Journal of Computer and Communications, 2022, 10, 129-139
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Research on the Application of Helmet Detection Based on YOLOv4

Yongze Ji[1], Yu Cao[1], Xu Cheng[2], Qiong Zhang[1]

[1]School of Information and Control Engineering, Liaoning Petrochemical University, Fushun, China
[2]Shenyang Agricultural University, Shenyang, China
Email: yucao_lnshu@163.com

How to cite this paper: Ji, Y.Z., Cao, Y., Cheng, X. and Zhang, Q. (2022) Research on the Application of Helmet Detection Based on YOLOv4. *Journal of Computer and Communications*, **10**, 129-139.
https://doi.org/10.4236/jcc.2022.108009

Received: July 15, 2022
Accepted: August 28, 2022
Published: August 31, 2022

Abstract

Helmets are one of the important measures to ensure the safety of construction workers. Because the harm caused by not wearing safety helmets as required is great, the wearing of safety helmets has also attracted more and more people's attention. At present, the main method of helmet detection is the YOLO series of algorithms. They often only focus on detection accuracy, ignoring the actual situation during deployment, that is, a balance between accuracy and speed is required. Therefore, this paper proposes a helmet detection application based on YOLOv4 algorithm, and combined with the MobileNet network, it has achieved good results in terms of detection accuracy and speed. Through transfer learning and tuning parameters, the mAP and FPS values detected in this paper on the public safety helmet datasets are 94.47% and 27.36%, which exceed the research work of some similar papers. This paper also combines YOLOv4 and MobileNetv3 networks to propose a mobileNet-based YOLOv4 helmet detection application. Its mAP and FPS values are 91.47% and 42.58%, respectively, which meet the accuracy and real-time requirements of current hardware deployment.

Keywords

YOLOv4, Helmet, Target Detection, MobileNet

1. Introduction

With the continuous development of computer vision, the application of computer vision is getting closer and closer to us, and gradually integrated into all aspects of our life. Safety helmets are one of the important safeguards to protect the lives of construction workers in the construction industry, production plants and other areas where risks exist, so this paper uses the target detection method to detect the wearing of safety helmets of workers in construction sites and remind them to wear the helmets in real time, which is important to protect their lives.

At present, target detection algorithms can be divided into traditional target detection algorithms and deep learning target detection algorithms. In this paper, deep learning target detection algorithm is the base line to explore its development and application in target detection. Deep learning target detection algorithm is further divided into two-stage target detection algorithm and one-stage target detection algorithm, due to meet the real-time requirements of helmet detection in construction site, this paper mainly studies the one-stage target detection algorithm. The most typical algorithms for one-stage target detection are SSD, YOLO, RetinaNet, CenterNet, EfficientDet, etc. Among them, the YOLO algorithm has received the attention of many researchers due to its superior performance in meeting the conditions of accuracy and real-time. This paper is no exception and explores the performance of the algorithm of the YOLO series as the main route for its performance in helmet detection applications.

Safety helmet detection is one of the research directions of target detection and is of great interest to research enthusiasts. Weihong Wu [1] proposed an application scenario of helmet detection in security surveillance by changing the backbone network of YOLOv3 to ResNet and added attention mechanisms, thus proposed an improved YOLOv3-B detection algorithm that improved the accuracy of helmet detection. Yuxin Huang [2] *et al.* proposed a portable and reliable helmet detection system by combining YOLOv2 with an embedded device. Shuai Li [3] *et al.* improved YOLOv4 by image enhancement techniques, redesigned the anchor size by K-means, and added dilated convolution and label smoothing techniques to improve its effectiveness in detecting helmets both in terms of small targets and speed. Zhao Rui [4] *et al.* proposed an improved YOLOv5s algorithm by replacing the slicing (Focus) structure of the YOLOv5 backbone network with DenseBlock and added the SE attention mechanism to the neck network, thus greatly improved the accuracy of target detection. Yufang Jin [5] *et al.* increased the detection effect of small targets by adding the output of feature layer of 128 × 128 to the output of feature layer of YOLOv4, additionally, enhanced the feature reuse and improved the detection effect of helmet by addition to the idea of dense connection. Chenglong Wang [6] *et al.* proposed a helmet detection method which is different from all of the above, and their algorithm used a combination of facial features and neural networks, combined with VGG networks for helmet detection to provide safety to construction workers. Sun [7] *et al.* enhanced the efficiency of small target detection by adding a self-attentive mechanism to the framework of Faster R-CNN, and the framework was further focused on small targets by anchors complementary enhancement, which achieved good results in helmet detection. In addition to using the above target detection framework, helmet detection can also be performed using OpenCV, and Zhao Zhen [8] implemented safety helmet detection based on OpenCV, which also achieved the effect of reminding construction workers to wear safety helmets in practice and reduced unsafe factors.

