JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Comprehensive Review and Performance Analysis of YOLO Architectures for Real-Time Object Detection

Mizanur Rashid[1], Abdullah Ibne Sayed[2], Md Masud Rana[3]

[1]School of Electronic and Information Engineering, Nanjing University of Information Science and Technology, China.
[2]CASCO Signal Ltd, Dhaka, Bangladesh.
[3]Huazhong University of Science and Technology, Wuhan, China.

Abstract : In the field of computer vision, object detection technology has garnered significant attention in recent years due to its wide-ranging applications. Among the various detection algorithms, You Only Look Once (YOLO) stands out as a pioneering approach that formulates object detection as a regression problem, enabling end-to-end training and inference. This unique methodology ensures an optimal balance between speed and accuracy, making YOLO a preferred choice for real-time applications. Over the years, the YOLO series algorithms have evolved considerably, demonstrating remarkable success across diverse domains such as autonomous driving, surveillance, robotics, and medical imaging. This paper provides a comprehensive investigation into the critical applications of YOLO algorithms, highlighting their practical implementations and impact. Furthermore, a detailed comparison is drawn between YOLO and other state-of-the-art object detection frameworks, emphasizing its advantages in terms of computational efficiency, scalability, and adaptability. Based on this analysis, the distinctive characteristics of YOLO-such as its unified detection pipeline, multi-scale feature learning, and lightweight variants-are systematically summarized. Finally, the study explores potential future directions for YOLO, including enhancements in small-object detection, robustness in occluded scenarios, and integration with emerging paradigms like transformer-based architectures. By synthesizing these insights, this review aims to serve as a valuable reference for researchers and practitioners seeking to leverage YOLO's capabilities for next-generation vision systems.

Keywords - YOLO, neural network, object detection and recognition.

I. INTRODUCTION

With the continuous development of deep learning, object detection technology has gradually become a research hotspot. Nowadays, machines can replace traditional manual detection to a certain extent, and even surpass manual detection in terms of accuracy and speed. In the field of object recognition, the more classic algorithms include R-CNN [1], which uses deep learning technology to automatically identify features in images, thereby classifying, predicting, and identifying samples. The fields of computer vision and deep learning continue to develop and make breakthroughs. After R-CNN, many image detection algorithms based on deep learning have emerged, such as Fast RCNN [2], Faster R-CNN [3], Mask R-CNN [4], and YOLO. The YOLO series of algorithms are different from the R-CNN series of algorithms. The YOLO series of algorithms are one-stage algorithms, while the R-CNN algorithm is a two-stage algorithm. Therefore, the YOLO algorithm has a faster detection speed and can be applied to real-time detection and video detection. It has received much attention in recent years.

II. RELATED WORK

The YOLO algorithms have been pivotal in advancing real-time object detection research. Building upon the foundational work of CNN-based architectures, such as R-CNN and its derivatives (Fast R-CNN, Faster R-CNN), which aimed at improving accuracy through region proposals, YOLO introduced a paradigm shift by formulating object detection as a single regression problem, enhancing processing speed significantly [5]. Numerous studies have explored enhancements to YOLO's capabilities by integrating CNN modifications and feature extraction techniques.

For instance, YOLOv2 and YOLOv3 iterations improved detection accuracy and small object recognition by utilizing multi-scale feature maps and the DarkNet-53 architecture, which incorporates residual connections. Researchers have also investigated integrating attention mechanisms and ensemble learning to boost performance further. Furthermore, [6] a growing body of work has emerged focusing on YOLO's application in diverse domains, such as surveillance, autonomous vehicles, and medical image analysis, demonstrating its versatility and resilience in complex environments.

Recent advancements explore hybrid models that combine YOLO with transformer architectures, recognizing their potential to enhance contextual awareness in challenging scenarios. Overall, the continual evolution of YOLO and its modifications represents a key area of research in object detection, pivotal for the development of robust, scalable systems adaptable to various practical applications.

