

ARTICLE TYPE

YOLOv7-RepFPN: Improving Real-Time Performance of Laparoscopic Tool Detection on Embedded Systems

Yuzhang Liu¹ | Yuichiro Hayashi¹ | Masahiro Oda^{2,1} | Takayuki Kitasaka³ | Kensaku Mori^{1,2,4}

¹Graduate School of Informatics, Nagoya University, Aichi, Japan

²Information and Communications, Nagoya University, Aichi, Japan

³Department of Information Science, Aichi Institute of Technology, Aichi, Japan

⁴Research Center of Medical Bigdata, National Institute of Informatics, Tokyo, Japan

Correspondence

Corresponding author Yuzhang Liu,
Email: yzliu@mori.m.is.nagoya-u.ac.jp.
Corresponding author Kensaku Mori,
Email: kensaku@is.nagoya-u.ac.jp.

Abstract

This study focuses on enhancing the inference speed of laparoscopic tool detection on embedded devices. Laparoscopy, a minimally invasive surgery (MIS) technique, markedly reduces patient recovery times and postoperative complications. Real-time laparoscopic tool detection helps assisting laparoscopy by providing information for surgical navigation, and its implementation on embedded devices is gaining interest due to the portability, network independence and scalability of the devices. However, embedded devices often face computation resource limitations, potentially hindering inference speed. To mitigate this concern, our work introduces a two-fold modification to the YOLOv7 model: we halve the feature channels and integrate RepBlock, yielding the YOLOv7-RepFPN model. This configuration leads to a significant reduction in computational complexity. Additionally, we employ the focal EIoU (efficient intersection of union) loss function for bounding box regression. Experiment results on an embedded device demonstrate that for frame-by-frame laparoscopic tool detection, our proposed YOLOv7-RepFPN achieves an mAP of 88.2% (with IoU set to 0.5) on a custom dataset based on EndoVis17, and an inference speed of 62.9 FPS. Contrasting with the original YOLOv7, which garnered an 89.3% mAP and 41.8 FPS under identical conditions, our methodology enhances the speed by 21.1 FPS while maintaining detection accuracy. This emphasizes the effectiveness of our work.

KEYWORDS

Laparoscopic Tool Detection, YOLOv7, RepVGG, focal EIoU, Embedded Devices.

1 | INTRODUCTION

Laparoscopy, a form of minimally invasive surgery, reduces patient trauma and hastens recovery times^{1,2}. However, it introduces unique challenges due to a limited field of view and heightened requirements for accurate hand-eye coordination³. Laparoscopic tool detection assists in managing these challenges by supplying information on tool identification and position, thereby facilitating surgical navigation⁴. Image-based laparoscopic tool detection is preferred as it offers real-time visual feedback on laparoscopic videos and removes the need for calibration compared to sensor-based techniques, thereby streamlining surgical procedures⁵.

Laparoscopic tool detection requires high inference speed without compromising accuracy. Currently, the task mainly relies on CNN models like RCNN^{6,7} and YOLO⁸. These models are evaluated on two factors. One is detection accuracy which is crucial for effective surgical information. The other is real-time inference speed, matching the laparoscopic video's frame rate⁹. A detection speed of 25-30 FPS is generally seen as real-time^{10,11}.

However, an inference speed of 25-30 FPS may be insufficient for clinical demands. For surgical navigation, advanced utilization of laparoscopic frame information could involve the extraction of more details, such as segmentation masks, laparoscopic tool tracking and pose^{12,13}, as well as surgical phases and actions^{14,15}, in addition to laparoscopic tool detection. Such tasks may demand modification of the detection model. These modifications can possibly increase complexity of model

and slow inference speed. For instance, augmenting YOLOv5¹⁶ with YOLACT¹⁷ for segmentation outputs slows the inference speed to less than two-thirds of the original YOLOv5 and struggles to maintain accuracy. To ensure the accuracy of these additional tasks, integrating LSTM or attention mechanisms into the detection model^{18,19} is proven feasible. However, these modules significantly increase computational complexity and hence, negatively impact the inference speed. Thus, pursuing higher detection speeds is significant for optimizing laparoscopic video information use.

The demand for deployment of medical image processing, including laparoscopic tool detection on embedded devices is increasing. This lends greater significance to enhance inference speed of laparoscopic tool detection. Embedded devices offer portability and save operating room space, enhancing clinical workflows. They also safeguard patient privacy by operating without network reliance and scale well for wide medical image processing applications^{20,21,22}. However, their limited computational capacity²³ can slow detection models, posing challenges for tasks that build on laparoscopic tool detection. For example, RCNN-based models^{24,25,26} struggle to meet the frame rate of laparoscopic videos even on powerful servers, due to their two-stage process of object localization and classification. In contrast, YOLO-based models^{27,28,29}, which are one-stage, are faster and show similar detection accuracy compared with RCNN-based models, making them more suitable for laparoscopic tool detection on embedded devices.

Existing YOLO-based models for laparoscopic tool detection often overlook the specific characteristics of laparoscopic videos. In this study, we chose YOLOv7³⁰, a model known for its superior speed and accuracy among YOLO models, as the baseline. We then made improvements, considering the distinctive features of laparoscopic videos.

Firstly, compared to general object detection datasets^{31,32} used for evaluating YOLO models, laparoscopic videos^{10,33} have fewer target categories and more uniform background information. Lightweight models, due to their low complexity, can achieve high accuracy in detecting laparoscopic tools while being faster on resource-limited embedded devices. However, current YOLO-based models for laparoscopic tool detection use large-scale structures for accuracy. To address this, we halved the number of feature channels within YOLOv7, creating a more lightweight structure, and introduced the inference-friendly RepBlock³⁴ to enhance the inference speed.

Secondly, objects in these datasets are often tilted, unlike the vertically or horizontally oriented objects typical in general datasets. Bounding boxes, aligned with the frame axis, may include unnecessary information. To improve detection accuracy, we modified the loss function for bounding box regression (BBR). This step, though crucial, is often overlooked by current YOLO-based models. Based on this, we adopted the focal EIou³⁵ as the loss function for BBR, aiming to improve detection accuracy.

