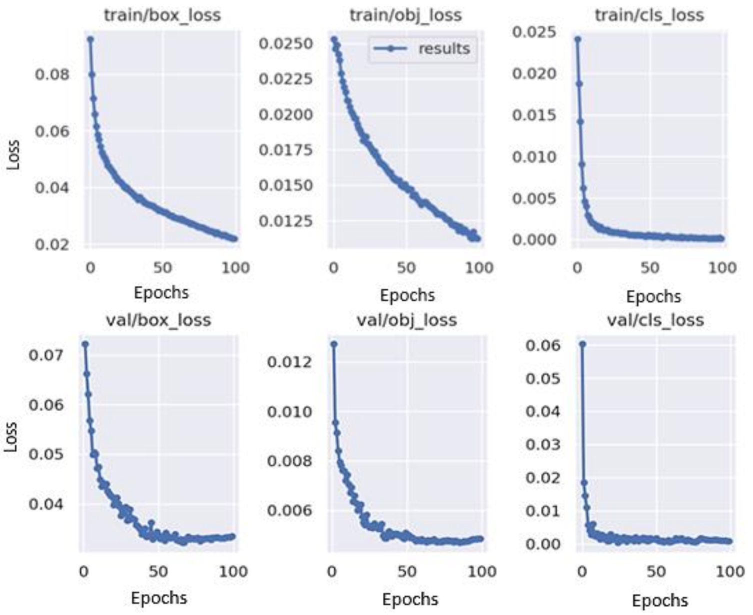

Figure 3. loss curves of Yolov5s-Head-Ghost

3.2. Discussion

This study evaluates the effectiveness of GhostNet in several locations within the YOLOv5 model. and shows that using GhostNet can achieve similar or better accuracy rate with fewer parameters and less computation, making it suitable for embedding devices. The Yolov5s-Head-Ghost performs most efficiently compared to other models, achieving better mAP with fewer weights and GFLOPs. The smaller model size of YOLOv5s means that it can perform inference faster than the larger models. This shows that GhostNet can effectively prune deep neural networks without compromising weapons detection performance. The pruning effect depends on the network depth, GhostNet is a module that can be easily applied to other classical models to reduce computation, but the tradeoff between accuracy and model size should still be considered for different memory situations.

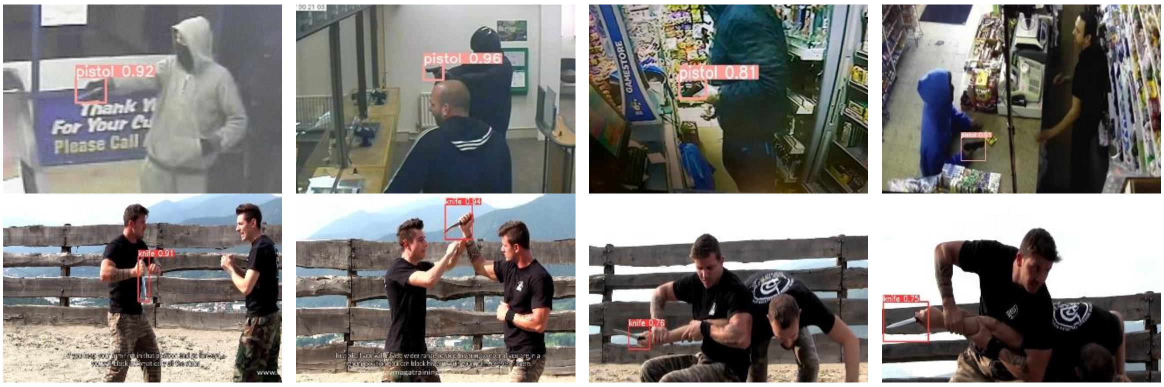

Figure 4. Sample detection results for the test set

4. CONCLUSION

In this research paper, GhostNet is presented as a potential solution to decrease the amount of computation required by deep neural networks. and create more efficient neural architectures that are suitable for embedding devices. The GhostNet module is shown to be easily implemented to yolov5 models while maintaining similar performance. This was demonstrated by comparing metrics such as mAP, FPS, and loss curves. By only integrating GhostNet in the head part, the proposed method is shown to improve mAP and reduce the loss. It is suggested that GhostNet could potentially replace ordinary convolution in the future, as it can achieve similar effects through cheaper operations. In future work, we aim to train the model on other larger datasets and further improve the detection time for a mobile application requiring a lightweight, accurate, fast detection model.

REFERENCES

[1] M. M. F.-Carrobles, O. Deniz, and F. Maroto, “Gun and knife detection based on faster R-CNN for video surveillance,” in *Iberian Conference on Pattern Recognition and Image Analysis*, Cham: Springer, 2019, pp. 441–452. doi: 10.1007/978-3-030-31321-0_38.

[2] J. L. S. González, C. Zaccaro, J. A. Á.-García, L. M. S. Morillo, and F. S. Caparrini, “Real-time gun detection in CCTV: An open problem,” *Neural Networks*, vol. 132, pp. 297–308, Dec. 2020. doi: 10.1016/j.neunet.2020.09.013.

[3] K. U. Sharma and N. V. Thakur, “A review and an approach for object detection in images,” *International Journal of Computational Vision and Robotics*, vol. 7, no. 1–2, pp. 196–237, 2017. doi: 10.1504/IJCVR.2017.081234.

[4] R. K. Tiwari and G. K. Verma, “A computer vision based framework for visual gun detection using harris interest point detector,” in *Procedia Computer Science*, 2015, vol. 54, pp. 703–712. doi: 10.1016/j.procs.2015.06.083.