III. YOLO SERIES ALGORITHM DEVELOPMENT PROCESS AND PRINCIPLE

Since the R-CNN series of algorithms cannot meet the current real-time detection requirements in terms of object detection speed, [5]first proposed the YOLO object detection algorithm in 2016. This is a one-stage detection algorithm based on regression. Up to now, the YOLO series has undergone five version updates, and the algorithm has been gradually improved. It has become the mainstream algorithm for real-time object detection.

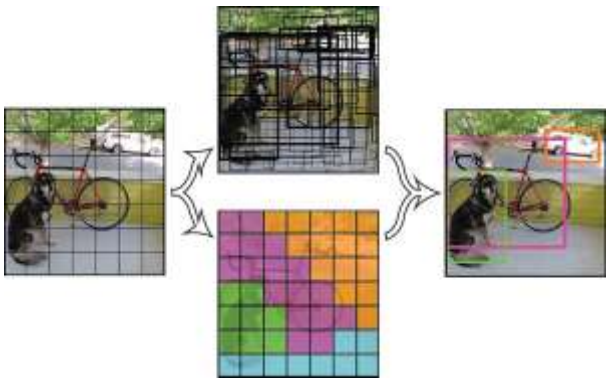

Figure 1. Implementation of YOLOv1.

In the YOLOv1 algorithm, the resolution of each image input (448 × 448) is fixed. The input image will be divided into S × S grids. Each grid is responsible for predicting what the object at the center point within its range belongs to, and will generate 2 bounding boxes (candidate boxes) to predict the shape of the object. Each grid will predict 5 parameter values, namely (x, y, w, h, confidence), where: S × S is the number of grid divisions; x, y are the offset values of the center of the predicted box relative to the grid boundary; w, h are the ratios of the width and height of the predicted box to the width and height of the image; confidence is the IOU value between the predicted box and the real box (1).

In the YOLOv1 algorithm, each grid predicts multiple bounding boxes, but in training, we hope that each object will only output one optimal box for prediction. Therefore, we need to introduce a value here, namely:

$$IOU = (\text{Detection Result} \cap \text{Ground Truth}) / (\text{Detection Result} \cup \text{Ground Truth}) \quad (1)$$

The convolutional network extracts and detects features in the image and outputs the optimal result through the NMS (non-maximum suppression) [6] method.

YOLOv1 uses the GoogLeNet [7] network structure, which uses 24 convolutional layers and 2 fully connected layers.

Since YOLOv1 has a fully connected layer, the input image resolution is fixed to 448 × 448, and the final output formula is:

A. YOLOv1

YOLOv1 is an end-to-end object recognition and detection method proposed by Joseph Redmon in 2016. It can predict the probability of object categories on a complete image. The implementation process of YOLOv1 is shown in Figure 1.

$$S \times S(B \times 5 + C) \quad (2)$$

Where: S × S is the number of divided grids (2); B is the number of prediction boxes, B = 2; 5 is the number of prediction parameters (x, y, w, h, confidence); C is the type of detection and recognition, C = 20.

In YOLOv1, there is a certain error between the predicted (x, y, w, h) and the real (x, y, w, h). Here, a loss function is introduced to minimize this error. The loss function is as follows:

$$\lambda_{\text{coord}} \sum_{i=0}^{s^2} \sum_{j=0}^{B} \amalg_{ij} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] + \lambda_{\text{coord}} \sum_{i=0}^{s^2} \sum_{j=0}^{B} \amalg_{ij} [(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2] +$$

$$\sum_{i=0}^{s^2} \sum_{j=0}^{B} \amalg_{ij}^{\text{obj}} (c_i - \hat{c}_i)^2 + \lambda_{\text{noobj}} \sum_{i=0}^{s^2} \sum_{j=0}^{B} \amalg_{ij}^{\text{noobj}} (c_i - \hat{c}_i)^2 + \sum_{i=0}^{s^2} \amalg_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p_i(c)})^2 \quad (3)$$