The primary objective of this study is to construct a lightweight model that can maximize the speed of laparoscopic tool detection on embedded devices while preserving accuracy. The primary contributions of this study are as follows:

- We propose YOLOv7-RepFPN model, which tailors to the lower complexity of laparoscopic frame data compared to general object detection data by reducing feature channels and integrating the efficient RepBlock.
- For improving accuracy of laparoscopic tool detection, we refined the training process with the application of the focal EIou (efficient intersection over union) loss function for bounding box regression (BBR).
- Comparative experiments conducted on an embedded device using a custom dataset based on a widely-used benchmark, have shown the effectiveness of our proposal. Furthermore, ablation studies have been carried out to reveal the impact of our modifications to YOLOv7.

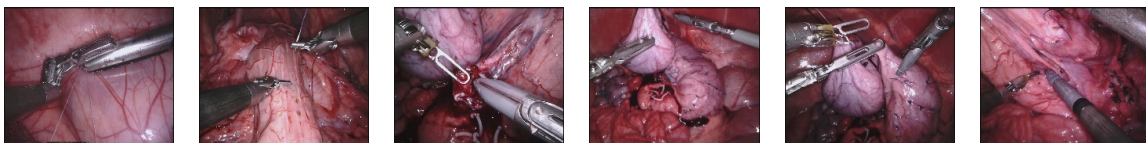


FIGURE 1 Examples of laparoscopic frames containing tools oriented at an angle. The laparoscopic frames in the figure are picked out from the EndoVis17 dataset.

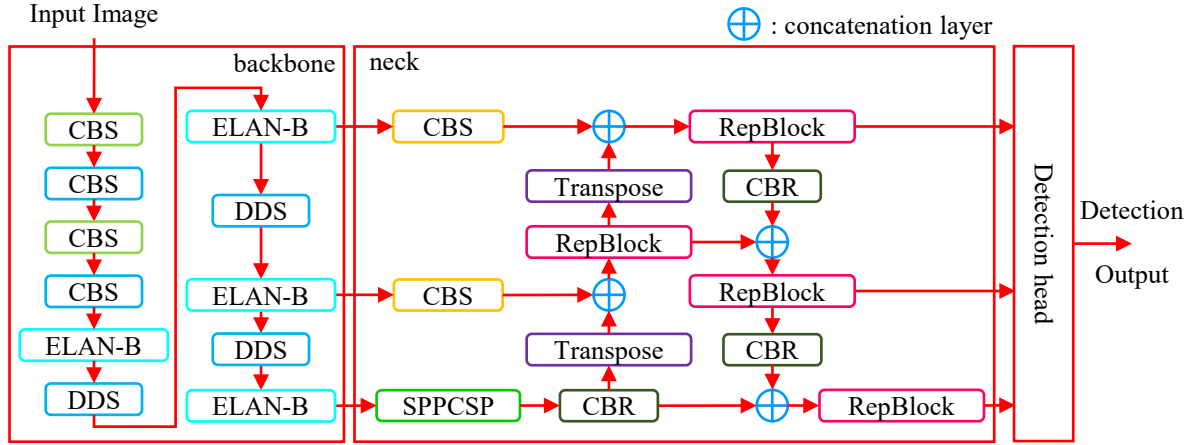


FIGURE 2 Architecture of the YOLOv7-RepFPN model. Detailed explanations of key components of the network are provided in subsequent part of the section.

2 | METHODOLOGY

2.1 | Characteristics of Laparoscopic Videos

In this study, our goal is to improve the speed of laparoscopic tool detection on embedded devices while maintaining detection accuracy. To do this, we adjust the YOLOv7 model based on the features of laparoscopic videos. Unlike the typical datasets for general object detection tasks, laparoscopic videos present the following distinctive characteristics:

- Laparoscopic videos present lower data complexity. For example, the COCO³¹ dataset used for YOLO performance evaluation includes over 200,000 diverse images with 80 target categories, spanning varied scenes like city streets, zoos, and landscapes, leading to complex background information. Laparoscopic image datasets, like Cholec80¹⁰ containing 80 laparoscopic videos, and EndoVis17³³ with 3000 laparoscopic frames extracted from different laparoscopic videos, only have 7 target categories. Since laparoscopy focuses on specific organs, the background in laparoscopic videos is more uniform³⁶, limited to abdominal organs and tissues.
- Laparoscopic tools in laparoscopic videos are usually oriented at an angle, as shown in Figure 1. Unlike the objects in the general detection datasets, which are typically vertical or horizontal. As a result, axis-aligned bounding boxes for target localization in detection tasks may contain extraneous information due to the tilted laparoscopic tools.

2.2 | Overview of the Methodology

In this study, changes in architecture and optimization approach were made to the YOLOv7 baseline.

In terms of architecture, considering the relatively low complexity of laparoscopic videos, we reduced the number of channels in YOLOv7 to make the model lightweight, as well as incorporate RepBlock³⁴ to enhance inference speed, leading to the formulation of YOLOv7-RepFPN model. Considering the prevalence of tilted tools in the laparoscopic image dataset, we substitute CIoU³⁷ (complete IoU) in YOLOv7 with focal EIou³⁵ loss function for BBR.

To aid understanding to our proposed method, section 2.3 presents a detailed discussion of the proposed YOLOv7-RepFPN model, and section 2.4 focuses on the implementation of the focal loss function for BBR.

