Automatic detection of indoor occupancy based on improved YOLOv5 model

Chao Wang¹ · Yunchu Zhang¹ · Yanfei Zhou¹ · Shaohan Sun¹ · Hanyuan Zhang¹ · Yepeng Wang²,³

Received: 16 February 2022 / Accepted: 12 August 2022 / Published online: 2 September 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

Abstract
Indoor occupancy detection is essential for energy efficiency control and Coronavirus Disease 2019 traceability. The number and location of people can be accurately identified and determined through classroom surveillance video analysis. This information is used to manage environmental equipment such as HVAC and lighting systems to reduce energy use. However, the mainstream one-stage YOLO algorithm still uses an anchor-based mechanism and couples detection heads to predict. This results in slow model convergence and poor detection performance for densely occluded targets. Therefore, this paper proposed a novel decoupled anchor-free VariFocal loss convolutional network algorithm DFV-YOLOv5 for occupancy detection to tackle these problems. The proposed method uses the YOLOv5 algorithm as a baseline. It uses the anchor-free mechanism to reduce the number of design parameters needing heuristic tuning. Afterwards, to reduce the coupling of the model, speed up the model’s convergence ability, and improve the model detection performance, the detection head is decoupled based on the YOLOv5 model. It can resolve the conflict between classification and regression tasks. In addition, we use the VariFocal loss to assign more weights to difficult data points to optimize the class imbalance problem and use the training target \( q \) to measure positive samples, treating positive and negative samples asymmetrically. The total loss function is redesigned, the \( L_1 \) loss is increased, and the ablation experiment verifies the effect of the improved loss. By applying a hybrid activation function of the sigmoid linear unit and rectified linear unit, we improved the model’s nonlinear representation and reduced the model’s inference time. Finally, a classroom dataset was constructed to validate the occupancy detection performance of the model. The proposed model was compared with mainstream target detection models regarding average mean precision, memory allocation, execution time, and the number of parameters on the VOC2012, CrowdHuman and self-built datasets. The experimental results show that the method significantly improves the detection accuracy and robustness, shortens the inference time, and proves the practicality of the algorithm in occupancy detection compared with the mainstream target detection model and related variants of the model.

Keywords Deep learning · YOLOv5 · Occupancy detection · Dense object detection

1 Introduction

With improving people’s living standards, the proportion of building energy consumption in total social energy consumption is increasing. For example, in the USA, annual building energy consumption accounts for about 40% of society’s total energy consumption [1], while in China, the proportion is about 27.5%. Heating, ventilation, and air conditioning (HVAC) and lighting systems consume most of the daily building energy consumption (more than 50%), of which lighting energy consumption is 20% to 30% of the total building electricity consumption, so it has excellent energy-saving potential [2].

As the prominent contradiction between energy supply and demand becomes more and more apparent, energy saving has become a critical topic worldwide. One of the leading causes of energy wastage in buildings is the control strategies used in traditional HVAC and lighting systems, which do not respond promptly to the dynamic changes in the number and distribution of people. In recent years, the method based on images and video is a very effective method to detect the occupancy of buildings and identify the activities of people. This method uses computer vision and digital image processing technology to process the
video image to obtain information such as the number, location, and even people’s behaviour in the video. Video analysis is usually done by detecting head, face, body contours, or body movements [3]. One energy-saving way is to use a vision-based closed-loop feedback control strategy with real-time information. The number of people in the building and their distribution in different locations are utilized as inputs to regulate the lighting system. When it is found that the number of people in an area is decreasing, some lights should be turned off automatically to reduce unnecessary energy consumption. Past studies have shown that good occupancy detection can bring up to 50% energy-saving potential for lighting [4].

The object detection algorithm based on image or video is mainly divided into traditional image processing technology and deep learning vision algorithm. Traditional vision techniques such as background subtraction, Histogram of Oriented Gradient (HOG) [5], and support vector machine (SVM) [6] were employed for vision-based person detection and counting. Based on the traditional vision technology, many researchers have also done much work on occupancy detection. Zaveri et al. [7] and Li [8] installed a camera in the corridor and used the background subtraction method to detect the number of people in the building. Liu [9] and others use multi-cameras and cascade algorithms to improve the accuracy of head detection. Yang [10] and others evaluate the performance of occupancy counting by using PTZ cameras to focus images on detecting faces in the classroom. Sun [11] et al. achieved good detection results by combining motion detection and static estimation using entrance/exit and indoor cameras by fusing motion detection and the head detection model FCHD, and finally by Kalman filtering and occupancy frequency histogram (OFH). Because of the low accuracy of occupancy detection, the traditional vision algorithm has not been widely used in engineering.