Where: $\amalg_{ij}^{obj}$ is the *j* candidate box in grid *i*, responsible for the prediction of obj; $\amalg_{i}^{obj}$ is to determine whether the center of obj is in grid i; $\amalg_{ij}^{noobj}$ is the *j* candidate box in grid *i* that is not responsible for the prediction of obj; *S* is the number of grids; *B* is the number of predicted boxes, in YOLOv1 *B = 2; xi, yi, wi, hi* are predicted values; $\hat{x}_i, \hat{y}_i, \hat{w}_i, \hat{h}_i$ are true values; λ_{coord}, λ_{noobj} are weight parameters (3).

Although the YOLOv1 object detection algorithm has the advantages of fast detection speed and simplicity, it also has many disadvantages, such as each grid can only predict one category, it cannot solve overlapping objects, and the detection effect of small objects is average.

B. YOLOv2

YOLOv2 was proposed [8]in 2017. It mainly improves some shortcomings of YOLOv1 and has significantly improved the accuracy and number of object detections. Since the number of detections of YOLOv2 is as high as more than 9,000, YOLOv2 is also called YOLO 9000. YOLOv2 uses the DarkNet-19 network structure and abandons the GoogLeNet network structure of the YOLOv1 version. The network structure has no fully connected layer and has 5 downsampling operations, all of which are convolution operations. The 1 × 1 convolution operation is to save parameters.

In the YOLOv1 version, since there are two fully connected layers, Dropout is used to prevent the network from overfitting [9],[10],[11],[12]. In the YOLOv2 version, there is no fully connected layer, so Dropout is abandoned. Batch Normalization is added after each convolution [13], so that the input of each layer of the network is normalized, which makes convergence relatively easier and enhances the generalization ability of the network. In addition, YOLOv2 has performed 10 fine-tunings at a large resolution based on the training of the YOLOv1 version, so that the network can adapt to higher resolution images; the k-means clustering algorithm is used to cluster the bounding boxes of the training set.

Since the receptive field of the feature map of the last layer is too small and small objects may be lost, the feature fusion is improved and the network is trained on the ImageNet [14] and COCO [15]datasets simultaneously using joint optimization technology.

C. YOLOv3

YOLOv3 was proposed [16] in 2018. Compared with YOLOv2 and YOLOv1, YOLOv3 mainly improved the network structure. It used the DarkNet-53 network structure to achieve simultaneous improvement in object detection speed and accuracy, making it more suitable for the detection of small targets. It introduced the idea of Res-Net (residual network) [17] and stacked more layers for feature extraction. At the same time, it adopted Spatial Pyramid Pooling Networks (spatial pyramid network algorithm)[18] to achieve multi-size input and same-size output.

D.YOLOv4

In April 2020, [19] improved YOLOv3 and proposed the more powerful YOLOv4 algorithm, which is characterized by its integration. Its network structure adopts CSP DarkNet-53 [20]. In order to make YOLOv4 adapt to inputs of different sizes, the SPP-Net structure is introduced. In order to make full use of feature fusion, PANet [21] is also introduced. YOLOv4 has conducted a large number of tests on some commonly used tricks in deep learning, mainly including the following contents.

1) Input: YOLOv4 improves the input through data enhancement, cmBN, and SAT self-adversarial training;
2) Backbone main network: The CSP network structure is integrated into the DarkNet-53 network structure, and YOLOv4 uses the upgraded CSP DarkNet-53 network;
3) Activation function: In order to obtain better accuracy and generalization, the Mish activation function is used in Backbone;
4) Dropblock: In YOLOv4, Dropblock, which is similar to the Dropout function, is used to alleviate overfitting[22];
5) Neck: The SPP module and FPN + PAN structure are used.