2.3 | Architecture of YOLOv7-RepFPN

Figure 2 shows the architecture of our YOLOv7-RepFPN. We kept YOLOv7's design for both the backbone and the detection head sections. Since the detection head shared the same structure with YOLOv4³⁸ and YOLOv5¹⁶, which have been thoroughly

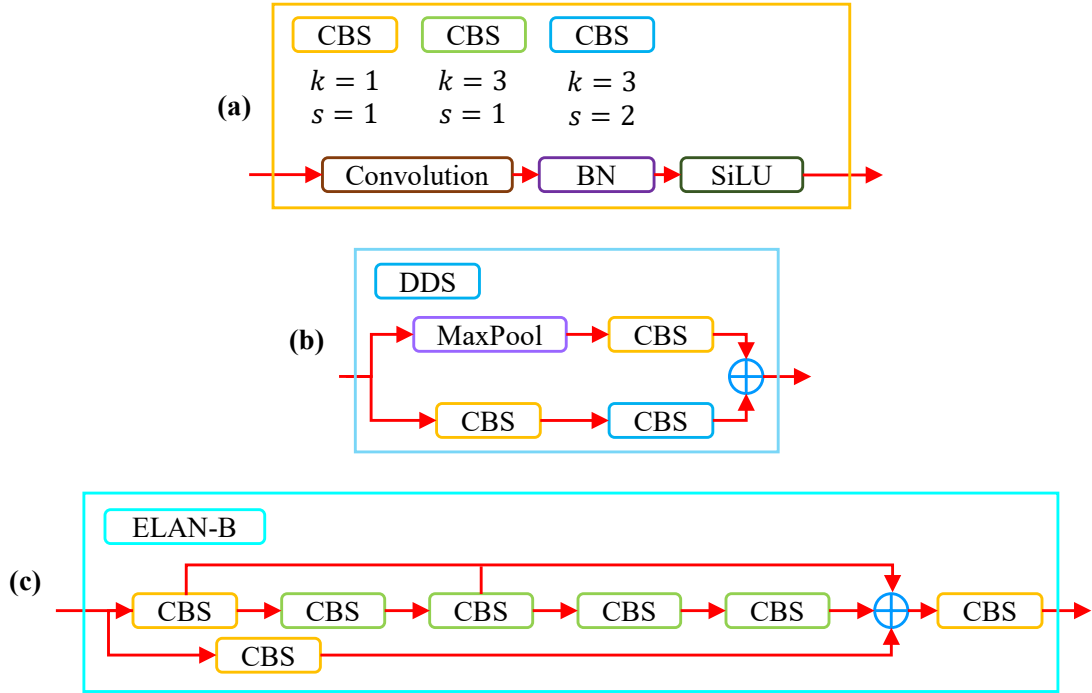


FIGURE 3 Structures of key modules in the backbone section of YOLOv7-RepFPN. (a): Structure of CBS module. BN and SiLU denote Batch Normalization and Sigmoid Linear Unit activation function. The color of the CBS module corresponds to its convolutional layer parameters, consistent across all figures in this paper. k and s represent the kernel size and stride of the convolutional layer, respectively. (b): Structure of DDS (double downsampling) module. MaxPool represents the max pooling layer. (c): Structure of ELAN-B module.

described in prior studies, we skip a detailed introduction. YOLOv7-RepFPN utilizes the YOLOv7³⁰ backbone for efficient feature extraction; key components of the backbone section are illustrated in Figure 3.

Figure 3(a) illustrates the basic convolutional block, CBS, of YOLOv7-RepFPN. This module is composed of a convolution layer, a batch normalization (BN) layer, and a SiLU activation function. By adjusting the kernel size and stride of the convolution layer, diverse functionalities can be achieved.

Figure 3(c) showcases the ELAN-B module from YOLOv7, which offers multiple forward paths, aiding in model optimization and mitigating overfitting for simpler data. In contrast, the DDS module, depicted in Figure 3(b), combines max pooling and convolution layers for downsampling, ensuring minimal information loss and feature detail preservation. These attributes influenced our choice to adopt YOLOv7's design.

Given the relatively low complexity of laparoscopic videos, we halved the number of feature channels in YOLOv7 for a streamlined structure. In the neck section, we employed the RepBlock for feature extraction and used the CBR module for down-sampling to enhance inference speed. The integration of RepBlock within a feature pyramid network (FPN)³⁹ design in the neck section is what gives our proposed model its name, YOLOv7-RepFPN.

2.3.1 | Determination of Feature Channel Numbers

As mentioned, the relatively low complexity of laparoscopic image data allows lightweight models with faster speeds to potentially achieve high accuracy in laparoscopic tool detection. Therefore, we first explored the most straightforward way to make the model lightweight, by adjusting the number of feature channels to achieve a more streamlined structure.

Recently-released versions of YOLO models, like YOLOv5¹⁶ and YOLOv6⁴⁰, have introduced parameters named Depth Multiple (DM) and Width Multiple (WM), respectively. depth of the model can be modulated with DM by altering the number of convolutional modules within certain blocks of the model, while number of feature channels can be adjusted with WM. By manipulating DM and WM, small, medium, and large-scale models of varied amount of parameters and computational loads can