Deep learning methods have been widely used in occupancy detection since 2014. For its efficient calculation and automatic learning of numerous features, the convolutional neural network (CNN) has been widely used in the field of computer vision. Taking advantage of the rapid development of CNN, a lot of work has been put into different engineering applications [12–15]. At the same time, some common benchmark datasets designed for natural scene images, such as MS COCO [16] and Pascal VOC [17] greatly promote the development of target detection applications. According to the different detection stages, the target detection model based on CNN can be divided into one-stage and two-stage models. Two-stage detectors include VFNet [18], CenterNet2 [19], Fast R-CNN [20], and Faster R-CNN [21]. The two-stage model includes two processes: candidate region selection and feature extraction, but because the detection is divided into numerous stages and there is redundancy in the calculation, real-time performance is not possible. To solve this problem, the YOLO [22–24] (You Only Look Once) series, SSD [25] and other one-stage object detection methods were proposed. These methods have greatly improved the recognition speed, but the positioning accuracy has decreased. With the emergence of YOLOv4 [26] and YOLOv5 [27, 28] detection algorithms, they gradually surpass the two-stage detector in performance and become the state-of-the-art (SOTA) in the one-stage target detector can be anchor-based mechanism and anchor-free mechanism, respectively. The anchor-based detectors include the YOLO series, YOLOv4, Scaled-YOLOv4 [29], YOLOv5, etc. Anchor-free detectors include CenterNet [30], YOLOX [31], RepPoints [32].

Deep learning-based object detection techniques are used in many real-life applications, such as autonomous driving, crime prevention, and engineering inspection [33–35]. Many advanced object detection techniques have been introduced in the building sector to identify the number of people in a building and applied to building energy control. Conti et al. [36] proposed a head detection system based on CNN, a common deep learning algorithm. Using a multi-level boosting classifier (CNN+HOG+k-means), Zou et al. [4] suggested a method for detecting head occupancy numbers. The combined use of deep learning and classical vision algorithms yielded a 95.3% accuracy for occupancy measurement in the experiments. Tien et al. [37] implemented occupancy activity detection using cameras based on convolutional neural networks. Detection and prediction of activities such as sitting, standing, walking and snoozing in buildings was achieved with an average detection accuracy of 80.62%. Many academics employed the YOLO technique to identify occupancy after 2018. Meng et al. [38] used the YOLO algorithm to create a CNN-based end-to-end model for dynamic occupancy load prediction in building spaces, allowing for real-time occupancy load estimation. Mutis et al. [39] implements a multi-stream deep neural network to identify human activities and uses the YOLOv3 deep neural network for object detection to estimate occupancy count in a room. Choi et al. [40] employed the YOLOv5 model to assess the performance of vision-based occupancy calculation methods in two offices, as well as a questionaire to gauge user approval of the technology. However, the detection of dense targets in large indoor scenes is greatly influenced by light changes at different times, inter-target occlusion, and scene depth, leading to false detections and missed detections, which affects the detection effect. Most existing occupancy detection algorithms use a one-stage YOLO algorithm, which can meet the real-time requirements. Still, because of the use of anchor-based mechanism (ABM) in this model, the model's
generalization ability is poor, and there are too many super-
parameters. The use of coupled prediction heads limits the
classification of the model and the accuracy of regression
tasks. In the occupation detection task, the significant dif-
fERENCE between the number of positive and negative
samples leads to the imbalance between positive and neg-
ative examples. However, only using BCE and Focal loss
cannot solve this problem, so the model’s performance is
limited.

In summary, to address the problems of current occu-
pancy detection models based on one-stage target detec-
tors, this paper proposes a new anchor-free \cite{41} decoupled
detector head with VariFocal loss convolution model DFV-
YOLOv5 for occupancy detection inside buildings. The
main contributions of this paper are as follows:

- The ABM is replaced by the anchor-free mechanism
(AFM), which first reduces the original prediction box
from three to only one point per pixel point, specifies
the centre of each object as a positive sample and
predefines a scale range, and then assigns a Feature
Pyramid Network (FPN) level to each object. AFM
eliminates the lack of generalization of samples due to
the clustering of anchor points by ABM. The problem
of imposing additional hyperparameters on the model is
solved by not having to obtain a priori boxes before
training.

- To eliminate the conflict between the classification task
and the regression task caused by detector head
coupling, $3 \times 3$ convolution and $1 \times 1$ convolution
are used to decouple the detector head. For each layer
feature of FPN, $1 \times 1$ Conv layer is used to reduce the
feature channel to 256. Then the detector head is
decoupled into two parallel $3 \times 3$ Conv layer branches,
one for the classification task and the other for the
regression task branch, and then add the IoU branch to
the regression branch. The use of decoupled detection
head can improve the model’s convergence speed and
detection performance.

- This study uses VariFocal loss to provide more weight
for challenging data points to optimize the class
imbalance problem. Positive samples are measured
using a training target $q$, Positive and negative samples
are processed asymmetrically to improve detection of
densely occluded targets.