E.YOLOv5

Shortly after YOLOv4 came out, relevant researchers launched the YOLOv5 algorithm in June 2020. Unlike the previous 4 versions, the YOLOv5 algorithm only has code and no relevant papers have been published. It is the result of engineering practice, so the specific performance cannot be compared with other object detection algorithms. Although there are no relevant papers on the YOLOv5 algorithm, it can be seen from its code that the YOLOv5 version is essentially the same as the YOLOv4 version, and the author did not compare it with other YOLO algorithms. In terms of size, YOLOv4 has 244 MB, while YOLOv5 has only 27 MB. The model size has been lightweight, but it is comparable to YOLOv4 in terms of object detection speed. In YOLOv5, the intermediate/hidden layer activation function uses Leaky ReLU, and the final detection layer activation function is Sigoid; the input end uses adaptive anchor box calculation and adaptive image scaling methods.

After several versions of improvements, the YOLO series of algorithms have become dominant in the field of object detection.

IV. ALGORITHM PERFORMANCE COMPARISON

The performance comparison of different object detection algorithms in the PASCAL VOC [23] series and MS COCO[24] test sets is shown in Table 1. Since these algorithms have differences in network structure, software and hardware, and parameters, in order to make the listed data have effective reference value, the results in the table are all performances under objective and approximate conditions. In the data set test, the two important indicators for measuring the quality of the algorithm are mean average precision (WmAP) and frames per second (FPS). The former mainly reflects the monitoring accuracy, and the latter mainly reflects the detection accuracy. From the summary table, it can be concluded that the YOLO algorithm exceeds other algorithms in terms of object detection speed, but the detection accuracy needs to be improved. From this, we can see the future development direction of the YOLO series of algorithms.

TABLE 1. Performance of each object detection algorithm in the dataset

Algorithm Name	Network structure	VOC 2007 (mAP@0.5)	VOC 2012 (mAP@0.5)	MS COCO (mAP@0.5:0.95)	FPS
YOLOv1	GoogLeNet	63.4	57.9		45.0
YOLOv2	Dark Net-19	78.6	73.4	21.6	59.0
YOLOv3	Dark Net-53			33.0	20.0
YOLOv4	CSP Dark Net-53			43.5	33.0
R-CNN	Alex Net	58.5	53.3		0.1
Fast R-CNN	VGG-16	70.0	68.4	19.7	0.5
Faster R-CNN	Res Net-101	76.4	75.9	21.9	5.0
SSD	VGG-16	74.3	75.9	26.8	59.0
DSSD	Res Net-101	81.5			5.5
RSSD	VGG-16	80.8			16.6
FSSD	VGG-16	80.9			35.7

V. APPLICATION EXAMPLES

The YOLO series of algorithms are used in many fields in object recognition and detection applications, such as security, military, medical, autonomous driving, transportation, industry, agriculture, etc.

A.TRANSPORTATION

Used the YOLOv1 algorithm to detect vehicles in real time[25]. The results showed that the algorithm had high detection speed and accuracy, and basically met the requirements of real-time monitoring. [26] used the improved YOLOv2 algorithm to detect traffic signs in real time. Experiments showed that the method had faster speed and better robustness, and the fastest detection speed could reach 0.017 s. [27] applied the YOLOv2 object detection algorithm to the automatic driving system (ADS) and the driver assistance system (DAS). Experiments showed that the algorithm improved the accuracy of vehicle detection without reducing the detection speed. Enhanced YOLOv3 micro-network algorithm and applied it to ship detection[28]. It achieved a good balance between detection accuracy and real-time performance and was more suitable for actual scenarios. Speed measurement in aircraft type detection[29]. At the beginning of the development of YOLOv5, the author did not compare it with other algorithms, so there is no performance test result of relevant data sets, and no comparison is made here. YOLOv5 algorithm, experiments show that the algorithm can effectively detect the type of aircraft in optical remote sensing images; [30]a new multi-sensor multi-level enhanced convolutional network model - MMEYOLO. MME-YOLO consists of two tightly coupled structures, namely the enhanced reasoning head and the LiDAR-Image composite module. The enhanced reasoning head provides the network with stronger reasoning capabilities through the attention-guided feature selection block and the anchor/anchor-free integration head. In addition, the LiDAR-Image composite module cascades the multi-level feature map from the LiDAR subnet to the image subnet, thereby enhancing the generalization of the detector in complex scenes. It has been tested that even at night peaks and inconsistent lighting conditions, it can ensure high accuracy.