This paper proposes to combine YOLOv4 target detection algorithm to detect helmet wearing in construction sites in real time, so as to remind construction workers to wear helmets in time. In order to improve the accuracy and real-time of helmet detection, this paper improves the accuracy of helmet detection through extensive experiments, combining the currently available target detection algorithms, adjusting parameters and migration learning. The mAP and FPS values of YOLOv4 helmet detection proposed in this paper are 94.47% and 27.36% respectively on the publicly available helmet detection datasets, and its detection effect is relatively excellent. In addition, in order to further deploy the model to hardware devices with limited resources and meet the requirement of real-time helmet detection, this paper proposes the application scenario of replacing the YOLOv4 backbone network with MobileNet for helmet detection, which greatly increases the possibility of deploying. The mAP and FPS values on the publicly available helmet detection datasets are 91.47% and 42.58% respectively, which also meet the accuracy and real-time requirements of application deployment and lower the deployment threshold.

2. Theory of Target Detection

2.1. YOLOv3

YOLOv3 [9] is the latest version proposed by Joseph Redmon, following YOLOv1 and YOLOv2. YOLOv3 uses DarkNet53 as the backbone network and Leaky ReLU activation function to construct a network structure, which is similar to feature pyramid networks, to enhance feature extraction, and uses YOLO Head to obtain the final prediction results. YOLOv3 divides the detection targets in the images into three scales: large, medium, and small, the images are divided into 13 × 13, 26 × 26, and 52 × 52, and each feature point corresponds to three prior anchors. The prior anchors of YOLOv3 need to be found before the training datasets using K-means, because each feature point uses three prior anchors, so we need nine prior anchors before train the datasets by K-means. YOLOv3 is the most important version in the development of the YOLO series, and many subsequent versions of YOLO are based on the improved version of YOLOv3 (**Figure 1**).

2.2. YOLOv4

YOLOv4 [10] is not a completely new version, but more precisely YOLOv4 is a series of tricks added to YOLOv3, which is a collection of tricks to increase the accuracy of target detection. YOLOv4 changes the backbone network of YOLOv3 to CSPDarknet53, introduces two characteristic pyramid structures, SPP and PANet, and uses the Mish activation function. Through these improvements, the version of YOLOv4 improves some performance on the basis of YOLOv3. By re-clustering the datasets anchors, the clustering results are obtained as shown