[5] M. E. Kundegorski, S. Akcay, M. Devereux, A. Mouton, and T. P. Breckon, “On using feature descriptors as visual words for object detection within X-ray baggage security screening,” 2016. doi: 10.1049/ic.2016.0080.

[6] J. Zhang, W. Xing, M. Xing, and G. Sun, “Terahertz image detection with the improved faster region-based convolutional neural network,” *Sensors*, vol. 18, no. 7, p. 2327, Jul. 2018. doi: 10.3390/s18072327.

[7] A. M. Atto, A. Benoit, and P. Lambert, “Timed-image based deep learning for action recognition in video sequences,” *Pattern Recognition*, vol. 104, p. 107353, Aug. 2020. doi: 10.1016/j.patcog.2020.107353.

[8] G. K. Verma and A. Dhillon, “A handheld gun detection using faster R-CNN deep learning,” in *Proceedings of the 7th International Conference on Computer and Communication Technology*, Nov. 2017, pp. 84–88. doi: 10.1145/3154979.3154988.

[9] M. T. Bhatti, M. G. Khan, M. Aslam, and M. J. Fiaz, “Weapon detection in real-time CCTV videos using deep learning,” *IEEE Access*, vol. 9, pp. 34366–34382, 2021. doi: 10.1109/ACCESS.2021.3059170.

[10] R. Kanehisa and A. Neto, “Firearm detection using convolutional neural networks,” in *Proceedings of the 11th International Conference on Agents and Artificial Intelligence*, 2019, pp. 707–714. doi: 10.5220/0007397707070714.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: unified, real-time object detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[12] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017, pp. 6517–6525. doi: 10.1109/CVPR.2017.690.

[13] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” *arXiv*, 2018.

[14] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” *arXiv*, 2020.

[15] “ultralytics/yolov5,” *Ultralytics*. [Online]. Available: https://github.com/ultralytics/yolov5

[16] C. Y. Wang, H. Y. Mark Liao, Y. H. Wu, P. Y. Chen, J. W. Hsieh, and I. H. Yeh, “CSPNet: A new backbone that can enhance learning capability of CNN,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 1571–1580. doi: 10.1109/CVPRW50498.2020.00203.

[17] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, “PANet: Few-shot image semantic segmentation with prototype alignment,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9196–9205. doi: 10.1109/ICCV.2019.00929.

[18] Y. Zhang, W. Cai, S. Fan, R. Song, and J. Jin, “Object detection based on YOLOv5 and GhostNet for Orchard Pests,” *Information*, vol. 13, no. 11, p. 548, Nov. 2022. doi: 10.3390/info13110548.

[19] X. Zhang, X. Zhou, M. Lin, and J. Sun, “ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 6848–6856. doi: 10.1109/CVPR.2018.00716.

[20] F. Chollet, “Xception: deep learning with depthwise separable convolutions,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 1800–1807. doi: 10.1109/CVPR.2017.195.

[21] H. A. Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, “MobileNets: efficient convolutional neural networks for mobile vision applications,” *arXiv Computer Vision and Pattern Recognition*, vol. 14, no. 2, pp. 53–57, 2009.

[22] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size,” *arXiv*, 2016.

[23] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, “GhostNet: More features from cheap operations,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1577–1586. doi: 10.1109/CVPR42600.2020.00165.

[24] K. Han *et al.*, “GhostNets on heterogeneous devices via cheap operations,” *International Journal of Computer Vision*, vol. 130, no. 4, pp. 1050–1069, Apr. 2022. doi: 10.1007/s11263-022-01575-y.

[25] F. Pérez-Hernández, S. Tabik, A. Lamas, R. Olmos, H. Fujita, and F. Herrera, “Object detection binary classifiers methodology based on deep learning to identify small objects handled similarly: Application in video surveillance,” *Knowledge-Based Systems*, vol. 194, p. 105590, Apr. 2020. doi: 10.1016/j.knosys.2020.105590.

BIOGRAPHIES OF AUTHORS

Aicha Khalfaoui was born in Errachidia, Morocco on February 28, 1996. She received her M.Sc. degree in industrial computing and instrumentation engineering from the Faculty of Sciences and Techniques of Errachidia. She is currently a Ph.D. student in the Laboratory of Electronics, Energy, Automatic, and Information Processing (EEA&TI) Hassan II University, Mohammedia-Casablanca, Morocco. Her work studies and interests focus on improving embedded real-time vision systems using Deep Learning techniques. She can be contacted at email: aicha96khalfaoui@gmail.com.

Abdelmajid Badri is holder of a doctorate in Electronics and Image Processing in 1992 at the University of Poitiers-France. In 1996, he obtained the diploma of the authorization to Manage Researches (HDR) at the University of Poitiers-France, image processing. He was a University Professor (PES-C) at the University Hassan II Mohammedia-Casablanca Morocco (FSTM). In 2018, he became director of the superior school of technology of Casablanca Morocco (EST). He is a member of the laboratory EEA&TI (Electronics, Energy, Automatic and Information Processing) which he managed since 1996. He managed several doctoral theses. He is a co author of several national and international publications. He can be contacted at email: Abdelmajid_badri@yahoo.fr.

Ilham El Mourabit is an Assistant Professor and Researcher, holder of a doctoral degree in Electronics and telecommunication systems from Hassan II university. She received her M.Sc. degree in Electronic and Automatic Systems Engineering (Telecommunication and Information Technologies specialty) from the Faculty of Sciences and Technology of Mohammedia, Morocco. Currently working as an assistant professor at the FSTM. She is a member of the EEA&TI Laboratory (Electronics, Energy, Automatic and Information Processing), at Hassan II University Casablanca. Her main research areas are geolocation technologies in wireless networks, image processing, computer vision, digital signal processing, and vehicular communications. She can be contacted at email: elmourabit.ilham@gmail.com.