B.MILITARY AND SECURITY FIELDS

The improved YOLOv2 algorithm to the recognition of armored vehicles[31]. It was verified that this method can effectively and accurately recognize specific armored targets in real time. The improved YOLOv3 algorithm to quickly identify and detect missiles during the attack process[32], proving that the algorithm has high application value in the military. An improved Fire-YOLO deep learning algorithm for detecting small targets, fire-like and smoke-like targets in forest fire images, as well as fire detection under different natural light conditions[33]. The Fire-YOLO detection model expands the feature extraction network from three dimensions, enhances the feature propagation of small fire target recognition, improves network performance, and reduces model parameters. Experiments have verified that this method can effectively detect small fires, fire-like and smoke-like targets. When the input image size is 416 × 416 resolution, the average detection time is 0.04 s per frame, which can meet the needs of real-time fire detection.

C. MEDICAL FIELD

Combined the YOLOv3 and U-Net methods to accurately extract overlapping and adherent chromosomes[34]. This method is of great significance to the development of automatic chromosome karyotype analysis technology;[35] the YOLOv5 algorithm in the left and right identification of medical surgical gloves. Experiments showed that the algorithm can basically realize the left and right identification of medical surgical gloves;[36] the YOLOv3 detection algorithm to the examination of breast X-ray lumps and distinguished between malignant and benign lumps. The experiment used breast X-ray photographs of different resolutions and detected the INbreast breast X-ray dataset based on YOLOv3. The average accuracy was 94.2% and 84.6% respectively. The lumps were classified as benign and malignant, and the detection results were good.

D. INDUSTRIAL FIELD

The YOLOv2 object detection algorithm in the intelligent identification and positioning of coal and rock, and compared it with Faster R-CNN and SSD[37]. The experimental results showed that the YOLOv2 algorithm was superior to the other two algorithms in terms of both accuracy and speed. YOLOv2 was more suitable for accurate and fast identification of coal and rock; [38]an improved YOLOv3 object detection algorithm. It was verified that the algorithm can successfully solve the problem of insensitivity in detecting bearing cover defects. Experiments show that the improved YOLOv3 can achieve real-time detection; [39]an improved Tiny-YOLOv3 object detection algorithm. On the basis of Tiny-YOLOv3, a residual network structure based on convolutional neural network was added. The detection accuracy was improved while ensuring the real-time detection, which is more suitable for the detection of obstacles in mines; [40]improved Tiny-YOLOv3 object detection algorithm based on the improved YOLOv4-Tiny network, proposed a detection algorithm for real-time detection of electronic components on a conveyor belt. Experiments show that compared with other mainstream algorithms, this algorithm can maintain the highest detection accuracy at the fastest speed; [41]proposed an improved YOLOv4 detection algorithm to address the problem that surface defects of aluminum are difficult to detect. The algorithm enhances the feature fusion capability of the neck network and adds the SENet attention mechanism to the network, which improves the detection accuracy overall. The proposed algorithm meets the requirements of aluminum defect detection; [42]developed a variant YOLOv4 algorithm based on the YOLOv4 network model. This variant algorithm can simultaneously achieve weld target detection, fewer defect classification parameters, and high speed in the appearance inspection of white body welds[43] automatic detection and reading of pointer instruments based on the YOLOv4 algorithm. The experimental results show the speed and effectiveness of the algorithm.