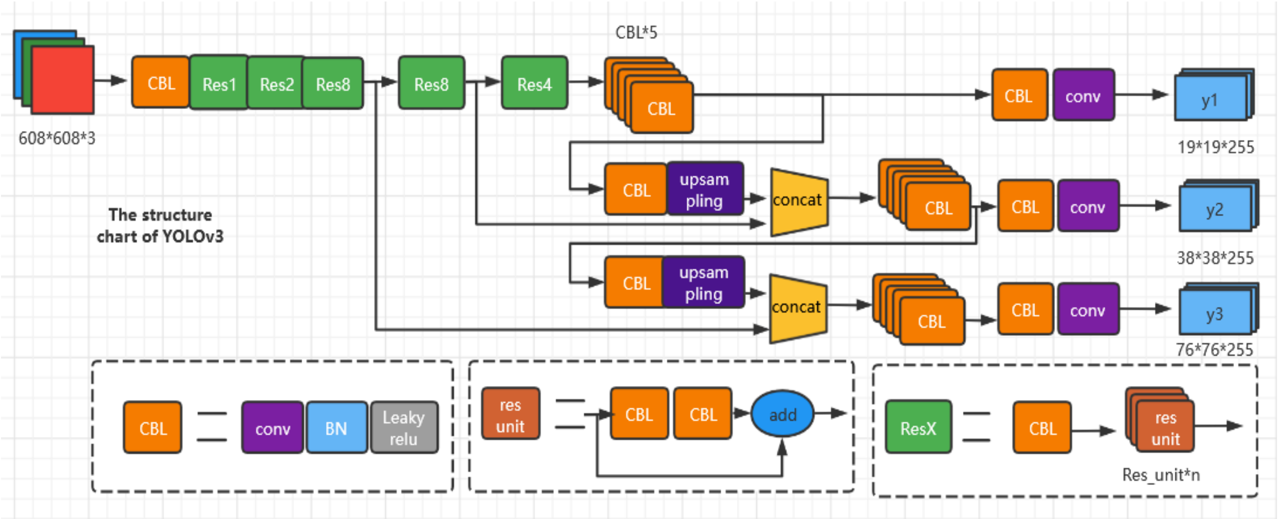

Figure 1. The structure chart of YOLOv3.

in **Table 1** below. In terms of training, YOLOv4 uses Mosaic data enhancement, Label Smoothing, CIOU and cosine annealing decay learning rate which are small tricks to make YOLOv4 better performance in target detection. The Mish activation function is calculated as follows (**Figure 2**)

$$f(x) = x * \tanh(\ln(1 + e^x)) \tag{1}$$

2.3. YOLOv5

YOLOv5 is another improved version of the YOLO series, the official website does not give a definitive paper, but the code is open source, YOLOv5 has a total of 5 versions, there are Yolov5n, Yolov5s, Yolov5m, Yolov5l, Yolov5x. Similar to YOLOv4, it also adopts the CSP structure, and the neck part adopts the FPN + PAN structure and uses the newer focus technology. YOLOv5 introduces the adaptive anchors calculation, so that the object detection algorithm will automatically calculate the size of the anchors without using k-means to generate the anchors before training, which reduces the complexity of the object detection algorithm to some extent. YOLOv5 uses GIOU Loss as the loss function of bounding box, which makes the target box regression more stable.

2.4. YOLOX

YOLOX [11] is an improved version of the latest YOLO series proposed by MEGVII in 2021 years, with six versions as Yolox-nano, Yolox-tiny, Yolox-s, Yolox-s, Yolox-l and Yolox-x, covering all ranges from small models to large models. Compared with the previous version of YOLO series, YOLOX introduces some new tricks, such as Decoupled head, Anchor-free, Multi positives, SimOTA, etc., and uses a combination of Mosaic and MixUp data enhancement in training, which makes its performance on the COCO dataset superior to other target detection algorithms.

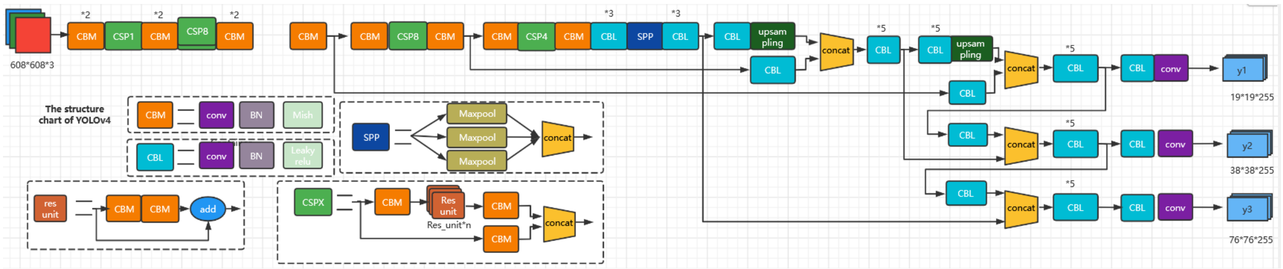

Figure 2. The structure chart of YOLOv4.