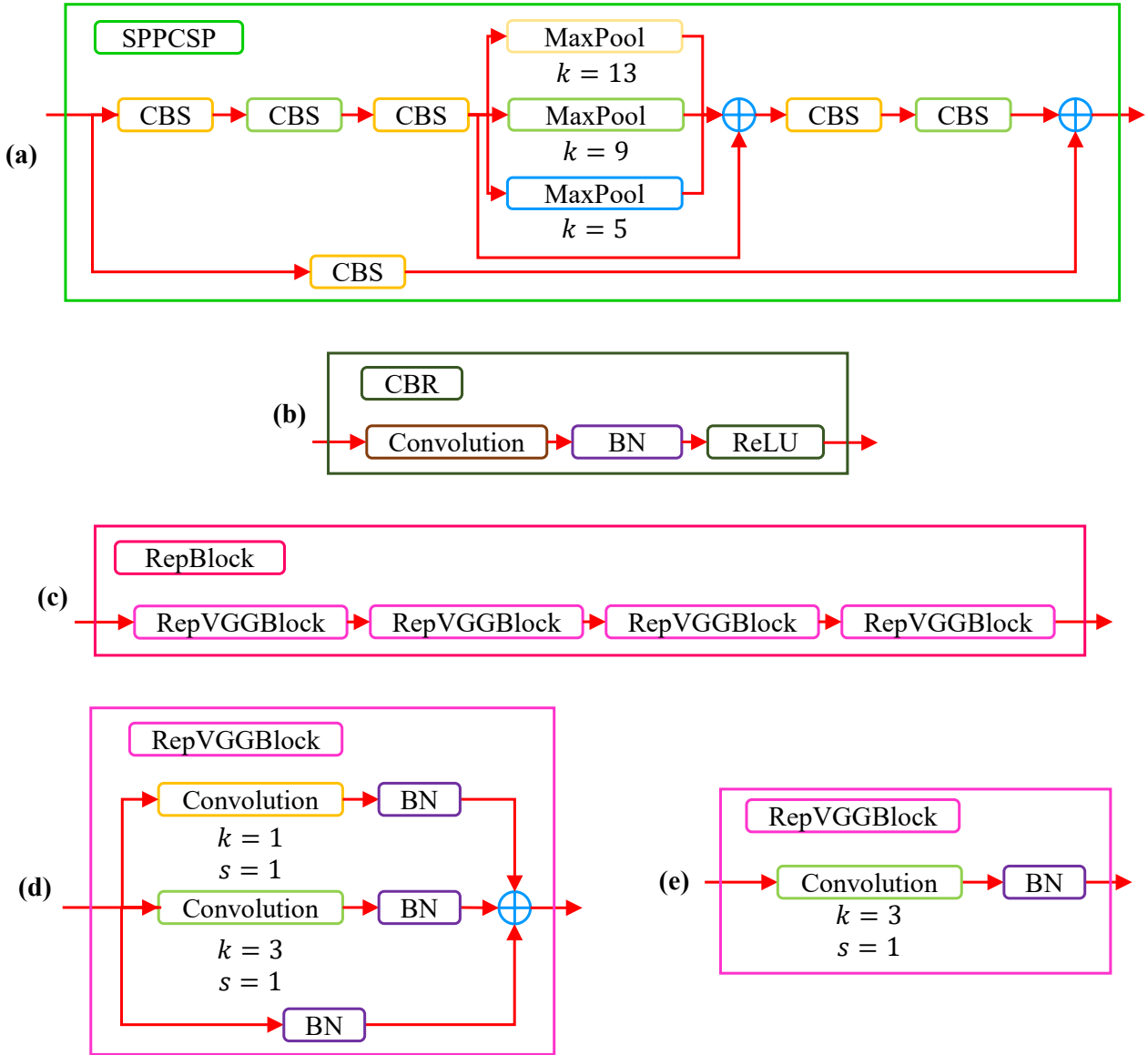


FIGURE 4 Structures of key modules in the neck section of YOLOv7-RepFPN. (a): Structure of SPPCSP module, k represents kernel sizes of the max pooling layer. (b): Structure of CBR module. ReLU denotes ReLU (rectified linear unit) activation function. (c): Structure of RepBlock. (d) and (e): RepVGGBlock structure for training and inference phase, color of the convolutional layer corresponds to its parameters, k and s represent the kernel size and stride of the convolutional layer, respectively.

be derived. YOLOv7 maintains DM and WM, but only provides a large-scale model. In this study, we adopted the settings of WM and DM from YOLOv5 and YOLOv6, We built a small-scale model (YOLOv7s) by reducing YOLOv7's feature channels by half, and a medium-scale model (YOLOv7m) by decreasing YOLOv7's feature channels to three quarters of the original amount. Our decision to halve the feature channels was confirmed by experimental comparisons with YOLOv7 (see Table 1).

2.3.2 | Design of the Neck section

Figure 4 illustrate the key modules within the neck section of our YOLOv7-RepFPN, which is designed to improve inference speed. The SPPCSP module inherited from YOLOv7³⁰ is shown in Figure 4(a). It employs spatial pyramid pooling (SPP) to handle various target sizes, with "CSP" indicating the grouping strategy from the cross-stage partial (CSP)⁴¹ network.

The CBR module used for downsampling in the neck section is displayed in Figure 4(b). This module is a sequence of a convolution layer, a batch normalization layer, and a ReLU activation function. It replaced DDS module, the only downsampling module in YOLOv7, in neck section of our YOLOv7-RepFPN. The CBR module omits pooling operations, has fewer convolution layers, and opts for the less computationally demanding ReLU for non-linear activation, making it a faster module for detection compared to the DDS module.

Figure 4(c) presents the RepBlock³⁴, used for feature extraction in the neck section. This block is constructed sequentially from RepVGGBlocks, borrowed from the structure of the RepVGG network⁴². Unlike ELAN-B, or the ELAN-B-like feature extraction module in neck section of YOLOv7, the feed-forward structure of RepBlock makes model scaling adjustments possible through the DM. The block reduces depth of the model by limiting the number of RepVGGBlocks to four.

The structures of RepVGGBlock during training and inference stages are outlined in Figure 4(d) and Figure 4(e), respectively. The term "Rep" signifies reparameterization, a technique adopted by the module to harmonize training and inference stages. During training, RepVGGBlock uses a multi-branch structure for diverse feature learning and robust gradient propagation. During inference, it re-parameterizes into a streamlined single-branch structure for computational efficiency and swift inference speed.

Both RepBlock and the combination of RepBlock and CBR module restructure into a design akin to the VGG network⁴³ during inference. This inference structure consists solely of convolutional layers with a 3x3 kernel size and ReLU activation functions. This straightforward, feed-forward architecture greatly enhances inference speed.

2.4 | Loss Function for Bounding Box Regression

As previously mentioned, laparoscopic images contain many tilted laparoscopic tools. This poses a higher demand on the accuracy of bounding boxes used for tool localization in detection tasks. Compared to targets in horizontal or vertical positions, the length and width of bounding boxes for the same targets in tilted positions can exhibit noticeable changes. We first mentioned why CIoU³⁷ had a hard time handling such cases by its derivatives and then introduced focal EIou³⁵ for solving this problem.