D. AGRICULTURE

A YOLOv3-Litchi algorithm based on YOLOv3[44], which can perform real-time detection of densely distributed litchi fruits in large visual scenes; [45] YOLOv5-Ours algorithm based on YOLOv5. Experiments showed that the algorithm can realize real-time detection of kiwifruit defects and is a robust kiwifruit flaw detection strategy;[46] Improved YOLOv4 tomato fruit detection method combined with transfer learning for tomato fruits in 6 complex conditions in greenhouses. The model was pre-trained using the ImageNet dataset and VGG-16, and the weight parameters of the model were improved. The research results showed that the improved model had an average detection accuracy of more than 90% for tomato fruits in 6 complex conditions, and the

detection accuracy of maturity was also better than the original model. This method realized the detection of tomato fruits in complex environments.

VI. Summary And Outlook

Object detection algorithms have important application value in many fields. The YOLO algorithm stood out in the early days due to its detection speed and once became the most popular object detection algorithm. This article reviews the development history of the YOLO series of algorithms and their applications in some fields. It is learned that the series of algorithms can still be optimized in practical application scenarios, which is also the future development trend of the YOLO algorithm. Based on the current status of target detection, the following outlook is made:

1) Although small target detection has been optimized in YOLOv3, the optimization in this aspect is not obvious in subsequent versions, and there is still a lot of room for improvement in small target detection or occluded part detection;
2) Very light changes have been implemented in the YOLOv5 version, which is still a development direction;
3) A suitable bounding box can save the time of drawing the box, thereby speeding up the detection speed;
4) The speed and accuracy of detection should be balanced, and one aspect should not be pursued blindly;
5) The YOLO algorithm can be integrated with other object detection algorithms to obtain the optimal detection algorithm by taking advantage of their strengths and weaknesses;
6) The data set is not complete and needs to be developed.

References

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA: IEEE, Jun. 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.

[2] R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile: IEEE, Dec. 2015, pp. 1440–1448. doi: 10.1109/ICCV.2015.169.

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," Jan. 06, 2016, *arXiv*: arXiv:1506.01497. doi: 10.48550/arXiv.1506.01497.

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," Jan. 24, 2018, *arXiv*: arXiv:1703.06870. doi: 10.48550/arXiv.1703.06870.

[5] 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)*, Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[6] A. Neubeck and L. Van Gool, "Efficient Non-Maximum Suppression," in *18th International Conference on Pattern Recognition (ICPR'06)*, Hong Kong, China: IEEE, 2006, pp. 850–855. doi: 10.1109/ICPR.2006.479.

[7] C. Szegedy *et al.*, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA: IEEE, Jun. 2015, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

[8] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI: IEEE, Jul. 2017, pp. 6517–6525. doi: 10.1109/CVPR.2017.690.

[9] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 56, pp. 1929–1958, 2014.

[10] T. Xiao, H. Li, W. Ouyang, and X. Wang, "Learning Deep Feature Representations with Domain Guided Dropout for Person Re-identification," Apr. 26, 2016, *arXiv*: arXiv:1604.07528. doi: 10.48550/arXiv.1604.07528.

[11] S. I. Wang and C. D. Manning, "Fast dropout training," in *International Conference on Machine Learning*, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:10357959

[12] P. Baldi and P. Sadowski, "The dropout learning algorithm," *Artif. Intell.*, vol. 210, pp. 78–122, May 2014, doi: 10.1016/j.artint.2014.02.004.

[13] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," Mar. 02, 2015, *arXiv*: arXiv:1502.03167. doi: 10.48550/arXiv.1502.03167.

[14] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," Jan. 30, 2015, *arXiv*: arXiv:1409.0575. doi: 10.48550/arXiv.1409.0575.

[15] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," Feb. 21, 2015, *arXiv*: arXiv:1405.0312. doi: 10.48550/arXiv.1405.0312.

[16] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," Apr. 08, 2018, *arXiv*: arXiv:1804.02767. doi: 10.48550/arXiv.1804.02767.

[17] X. Zhang, J. Zou, K. He, and J. Sun, "Accelerating Very Deep Convolutional Networks for Classification and Detection," Nov. 18, 2015, *arXiv*: arXiv:1505.06798. doi: 10.48550/arXiv.1505.06798.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," vol. 8691, 2014, pp. 346–361. doi: 10.1007/978-3-319-10578-9_23.