- Due to the lack of publicly available occupancy
detection datasets, a classroom dataset was constructed
to capture and annotate the occupancy of classroom
personnel using multiple viewpoint cameras, which
were used to evaluate the performance of the detection
model.

- Inspired by recent research results, this paper investi-
gates the transformer \cite{42} attention mechanism, and the

Bidirectional Feature Pyramid Network (BiFPN) \cite{43}
module for multi-scale feature fusion on improving the
network structure, improving the feature representation
of shallow networks, and on detection performance,
using the hybrid activation functions SiLU and ReLu to
investigate their effects on model detection speed.

- To fully validate the performance of the proposed
occupancy detection model, ablation experiments were
conducted on the public dataset VOC2012, CrowdHu-
man and the self-built dataset and compared with the
current mainstream algorithm and its improved models
in terms of mAP, inference time (Latency), model
parameters and neural network model computation
(GFLOPs). The experimental results show that the
algorithm achieves the best detection performance on
all three datasets.

2 YOLOv5 occupancy detection model

2.1 YOLOv5 network

The DFV-YOLOv5 algorithm proposed in this paper is a
new design and improvement of components based on
YOLOv5 architecture. The anchor-free mechanism is
used to replace the ABM, no longer clustering the data
to obtain the anchor set of the input. At the output, the
decoupling detection head is used to speed up the con-
vergence of the model. VariFocal loss is used to balance
the positive and negative samples on the side of the loss
function. YOLOv5 algorithm is a one-stage object
detection algorithm issued by UitralyticsLLC Company.
Compared with YOLOv4, YOLOv5 has the advantages

![ReLU and SiLU activation function curve](image-url)

**Fig. 1** ReLU and SiLU activation function curve
of fast detection speed, high accuracy, less training time, fast inference speed, and so on. The network structure of YOLOv5 is divided into four parts: input, backbone, neck, and head. When input data, Mosaic [44] data augmentation, adaptive anchor calculation, adaptive image scaling, and other methods are used to improve the performance of small object detection. The backbone network uses Focus structure and C3Darknet-53 structure, neck is composed of FPN [45] + PANet [46] structure, head uses CIoU_loss [47] as the loss function of bounding box regression, and DIOU_nms [48] filters redundant prediction boxes. The SiLU activation function is used to replace the ReLU [49] activation function, which enhances the nonlinear expression ability of the model, speeds up the convergence of the model, and is easier to train. The activation function curve of ReLU and SiLU is shown in Fig. 1. The architecture of YOLOv5 is shown in Fig. 2. The expressions for the ReLU and SiLU activation functions are shown below.

\[
\begin{align*}
    f(x)_{\text{ReLU}} &= \begin{cases} 
        0, & \text{if } x < 0 \\
        x, & \text{if } x \geq 0
    \end{cases} \\
    f(x)_{\text{SiLU}} &= x \cdot \delta(x) = x \cdot \frac{1}{1 + e^{-x}}
\end{align*}
\]

2.2 Focal loss

The loss of the YOLOv5 classification branch is binary cross-entropy loss (BCE), the confidence loss branch is BCE, and the location branch is CIoU loss. To solve the imbalance between positive and negative samples in the detection process, FocalLoss is used to optimize the confidence loss function. The calculation is as shown in the formula:

\[
\begin{align*}
    FL(p_t, y) &= \begin{cases} 
        -\alpha(1-p_t)^{\gamma}\log(p_t), & \text{if } y = 1 \\
        -(1-\alpha)p_t^{\gamma}\log(1-p_t), & \text{otherwise}
    \end{cases}
\end{align*}
\]

where \( p_t \) is the prediction probability of the foreground, when \( y = 1 \) the prediction object is a positive sample, the otherwise are negative samples. \( \alpha \) is the decay of the first power function, and \( \gamma \) is the exponential decay.

Prediction box location uses CIoU loss. The CIoU considers scale information on top of the IoU in terms of overlap of borders, centre distance, and aspect ratio. The loss function is as follows:

\[
\begin{align*}
    \text{IoU} &= \frac{|B \cap B_{gt}|}{|B \cup B_{gt}|} \\
    \text{CIoU} &= \text{IoU}(B, B_{gt}) - \frac{p^2(B, B_{gt})}{C^2} \\
    \beta &= \frac{v}{(1 - \text{IoU}) + v} \\
    v &= \frac{4}{\pi^2} \left( \arctan \frac{w_{gt}}{h_{gt}} - \arctan \frac{w}{h} \right)^2 \\
    L_{\text{CIoU}} &= 1 - \text{IoU} + \frac{d^2}{C^2} + v \beta
\end{align*}
\]

Fig. 2 The architecture of the YOLOv5
In this formula, \( B \) is the reference standard box of the training label, \( B_i \) is the bounding box of the detection, \( C \) is the \( B_i \), and \( B_j \) is the minimum outsourced bounding box. Here \( d \) represents the Euclidean distance between the two centres of the calculation, \( v \) is a parameter measuring the consistency of the aspect ratio and takes values from 0 to 1, and \( \beta \) is a balancing factor measuring the loss due to the aspect ratio and the loss due to the IoU component.