Table 1. K-means result.

feature map	13'13	26'26	52'52
receptive field	big	middle	small
anchor box	42'72	67'115	125'215

2.5. MobileNet Series

While pursuing the accuracy of the model, some researchers put the target on the balance of detection accuracy and speed, they start to focus their research on lightweight networks, among which MobileNet is one of the many lightweight networks that perform quite well. So there are three versions of MobileNet. MobileNetv1 [12] proposes separable convolutional neural network, which divides a standard convolution into a depthwise convolution and a pointwise convolution, thus greatly reducing the number of parameters and the computation of the network, making the number of parameters and the accuracy of the network reach a good balance. The core of MobileNetv2 [13] is the inverted residual block, which reduces the number of parameters and increases the accuracy of the designed network. MobileNetv3 [14] adds an attention mechanism (SE module) and is designed to utilize the network architecture search [15] (NAS) algorithm to further improve the performance of the network.

2.6. Metrics for Model Evaluation

The main metrics for the evaluation of target detection are accuracy (mean average precision, mAP) and speed (frames per second, FPS). Two important metrics in the calculation of accuracy are precision and recall. The precision is the proportion of the right part of the prediction to the prediction result, which is calculated as

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

The recall is the proportion of the correct prediction to the true sample and is calculated as

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

where TP is true positive samples, *i.e.* positive samples are correctly classified as positive samples, FP is false positive samples, *i.e.* negative samples are incorrectly classified as positive samples, and FN is false negative samples, *i.e.* positive samples are incorrectly classified as negative samples. It is obvious that the performance of a target detection algorithm requires the use of two parameters, precision and recall. Subsequently, for simplicity, the AP (average precision) metric is introduced, and the area under the PR curve is the AP value, using precision and recall as the vertical/horizontal axes of the Cartesian coordinate system, and plotting the PR curve according to the different recall values and their corresponding precision values. As for the calculation of mAP (mean average precision), in fact, the target detection cannot be just one class of targets, and we get the mAP value by taking the average of the AP values of each class.

Another metric we commonly use to calculate the accuracy of target detection is IOU (Intersection over Union), which is a measure of the accuracy of detecting the corresponding object in given data sets. The IOU represents the intersection rate or overlap between the candidate bound and the ground bound, that is, the ratio of their intersection to their concurrence. The higher the correlation, the larger the value. In the most desired case they are the complete overlap, *i.e.* the ratio is 1, which is calculated as

$$\text{IOU} = \frac{A \cap B}{A \cup B} \tag{4}$$

where A is the candidate box and B is the original marked box. yolov4 used is CIOU and the calculation formula is as follows

$$\text{CIOU} = \text{IOU} - \frac{\rho^2(b, b^{gt})}{c^2} - \alpha\upsilon \tag{5}$$

where $\rho^2(b, b^{gt})$ represents the Euclidean distance between the center points of the prediction anchors and the real anchors, b represents the diagonal distance of the smallest closed area that can contain both the prediction anchors and the real anchors, and are calculated as follows

$$\alpha = \frac{\upsilon}{1 - \text{IOU} + \upsilon} \tag{6}$$

$$\upsilon = \frac{4}{\pi^2}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h})^2 \tag{7}$$

FPS is another important performance metric for target detection algorithms, it means the number of images that can be processed within per second. Only with high speed can real-time detection be achieved, which is extremely important for some application scenarios.