[19] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 23, 2020, *arXiv*: arXiv:2004.10934. doi: 10.48550/arXiv.2004.10934.

[20] C.-Y. Wang, H.-Y. M. Liao, I.-H. Yeh, Y.-H. Wu, P.-Y. Chen, and J.-W. Hsieh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," Nov. 27, 2019, *arXiv*: arXiv:1911.11929. doi: 10.48550/arXiv.1911.11929.

[21] M. Zhang *et al.*, "Application of Lightweight Convolutional Neural Network for Damage Detection of Conveyor Belt," *Appl. Sci.*, vol. 11, no. 16, p. 7282, Aug. 2021, doi: 10.3390/app11167282.

[22] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "DropBlock: A regularization method for convolutional networks," Oct. 30, 2018, *arXiv*: arXiv:1810.12890. doi: 10.48550/arXiv.1810.12890.

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

[24] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," Feb. 21, 2015, *arXiv*: arXiv:1405.0312. doi: 10.48550/arXiv.1405.0312.
[25] Y. Zhang, Z. Guo, J. Wu, Y. Tian, H. Tang, and X. Guo, "Real-Time Vehicle Detection Based on Improved YOLO v5," *Sustainability*, vol. 14, no. 19, p. 12274, Sep. 2022, doi: 10.3390/su141912274.
[26] J. Zhang, M. Huang, X. Jin, and X. Li, "A Real-Time Chinese Traffic Sign Detection Algorithm Based on Modified YOLOv2," *Algorithms*, vol. 10, no. 4, p. 127, Nov. 2017, doi: 10.3390/a10040127.
[27] X. Han, J. Chang, and K. Wang, "Real-time object detection based on YOLO-v2 for tiny vehicle object," *Procedia Comput. Sci.*, vol. 183, pp. 61–72, 2021, doi: 10.1016/j.procs.2021.02.031.
[28] H. Li, L. Deng, C. Yang, J. Liu, and Z. Gu, "Enhanced YOLO v3 Tiny Network for Real-Time Ship Detection From Visual Image," *IEEE Access*, vol. 9, pp. 16692–16706, 2021, doi: 10.1109/ACCESS.2021.3053956.
[29] S. Luo, J. Yu, Y. Xi, and X. Liao, "Aircraft Target Detection in Remote Sensing Images Based on Improved YOLOv5," *IEEE Access*, vol. 10, pp. 5184–5192, 2022, doi: 10.1109/ACCESS.2022.3140876.
[30] J. Zhu, X. Li, P. Jin, Q. Xu, Z. Sun, and X. Song, "MME-YOLO: Multi-Sensor Multi-Level Enhanced YOLO for Robust Vehicle Detection in Traffic Surveillance," *Sensors*, vol. 21, no. 1, p. 27, Dec. 2020, doi: 10.3390/s21010027.
[31] J. Sang *et al.*, "An Improved YOLOv2 for Vehicle Detection," *Sensors*, vol. 18, no. 12, p. 4272, Dec. 2018, doi: 10.3390/s18124272.
[32] 刘志赢, "Application of Cascade R-CNN and YOLOv3 in Missile Target Recognition," *J. Image Signal Process.*, vol. 09, no. 02, pp. 102–110, 2020, doi: 10.12677/JISP.2020.92013.
[33] L. Zhao, L. Zhi, C. Zhao, and W. Zheng, "Fire-YOLO: A Small Target Object Detection Method for Fire Inspection," *Sustainability*, vol. 14, no. 9, p. 4930, Apr. 2022, doi: 10.3390/su14094930.
[34] H. Bai, T. Zhang, C. Lu, W. Chen, F. Xu, and Z.-B. Han, "Chromosome Extraction Based on U-Net and YOLOv3," *IEEE Access*, vol. 8, pp. 178563–178569, 2020, doi: 10.1109/ACCESS.2020.3026483.