CIoU loss function considers IoU (intersection over union) of the predicting bounding box (PBB) and the ground truth bounding box (GBB), distance between centers of PBB and GBB, and aspect ratio of PBB and GBB. CIoU loss is expressed as follows:

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(\mathbf{b}, \mathbf{b}^{gt})}{c^2} + \alpha v, \quad (1)$$

In the given equation, \mathbf{b} and \mathbf{b}^{gt} denote the center points of PBB and GBB respectively. The term $\rho^2(\mathbf{b}, \mathbf{b}^{gt})$ represents the squared Euclidean distance between the center points of PBB and GBB, while c corresponds to the diagonal distance within the minimum bounding box enclosing both PBB and GBB. The term αv is a penalty term concentrating on the aspect ratios of PBB and GBB. Here, v serves to measure the consistency of these aspect ratios, and it is defined as:

$$v = \frac{4}{\pi^2} \left(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right)^2, \quad (2)$$

where w^{gt} , h^{gt} , w , h stands for width and height of GBB and PBB, α is defined on basis of v by the following equation:

$$\alpha = \frac{v}{(1 - IoU) + v}. \quad (3)$$

The limitation of CIoU stems from the definition of v . Taking derivatives of v with respect to w and h , the following equation is reached:

$$\begin{aligned} \frac{\partial v}{\partial w} &= \frac{8}{\pi^2} \left(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right) \frac{h}{w^2 + h^2}, \\ \frac{\partial v}{\partial h} &= -\frac{8}{\pi^2} \left(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right) \frac{w}{w^2 + h^2}. \end{aligned} \quad (4)$$

Based on the equation, it can be derived that $\frac{\partial v}{\partial w} = -\frac{h}{w} \frac{\partial v}{\partial h}$, indicating that $\frac{\partial v}{\partial w}$ and $\frac{\partial v}{\partial h}$ have opposite signs for the process of BBR. Due to this characteristic, w and h cannot simultaneously increase or decrease during the training process. This influences training efficiency, as well as detection accuracy of the model.

EIoU³⁵ loss was proposed considering the aforementioned limitation of CIoU, its penalty terms is based on width and height of PBB. EIoU is defined by:

$$L_{EIoU} = 1 - IoU + \frac{\rho^2(\mathbf{b}, \mathbf{b}^{gt})}{c^2} + \frac{\rho^2(w, w^{gt})}{c_w^2} + \frac{\rho^2(h, h^{gt})}{c_h^2}, \quad (5)$$

where c_w and c_h denote width and height of the minimum bounding box which encloses both PBB and GBB, $\rho^2(w, w^{gt})$ and $\rho^2(h, h^{gt})$ respectively denote the squared differences in width and height between PBB and GBB. Given that the penalty terms considering the w and h are independent of each other, for training process, w and h can increase or decrease simultaneously. The EIoU loss function is more flexible in terms of bounding box shape compared to the CIoU loss function. Thus, for laparoscopic tool detection, with the frequent appearance of tilted tools, using the EIoU loss function for BBR can improve localization accuracy.

For bounding box regression, to give higher weights to high-quality anchors, focal EIoU is ultimately selected as the loss function. Its definition is as follows:

$$L_{Focal-EIoU} = IoU^{\frac{1}{2}} L_{EIoU}. \quad (6)$$

3 | EXPERIMENT

3.1 | Experimental Settings and Dataset

Dataset used in this study was manually organized on basis of EndoVis17³³, YOLO-format labels were obtained by taking the minimum enclosing box from the results of laparoscopic tool segmentation. The EndoVis17 dataset encompasses 7 categories of laparoscopic tools across ten cases (1-10), with each case containing 300 images. Our dataset division adhered to the following principles:

- Training and validation datasets include images from the same case, but the test dataset does not share cases with either training or validation sets, ensuring the validity of experimental results.
- The training dataset represents all seven laparoscopic tool categories.
- The test dataset encompasses as many tool categories as possible.

Given these principles, 1,890 images from Cases 1-6 and 8 were allocated for training, and the subsequent 210 images from these cases were designated for validation. The test dataset comprises 900 images from Cases 7, 9, and 10. All images were resized to 640x640 pixels.

Our training was conducted over 300 epochs using Adam as the optimizer and an initial learning rate of 0.001. To mitigate overfitting's potential influence on detection accuracy, we utilized the optimal model from the training epochs following the strategy of YOLOv7³⁰. The model was trained using a single NVIDIA Tesla V100 GPU and tested on the NVIDIA Jetson AGX Orin 32 GB platform. To substantiate the effectiveness of our proposed model, we performed comparative experiments with different scales of the YOLOv5, YOLOv6 and YOLOv7 models. Given that the detection model inputs data frame-by-frame in a clinical setting, we carried out detection on the test dataset on a frame-by-frame basis in this study.

We employed several evaluation metrics, including the number of parameters, GFLOPs (Giga Floating Point Operations Per Second), mAP₅₀ (mean Average Precision when IoU threshold is 0.50), mAP_{50:95} (average mAP with IoU threshold from 0.50 to 0.95), and FPS. The number of parameters and GFLOPs broadly depict the model structure's scale and complexity, while mAP₅₀ and mAP_{50:95} are prevalent indicators for detection accuracy. FPS provides an indication of the model's inference speed.

3.2 | Result Analysis

Table 1 presents the experimental results of different models. The information in the table reveals that:

- YOLOv7-tiny and YOLOv6-n are ultra-small-sized models designed for highly resource-constrained scenarios. They excel in inference speed, achieving around 70 FPS. While YOLOv7-RepFPN is slower than the fastest model, YOLOv7-tiny, by 9 FPS, it still maintains a rapid inference speed and offers a significant advantage in terms of accuracy. Relative to

TABLE 1 Quantitative evaluation of comparative experiments. The abbreviations n, s, m, l, x stand for nano, small, medium, large, and extra-large, respectively, indicating scale of the model.

| Model | Parameters/M | GFLOPs | mAP ₅₀ | mAP _{50:95} | FPS ^a |
|---------------|--------------|-------------|-------------------|----------------------|-------------------|
| YOLOv5-s | 7.03 | 15.8 | 0.845 | 0.662 | 64.1 ± 0.3 |
| YOLOv5-m | 20.88 | 47.9 | 0.858 | 0.690 | 52.6 ± 0.3 |
| YOLOv5-l | 46.14 | 107.7 | 0.865 | 0.701 | 38.5 ± 0.2 |
| YOLOv5-x | 86.21 | 203.9 | 0.884 | 0.696 | 21.3 ± 0.1 |
| YOLOv6-n | 4.63 | 11.3 | 0.794 | 0.585 | 69.7 ± 0.5 |
| YOLOv6-s | 18.50 | 45.2 | 0.857 | 0.667 | 57.5 ± 0.2 |
| YOLOv6-m | 34.81 | 85.6 | 0.862 | 0.658 | 44.2 ± 0.1 |
| YOLOv6-l | 59.54 | 150.5 | 0.849 | 0.653 | 25.3 ± 0.1 |
| YOLOv7-tiny | 6.02 | 13.1 | 0.799 | 0.592 | 71.9 ± 0.7 |
| YOLOv7s | 9.15 | 26.0 | 0.849 | 0.654 | 53.5 ± 0.2 |
| YOLOv7m | 20.96 | 59.5 | 0.864 | 0.668 | 47.8 ± 0.2 |
| YOLOv7 | 36.51 | 103.3 | 0.893 | 0.699 | 41.8 ± 0.2 |
| YOLOv7-RepFPN | 10.58 | 30.7 | 0.882 | 0.681 | 62.9 ± 0.3 |

^aFor the FPS metric, we documented the mean and standard deviation from ten separate tests.