3. Experiment

3.1. Description of Data

To facilitate comparison with other papers, the datasets used in this paper is Safety Helmet Wearing Dataset (SHWD), which provides datasets for safety helmet wearing and human head detection with a total of 7851 images, including

9044 human safety helmet wearing objects and 111,514 normal head objects (https://github.com/njvisionpower/Safety-Helmet-Wearing-Dataset).

3.2. Experimental Comparison

The effect of target detection is influenced by various factors, among which different experimental parameters, different data sets, different detection targets and different detection algorithm can lead to different detection results, among which the setting of experimental parameters and laboratory hardware conditions are extremely important. The experiment was conducted on a desktop with 32 g RAM, i7-10700K processor, and NVIDIA GeForce RTX 3070 graphics card. Constrained by the hardware conditions of the experiment, the input images are all 640 × 640 in size, and the batch size is set to 4 before all network freeze and 2 after the freeze. In total, the target detection algorithms used in this paper are Centernet, Retinanet, Efficientdet, SSD, YOLOv3, YOLOv4, YOLOv5, and YOLOX, etc. Their specific detection results are shown in **Table 2** below, and the score threshold values of the performance metrics in the table are all 0.5.

3.3. Results Presentation and Analysis (Figure 3)

As shown in **Table 2** this paper has done many groups of experiments to study of helmet detection, in which YOLOv4 performs the best with mAP and FPS values of 94.47% and 27.36%, respectively, and its detection accuracy exceeded all the target detection methods listed above, in which Hat's AP value is 94.43% and Person's AP value is 94.51%, both of which detection result are good, so their overall target detection performs the best. Among all the target detection algorithms of the above table, Efficientdet and Retinanet, the two target detection effects are relatively poor, their mAP values are 64.17% and 62.11%, respectively, and a closer observation of the data in **Table 2** shows that there are some similarities in the reasons for their poor detection effects, their detection AP values for Hat are 83.01% and 89.4%, However, the AP values of Person are 45.84% and 34.83%, which are not satisfactory. The reason for this may be because of the effect of Efficientdet and Retinanet on feature extraction of human face is not very effective, due to this paper mainly studies the target detection algorithm of YOLO series, so it will not be further explored.

With the growing development of target detection algorithm, its detection effect is getting better and better, so some researchers want to deploy the target detection program on the hardware, which can brighten our life, but limited by the performance bottleneck of hardware resources, it is impossible to deploy larger networks on the hardware with limited capabilities, so the lightweight neural networks have attracted attention from more and more researchers, and among the better-performing lightweight networks are ShuffleNet [16], CondenseNet [17], MobileNet, Xception [18], and SqueezeNet [19], which have achieved good detection results in terms of detection accuracy and speed. One of the better-performing and most popular target detection algorithms is MobileNet.