[35] A. W. Kiefer, D. Willoughby, R. P. MacPherson, R. Hubal, and S. F. Eckel, "Enhanced 2D Hand Pose Estimation for Gloved Medical Applications: A Preliminary Model," *Sensors*, vol. 24, no. 18, p. 6005, Sep. 2024, doi: 10.3390/s24186005.
[36] G. H. Aly, M. Marey, S. A. El-Sayed, and M. F. Tolba, "YOLO Based Breast Masses Detection and Classification in Full-Field Digital Mammograms," *Comput. Methods Programs Biomed.*, vol. 200, p. 105823, Mar. 2021, doi: 10.1016/j.cmpb.2020.105823.
[37] S. Chuanmeng, L. Xinyu, C. Jiaxin, W. Zhibo, and L. Yong, "Coal-Rock Image Recognition Method for Complex and Harsh Environment in Coal Mine Using Deep Learning Models," *IEEE Access*, vol. 11, pp. 80794–80805, 2023, doi: 10.1109/ACCESS.2023.3300243.
[38] Z. Zheng, J. Zhao, and Y. Li, "Research on Detecting Bearing-Cover Defects Based on Improved YOLOv3," *IEEE Access*, vol. 9, pp. 10304–10315, 2021, doi: 10.1109/ACCESS.2021.3050484.
[39] D. Xiao, F. Shan, Z. Li, B. T. Le, X. Liu, and X. Li, "A Target Detection Model Based on Improved Tiny-Yolov3 Under the Environment of Mining Truck," *IEEE Access*, vol. 7, pp. 123757–123764, 2019, doi: 10.1109/ACCESS.2019.2928603.
[40] L. Wang, X. Liu, J. Ma, W. Su, and H. Li, "Real-Time Steel Surface Defect Detection with Improved Multi-Scale YOLO-v5," *Processes*, vol. 11, no. 5, p. 1357, Apr. 2023, doi: 10.3390/pr11051357.
[41] J. Wang, Y. S. Zhang, F. Pan, and L. Wang, "Defect Detection of Aluminum Profiles based on Improved Feature Pyramids," *MATEC Web Conf.*, vol. 380, p. 01016, 2023, doi: 10.1051/matecconf/202338001016.
[42] H. Huang, X. Peng, S. Wu, W. Ou, X. Hu, and L. Chen, "An Automotive Body-in-White Welding Stud Flexible and Efficient Recognition System," *IEEE Access*, vol. 13, pp. 51938–51955, 2025, doi: 10.1109/ACCESS.2025.3553691.
[43] J. Peng, M. Xu, and Y. Yan, "Automatic Recognition of Pointer Meter Reading Based on Yolov4 and Improved U-net Algorithm," in *2021 IEEE International Conference on Electronic Technology, Communication and Information (ICETCI)*, Changchun, China: IEEE, Aug. 2021, pp. 52–57. doi: 10.1109/ICETCI53161.2021.9563496.
[44] H. Wang *et al.*, "YOLOv3-Litchi Detection Method of Densely Distributed Litchi in Large Vision Scenes," *Math. Probl. Eng.*, vol. 2021, pp. 1–11, Feb. 2021, doi: 10.1155/2021/8883015.
[45] J. Yao, J. Qi, J. Zhang, H. Shao, J. Yang, and X. Li, "A Real-Time Detection Algorithm for Kiwifruit Defects Based on YOLOv5," *Electronics*, vol. 10, no. 14, p. 1711, Jul. 2021, doi: 10.3390/electronics10141711.
[46] P. L. T. Mbouembe, G. Liu, J. Sikati, S. C. Kim, and J. H. Kim, "An efficient tomato-detection method based on improved YOLOv4-tiny model in complex environment," *Front. Plant Sci.*, vol. 14, p. 1150958, Apr. 2023, doi: 10.3389/fpls.2023.1150958.