YOLOv7-tiny, the top performer in mAP metrics among ultra-compact models, our YOLOv7-RepFPN model enhances mAP₅₀ by 0.083 and mAP_{50:95} by 0.089.

- Small-scale models offer a balance between detection accuracy and speed, improving upon the accuracy of ultra-small-scale models while preserving high inference speed. For instance, YOLOv7s boosts mAP₅₀ by 0.05 and mAP_{50:95} by 0.062 compared to YOLOv7-tiny. While maintaining the same detection speed, YOLOv7-RepFPN outperforms these small-scaled models in terms of accuracy. Against YOLOv5-s, the fastest small-scale model, YOLOv7-RepFPN improves mAP₅₀ and mAP_{50:95} by 0.037 and 0.019, with only a slight decrease in inference speed by 1.2 FPS. When compared to YOLOv6-s, the highest mAP small-scale model, YOLOv7-RepFPN enhances mAP₅₀ and mAP_{50:95} by 0.025 and 0.014, and speed by 5.4 FPS.
- With increasing model scale, both YOLOv5 and YOLOv7 show improved detection accuracy, but inference speeds substantially decline. As an example, the medium-scale YOLOv5-m model enhances mAP₅₀ and mAP_{50:95} by 0.013 and 0.028 respectively over the smaller YOLOv5-s, but suffers a speed reduction of 11.5 FPS. Furthermore, the larger YOLOv5-l model improves upon YOLOv5-m by increasing mAP₅₀ and mAP_{50:95} by 0.007 and 0.011 respectively, but decreases speed by 14.1 FPS. When scaling up to the largest YOLOv5-x model, mAP₅₀ is increased by 0.019, while mAP_{50:95} drops by 0.005, and speed falls to 21.3 FPS—less than a third of YOLOv5-s. Against YOLOv7, the best-performing large-scale model in speed and accuracy, YOLOv7-RepFPN enhances speed by 21.1 FPS while maintaining similar mAP metrics, with minor decreases of 0.011 in mAP₅₀ and 0.018 in mAP_{50:95}.

3.3 | Ablation Studies

Ablation studies were performed on our architectural and optimization modifications to YOLOv7, with the results presented in Table 2. For simplicity in this section, the YOLOv7 model with halved feature channels is referred to as the YOLOv7s model. It can be concluded from Table 2 that:

TABLE 2 Ablation study for our proposed revisions (✓ for corresponding revision). Effectiveness of each combination of the revisions are also evaluated.

| Halved Feature Channels | RepBlock | Focal EIoU | Parameters/M | GFLOPs | mAP ₅₀ | mAP _{50:95} | FPS ^a |
|-------------------------|----------|------------|--------------|--------|-------------------|----------------------|------------------|
| | | | 36.51 | 103.3 | 0.893 | 0.699 | 41.8 ± 0.2 |
| ✓ | | | 9.15 | 26.0 | 0.849 | 0.654 | 53.5 ± 0.2 |
| ✓ | ✓ | | 10.58 | 30.7 | 0.874 | 0.660 | 62.9 ± 0.3 |
| ✓ | | ✓ | 9.15 | 26.0 | 0.855 | 0.667 | 53.5 ± 0.2 |
| ✓ | ✓ | ✓ | 10.58 | 30.7 | 0.882 | 0.681 | 62.9 ± 0.3 |

^aFor the FPS metric, we documented the mean and standard deviation from ten separate tests.

- Halving the feature channels for lightweight modification of the model resulted in a substantial reduction in both the number of parameters and FLOPs. Compared with YOLOv7, inference speed of YOLOv7s increased by 11.7 FPS. However, a decrease of 0.044 in both mAP_{50} and a decrease of 0.045 in $mAP_{50:95}$ was also observed. This decrease could be due to the factor that the reduction in feature channels may lead to insufficient feature extraction.
- To simplify the model structure and improve inference speed, we introduced RepBlock in the neck section for feature extraction. Compared to YOLOv7s, our YOLOv7-RepFPN model with RepBlock not only improved inference speed by 9.4 FPS but also increased the mAP_{50} and $mAP_{50:95}$ metrics by 0.025 and 0.006 respectively.
- With the goal of improving detection accuracy, the focal EIoU loss function was used for bounding box regression. Whether applied to the YOLOv7s or YOLOv7-RepFPN, use of EIoU loss function led to improvements in the mAP metrics. For YOLOv7s, application of EIoU resulted in a 0.006 increase in mAP_{50} and a 0.013 increase in $mAP_{50:95}$. For YOLOv7-RepFPN, use of EIoU resulted in a 0.018 increase in mAP_{50} and a 0.021 increase in $mAP_{50:95}$.

3.4 | Discussion

Our YOLOv7-RepFPN model strikes a balance between speed and detection accuracy, achieving an inference speed of 62.9 FPS for frame-by-frame laparoscopic tool detection. This speed surpasses twice the real-time requirement for laparoscopic tool detection, while maintaining precision close to that of large-scale YOLO models. However, limitations exist. Analysis of the results presented in Table 1 indicates that while our model is optimized for speed, it registers a lower mAP than the large-scale YOLOv7, baseline of the study. This gap tends to widen with an increase in the IoU threshold.

Figure 5 visualizes tool detection and failure instances, also reveals limitation that the model struggles with localization when separate tips of the same target appear at frame edges. It may only detect one larger tip or identify two tips as distinct targets of the same category, as shown in Figure 5(b). This issue is also present in the original YOLOv7, YOLOv5, and YOLOv6 models under the same experimental conditions and dataset.