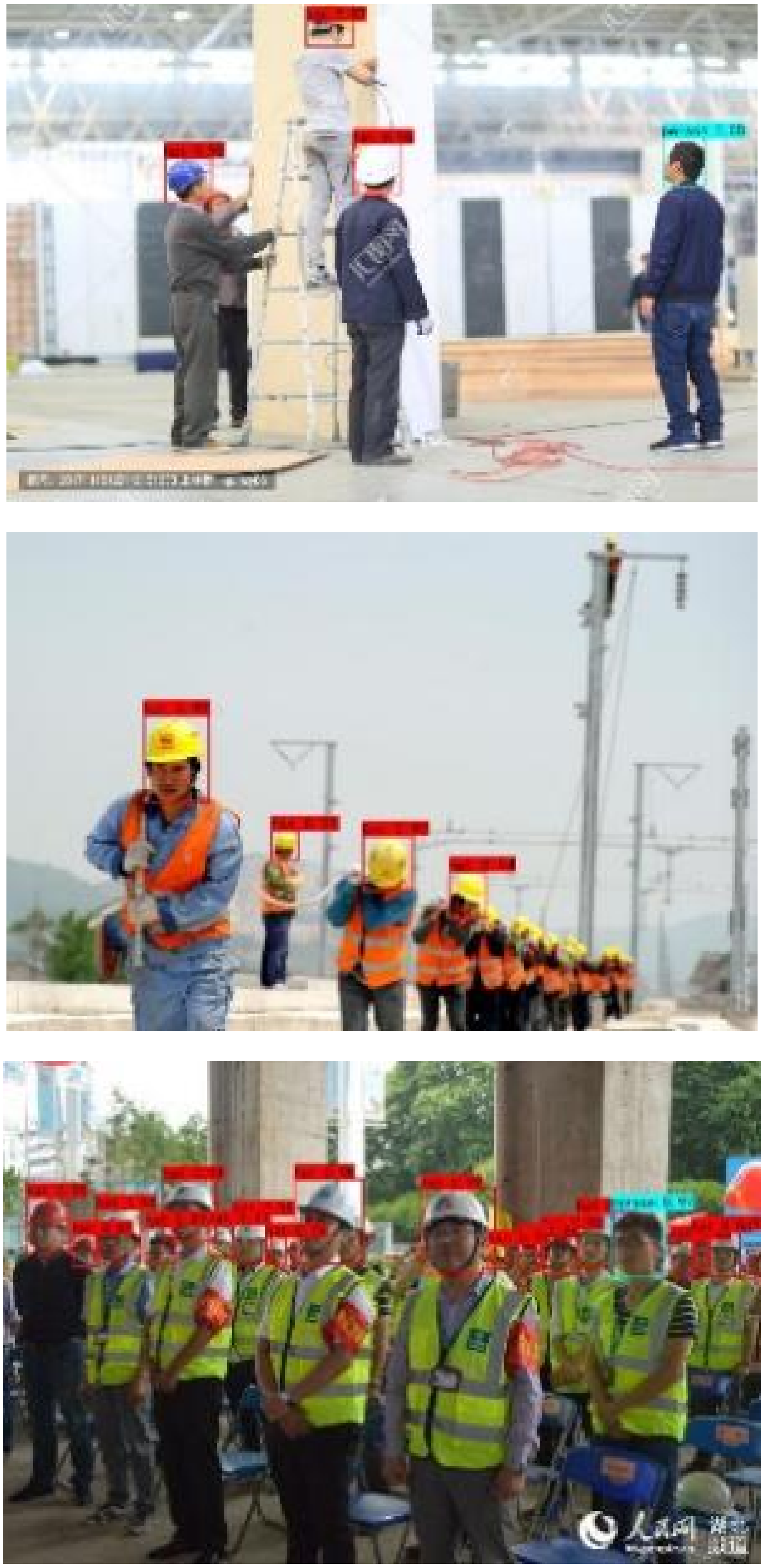

Figure 3. YOLOv4 mobileNetv3 detection result.

Table 2. Comparison of target detection algorithms.