The limitations above might be related to the design of the YOLOv7 detection head. YOLOv7 uses a simple, non-decoupled classification head to favor speed. Reducing the number of feature channels, hence feature information, could contribute to decreased detection accuracy. Additionally, the non-decoupled head, treating classification and localization as one task, might

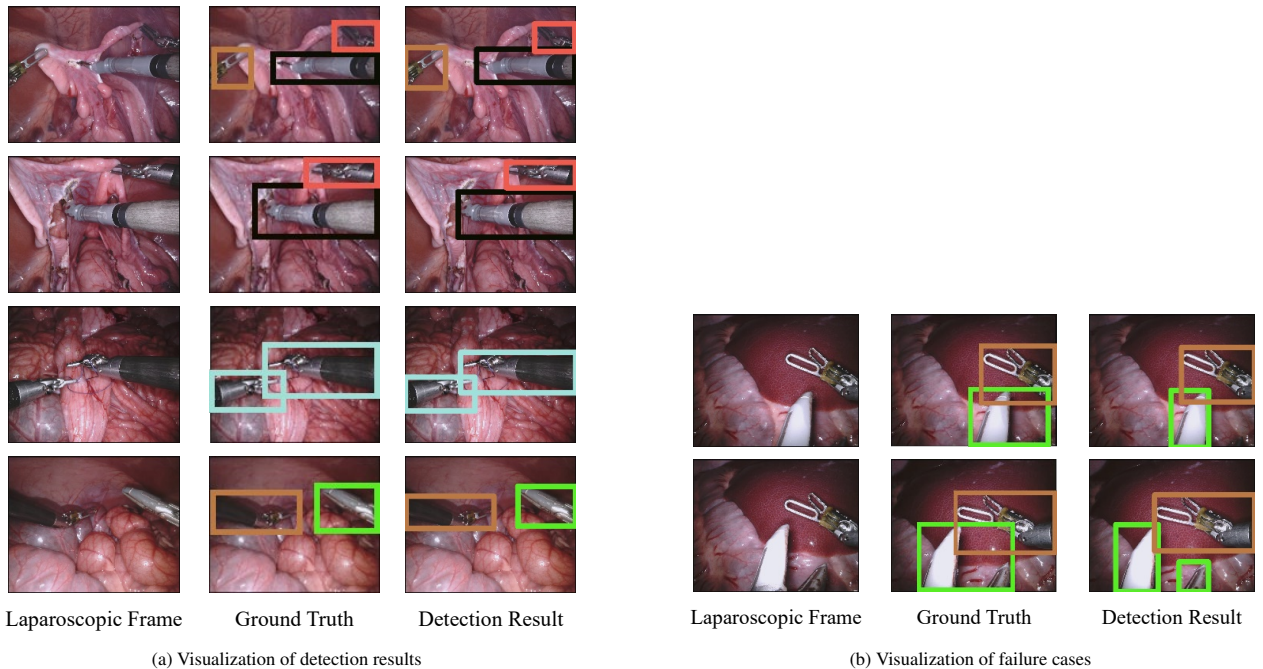


FIGURE 5 Visualization of some detection results as well as failure cases. Different colors of bounding boxes suggests different categories of laparoscopic tools.

impact detection precision. Lastly, the oversight of tool tips at image edges might be due to YOLO's inability to extract and merge more scaled feature information.

4 | CONCLUSIONS

Our study aimed to increase the speed of laparoscopic tool detection on embedded devices without sacrificing accuracy. By optimizing the architecture of YOLOv7, we developed YOLOv7-RepFPN. The model exhibited a detection accuracy of mAP_{50} of 88.2%, while achieving an inference speed of 62.9 FPS, which substantially exceeds real-time requirements.

Although YOLOv7-RepFPN is proficient in its core objectives, it exhibits limitations in specific scenarios, such as detecting tool tips at the edge area of images, this limitation is also discovered when analyzing detection results of other YOLO models. To address these issues, incorporating a decoupled detection head or modules based on attention mechanisms is a potential avenue for future work.

The speed advantage of YOLOv7-RepFPN affords the possibility of further architectural enhancements. The increased efficiency enables the incorporation of more advanced modules, like attention mechanisms, without sacrificing real-time performance. This is in line with our original research goals and paves the way for extended functionalities such as segmentation, localization, and tool pose estimation.

ACKNOWLEDGMENTS

Parts of this research were supported by the MEXT/JSPS KAKENHI grant numbers 21K19898, 26108006 and 17H00867, JST Moonshot R&D grant number JPMJMS2033, JST CREST grant number JPMJCR20D5, as well as JST SPRING, grant number JPMJSP2125. The author (Y.L.) would like to take this opportunity to thank the "Interdisciplinary Frontier Next-Generation Researcher Program of the Tokai Higher Education and Research System."

CONFLICT OF INTEREST

The authors declare no potential conflict of interests.

REFERENCES

- Jacob BP, Davis SS, Dakin GF, et al. The SAGES Manual of Hernia Surgery. In: Cambridge International Law Journal. 2019.
- Cundy TP, Harling L, Hughes-Hallett A, et al. Meta-analysis of robot-assisted vs conventional laparoscopic and open pyeloplasty in children. *BJU international*. 2014;114(4):582–594.
- Allan M, Ourselin S, Thompson S, Hawkes DJ, Kelly J, Stoyanov D. Toward detection and localization of instruments in minimally invasive surgery. *IEEE Transactions on Biomedical Engineering*. 2012;60(4):1050–1058.
- Ryu J, Choi J, Kim H. Endoscopic vision-based tracking of multiple surgical instruments during robot-assisted surgery. *Artificial organs*. 2013;37(1):107–112.
- Alshirbaji TA, Jalal NA, Docherty PD, Neumuth T, Möller K. A deep learning spatial-temporal framework for detecting surgical tools in laparoscopic videos. *Biomedical Signal Processing and Control*. 2021;68:102801.
- Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*. 2015;28.
- He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proceedings of the IEEE international conference on computer vision. IEEE. 2017:2961–2969.
- Redmon J, Farhadi A. YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*. 2018.
- Bouget D, Allan M, Stoyanov D, Jannin P. Vision-based and marker-less surgical tool detection and tracking: a review of the literature. *Medical image analysis*. 2017;35:633–654.
- Twinanda AP, Shehata S, Mutter D, Marescaux J, De Mathelin M, Padoy N. EndoNet: a deep architecture for recognition tasks on laparoscopic videos. *IEEE transactions on medical imaging*. 2016;36(1):86–97.
- Sarikaya D, Corso JJ, Guru KA. Detection and Localization of Robotic Tools in Robot-Assisted Surgery Videos Using Deep Neural Networks for Region Proposal and Detection. *IEEE Transactions on Medical Imaging*. 2017;36(7):1542–1549.
- Hasan MK, Calvet L, Rabbani N, Bartoli A. Detection, segmentation, and 3D pose estimation of surgical tools using convolutional neural networks and algebraic geometry. *Medical Image Analysis*. 2021;70:101994.
- Kanakatte A, Ramaswamy A, Gubbi J, Ghose A, Purushothaman B. Surgical tool segmentation and localization using spatio-temporal deep network. In: 2020 42nd annual international conference of the IEEE engineering in medicine & biology society (EMBC). IEEE. 2020:1658–1661.
- Namazi B, Sankaranarayanan G, Devarajan V. Attention-Based Surgical Phase Boundaries Detection in Laparoscopic Videos. In: 2019 International Conference on Computational Science and Computational Intelligence (CSCI). 2019:577–583.
- Nwoye CI, Yu T, Gonzalez C, et al. Rendezvous: Attention mechanisms for the recognition of surgical action triplets in endoscopic videos. *Medical Image Analysis*. 2022;78:1024–1033.
- Ultralytics . YOLOv5. <https://github.com/ultralytics/yolov5>; 2021.
- Bolya D, Zhou C, Xiao F, Lee YJ. YOLACT: Real-time instance segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019:9157–9166.

18. Zou X, Liu W, Wang J, Tao R, Zheng G. ARST: auto-regressive surgical transformer for phase recognition from laparoscopic videos. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*. 2022:1–7.
19. Cheng X, Zhong Y, Harandi M, Drummond T, Wang Z, Ge Z. Deep laparoscopic stereo matching with transformers. In: Proceedings of Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference. Springer. 2022:464–474.
20. Maier-Hein L, Jannin P. Surgical data science for next-generation interventions. *Nature Biomedical Engineering*. 2017;1(9):691–696.
21. Hashimoto DA, Rosman G, Rus D, Meireles OR. Artificial intelligence in surgery: promises and perils. *Annals of surgery*. 2018;268(1):70–76.
22. Rieke N, Hancox J, Li W, et al. The future of digital health with federated learning. *NPJ digital medicine*. 2020;3(1):119.
23. Shi W, Cao J, Zhang Q, Li Y, Xu L. Edge computing: Vision and challenges. *IEEE internet of things journal*. 2016;3(5):637–646.
24. Ciaparrone G, Bardozzo F, Priscoli MD, Kallewaard JL, Zuluaga MR, Tagliaferri R. A comparative analysis of multi-backbone Mask R-CNN for surgical tools detection. In: 2020 International Joint Conference on Neural Networks (IJCNN). IEEE. 2020:1–8.
25. Colleoni E, Moccia S, Du X, De Momi E, Stoyanov D. Deep learning based robotic tool detection and articulation estimation with spatio-temporal layers. *IEEE Robotics and Automation Letters*. 2019;4(3):2714–2721.
26. Liu Y, Zhao Z, Chang F, Hu S. An anchor-free convolutional neural network for real-time surgical tool detection in robot-assisted surgery. *IEEE Access*. 2020;8:78193–78201.
27. Koskinen J, Torkamani-Azar M, Hussein A, Huotarinen A, Bednarik R. Automated tool detection with deep learning for monitoring kinematics and eye-hand coordination in microsurgery. *Computers in Biology and Medicine*. 2022;141:105–121.
28. Zhang B, Wang SS, Dong L, Chen P. Surgical Tools Detection Based on Modulated Anchoring Network in Laparoscopic Videos. *IEEE Access*. 2020;8:23748–23758.
29. Yang Y, Zhao Z, Shi P, Hu S. An efficient one-stage detector for real-time surgical tools detection in robot-assisted surgery. In: Proceedings of Medical Image Understanding and Analysis: 25th Annual Conference, MIUA 2021. Springer. 2021:18–29.
30. Wang CY, Bochkovskiy A, Liao HYM. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*. 2022.
31. Lin TY, Maire M, Belongie S, et al. Microsoft COCO: Common objects in context. In: Proceedings of Computer Vision–ECCV 2014: 13th European Conference. Springer. 2014:740–755.
32. Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge. *International journal of computer vision*. 2015;115:211–252.
33. Allan M, Shvets A, Kurmann T, et al. 2017 robotic instrument segmentation challenge. *arXiv preprint arXiv:1902.06426*. 2019.
34. Weng K, Chu X, Xu X, Huang J, Wei X. EfficientRep: An Efficient Repvgg-style ConvNets with Hardware-aware Neural Network Design. *arXiv preprint arXiv:2302.00386*. 2023.
35. Zhang YF, Ren W, Zhang Z, Jia Z, Wang L, Tan T. Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing*. 2022;506:146–157.
36. Yang C, Zhao Z, Hu S. Image-based laparoscopic tool detection and tracking using convolutional neural networks: a review of the literature. *Computer Assisted Surgery*. 2020;25(1):15–28.
37. Zheng Z, Wang P, Liu W, Li J, Ye R, Ren D. Distance-IoU loss: Faster and better learning for bounding box regression. In: Proceedings of the AAAI conference on artificial intelligence. 2020:12993–13000.
38. Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*. 2020.
39. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017:2117–2125.
40. Li C, Li L, Jiang H, et al. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*. 2022.
41. Wang CY, Liao HYM, Wu YH, Chen PY, Hsieh JW, Yeh IH. CSPNet: A new backbone that can enhance learning capability of CNN. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2020:390–391.
42. Ding X, Zhang X, Ma N, Han J, Ding G, Sun J. RepVGG: Making VGG-style ConvNets great again. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021:13733–13742.
43. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. 2014.