method	backbone	tricks	FPS	mAP	Hat			Persion		
					AP	Precision	recall	AP	Precision	recall
CenterNet	Resnet50	-	50.82	86.09	92.23	98.13	79.26	79.94	97.62	43.16
EfficientDet	Efficientnet	-	15.94	64.17	83.01	77.70	78.02	45.34	96.10	31.92
RetinaNet	Resnet50	-	33.53	62.11	89.40	96.15	82.42	34.83	94.92	26.36
SSD	Mobilenetv2	-	88.13	80.82	87.26	93.75	67.99	74.38	88.44	44.26
SSD	VGG16	-	42.47	91.51	93.18	93.61	84.48	89.83	94.73	54.73
YOLOv3	Darknet53	-	35.14	93.47	94.04	94.14	85.99	92.90	91.59	86.40
YOLOv4	CSPDarknet53	-	**27.36**	**94.47**	94.43	96.02	86.13	94.51	93.37	87.89
YOLOv4	CSPDarknet53	CBAM	26.80	94.03	93.89	96.19	86.68	94.18	93.25	87.46
YOLOv4	CSPDarknet53	ECA	27.52	94.01	93.77	95.65	87.5	94.24	93.07	87.34
YOLOv4	Densenet-121	-	24.88	92.67	92.74	95.05	84.34	92.6	91.99	86.65
YOLOv4	Ghostnet	-	35.73	90.45	0.88	96.58	81.46	0.84	92.54	76.62
YOLOv4	Mobilenetv1	-	59.77	91.11	91.05	94.50	85.03	91.17	91.61	83.21
YOLOv4	Mobilenetv2	-	50.07	91.35	92.10	95.88	83.10	90.59	92.14	81.23
YOLOv4	Mobilenetv3	-	**42.58**	**91.47**	93.23	95.81	84.89	89.71	93.3	76.91
YOLOv4-tiny	CSPDarknet53	-	127.41	79.65	90.8	94.84	83.24	68.49	93.01	54.28
YOLOv4-tiny	CSPDarknet53	CBAM	97.63	81.78	92.58	94.38	83.10	70.97	91.23	59.32
YOLOv4-tiny	CSPDarknet53	ECA	116.59	81.19	91.66	93.30	84.20	70.73	91.73	58.97
YOLOv4-tiny	CSPDarknet53	SE	117.13	81.18	91.50	93.97	83.52	70.87	91.82	58.85
YOLOv5-s	Modified-CSPDarkNet	-	58.15	88.5	94.52	94.89	86.81	82.47	93.81	63.46
YOLOv5-m	Modified-CSPDarkNet	-	44.92	91.53	95.77	92.87	89.42	87.28	94.03	73.51
YOLOX -nano	Modified-CSPDarkNet	-	50.66	86.30	92.88	95.01	83.65	79.71	91.72	60.29
YOLOX-tiny	Modified-CSPDarkNet	-	53.62	89.90	94.66	95.32	86.68	85.14	92.46	70.02
YOLOX-s	Modified-CSPDarkNet	-	51.34	91.28	95.48	95.80	87.64	87.07	92.59	76.15
YOLOX-m	Modified-CSPDarkNet	-	41.35	92.17	95.68	96.50	87.23	88.67	92.68	78.76
YOLOX-l	Modified-CSPDarkNet	-	32.79	92.59	94.98	96.04	86.68	90.19	93.30	80.65

In this paper, all three versions of MobileNet are integrated into YOLOv4, a target detection algorithm, and the target detection mAP and FPS values are 91.11%, 59.77%, 91.35% and 50.07%, 91.47%, 42.58%, respectively, by retraining the helmet detection datasets through migration learning. they all have good detection effects, and further meet the requirements of deploying real-time target detection under the limited hardware conditions at present, so the algorithm can be applied to construction sites in order to remind construction workers to wear helmets in a timely manner in real time, which protects their lives to a certain extent and decreases the degree of injuries suffered from not wearing helmets as required when safety problems occur at construction sites.

4. Conclusions

In this paper, based on publicly available helmet detection datasets, we study helmet detection with Centernet, Retinanet, Efficientdet, SSD, YOLOv3, YOLOv4, YOLOv5 and YOLOX as target detection algorithms respectively. YOLOv4 is an improved version of YOLOv3, and its authors have done a lot of experiments to integrate various effective tricks into YOLOv4, so its detection performance is superior. The YOLOv4 target detection algorithm combined with MobileNet achieves good detection results in terms of accuracy and speed, and meets the requirements of real-time detection. Among the above target detection algorithms, YOLOv5 and YOLOX are the target detection algorithms that emerged after YOLOv4. It is reasonable to say that their detection effect should be better than YOLOv4, which is also illustrated by their performance on the COCO datasets in the authors' paper, but in this datasets of helmet detection, YOLOv4 outperforms them, so the helmet detection system based on YOLOv4 can better achieve the effect of detecting helmets, and it is of great significance for target detection research to be landed for application.

Research on target detection for helmet-based targets is influenced by a variety of factors, of which target detection algorithms and datasets are the most important. This study will consider improving the existing target detection algorithms in the future to achieve the optimal results of overall detection speed and accuracy using a more lightweight network to meet the needs of target detection research landed as an application.

Acknowledgements

This article is supported by National Natural Science Foundation of China Youth Fund, project approval number: 71902121, research on inverse optimization method of fresh cold chain stowage planning based on "time-temperature" flow data analysis.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Wu, W.H. (2021) Safety Helmet Detection Method in Security Monitoring. *Electronic Technology and Software Engineering*, 192-193.

[2] Huang, Y.X. and Huang, H. (2021) Design of Embedded Safety Helmet Detection System Based on YOLO Algorithm. *Electronic Testing*, 58-59+73.

[3] Li, S., Li, L.H., Wang, S.G., Tian, J.Y. and Li, J.F. (2022) Improved YOLOv4 Algorithm for Helmet Detection. *Modern Electronic Technology*, **45**, 103-110.

[4] Zhao, R., Liu, H., Liu, P.L., Lei, Y. and Li, D. (2022) Helmet Detection Algorithm

Based on Improved YOLOv5s. *Journal of Beihang University*, 1-16.

[5] Jin, Y.F., Wu, X., Dong, H., Yu, L. and Zhang, W.A. (2021) Safety Helmet Wearing Detection Algorithm Based on Improved YOLO v4. *Computer Science*, **48**, 268-275.

[6] Wang, C.L., Zhao, Q. and Guo, T. (2021) Deep Learning Helmet Detection Based on Facial Features. *Journal of Shanghai Electric Power University*, **37**, 303-307.

[7] Sun, G.D., Li, C. and Zhang, H. (2022) Helmet Wearing Detection Method Integrating Self-Attention Mechanism. *Computer Engineering and Applications*, 1-7. http://kns.cnki.net/kcms/detail/11.2127.TP.20210621.1819.010.html

[8] Zhao, Z. (2017) Implementation of Human Helmet Detection Based on OpenCV. *Electronic Testing*, 24-25.

[9] Redmon, J. and Farhadi, A. (2018) YOLOv3: An Incremental Improvement. arXiv e-prints.

[10] Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y.M. (2020) Yolov4: Optimal Speed and Accuracy of Object Detection. arXiv preprint arXiv: 2004.10934.

[11] Ge, Z., Liu, S., Wang, F., Li, Z. and Sun, J. (2021) YOLOX: Exceeding YOLO Series in 2021.

[12] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M. and Adam, H. (2017) Mobilenets: Effi-cient Convolutional Neural Networks for Mobile Vison Application. *CoRR*.

[13] Sandler, M., Howard, A., Zhu, M., *et al.* (2018) MobileNetV2: Inverted Residuals and Linear Bottlenecks. *IEEE*.

[14] Howard, A., Sandler, M., Chu, G., Chen, L.C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R. and Vasudevan, V. (IEEE) Searching for MobileNetV3. 2019 *IEEE/CVF International Conference on Computer Vision* (*ICCV*), 2020.

[15] Zoph, B., Vasudevan, V., Shlens, J. and Le, Q.V. (2017) Learning Transferable Architectures for Scalable Image Recognition.

[16] Zhang, X., Zhou, X., Lin, M. and Sun, J. (2017) ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices.

[17] Huang, G. and Liu, S. (2017) VDM Laurens, KQ Weinberger. CondenseNet: An Efficient DenseNet using Learned Group Convolutions.

[18] Chollet, F. (2017) Xception: Deep Learning with Depthwise Separable Convolutions. *IEEE*.

[19] Iandola, F.N., Moskewicz, M.W., Ashraf, K., Han, S., Dally, W.J. and Keutzer, K. (2016) Squeezenet: Alexnet-Level Accuracy with 50x Fewer Parameters and <1mb Model Size. *CoRR*.