The loss function in the present YOLOv5 is defined as follows:

\[
L_{total} = L_{conf} + L_{cls} + L_{IoU}
\]

Confidence = \( \Pr(\text{Class}_i|\text{Object}) \times \Pr(\text{Object}) \times \text{IoU} \)

\[
L_{conf} = \lambda_{obj} \sum_{i=0}^{s^2} \sum_{j=0}^{B_i} l_{ij}^{obj} \left[ -\alpha (1 - C_i^j)^{\gamma} C_i^j \log C_i^j - (1 - \alpha) (C_i^j)^{\gamma} (1 - C_i^j) \log (1 - C_i^j) \right] + \lambda_{noobj} \sum_{i=0}^{s^2} \sum_{j=0}^{B_{noobj}} l_{ij}^{noobj} \left[ -\alpha (1 - C_i^j)^{\gamma} C_i^j \log C_i^j - (1 - \alpha) (C_i^j)^{\gamma} (1 - C_i^j) \log (1 - C_i^j) \right]
\]

\[
L_{cls} = \sum_{i=0}^{s^2} \sum_{c \in \text{classes}} l_{ij}^{obj} \left[ -\hat{P}_i(c) \log P_i(c) - (1 - \hat{P}_i(c)) \log (1 - P_i(c)) \right]
\]

The first term in above formula is position loss, the second and third terms are category loss and bounding box regression loss respectively. Where \( s^2 \) represents the number of cells, \( B \) represents the number of predicted bounding boxes, \( l_{ij}^{obj} \) represents the object in \( i \)-th cell and \( j \)-th bounding box, \( \lambda_{obj} \) and \( \lambda_{noobj} \) represent the weight factor of the grid, \( C_i^j \) and \( \hat{C}_i^j \) represent confidence values for predicted and actual targets in \( j \)-th bounding box and \( i \)-th grid, \( P_i^j \) and \( \hat{P}_i^j \) represent probability values for predicted and actual targets in \( j \)-th bounding box and \( i \)-th grid. \( \Pr(\text{Class}_i|\text{Object}) \) is the probability of belonging to a specific class under the condition of an object, \( \Pr(\text{Object}) \) represents whether a bounding box contains object.

**2.3 The problems of the YOLOv5 occupancy detection model**

In the field of building energy efficiency, many researchers have detected occupancy rates in rooms based on video analysis to control HVAC and lighting systems in buildings, with tremendous energy saving potential, so improving the accuracy of occupancy detection models is a primary concern. YOLOv5 implements SOTA in the field of target detection. However, the performance is poor in detecting dense or occluded targets. In particular, K-means clustering analysis and genetic algorithms need to be applied to the training data before the model is trained to determine the best set of anchors, which have data correlation and poor generalization performance and increase the complexity of the detection head. The coupled probe heads lead to slow convergence of the model and limit the detection performance of the model. The whole process of occupancy detection is shown in Fig. 3.

**3 Proposed method**

**3.1 DFV-YOLOv5 detector**

DFV-YOLOv5 uses the anchor-free mechanism without clustering the dataset. It inputs it directly into the detector, using a mosaic approach to data enhancement, after which the input is sliced and expanded to four channels. After feature extraction and dimensionality reduction of the C3-DarkNet53 backbone network, in the top-down process of FPN, the low-level features and high-level features are fused by upsampling to obtain a feature map for prediction in the bottom-up approach of PAN, the shallow layer is positioned. Information is passed to deeper layers to enhance multi-scale localization capabilities. The final prediction of the feature map is made at the decoupled head. The whole process of occupancy detection is shown in Fig. 4. The network structure of DFV-YOLOV5 is shown in Fig. 5.

**3.2 Improvement of DFV-YOLOv5**

**3.2.1 Anchor-free**

YOLOv5 follows the original anchor-based mechanism (ABM) of YOLOv3 and performs a clustering analysis to determine the best set of anchors before training, resulting in better detection performance. YOLOv5 first divides the original image into several grids during the training phase. Anchors of different sizes are then obtained in each grid by clustering, which is then compared with ground truth box IoU, encoded to link training targets or supervised information. The prediction box is received in the prediction phase by decoding the prediction offsets and the anchors in the grid. However, this mechanism has poor generalization performance due to the data correlation between the anchor point sets. In addition, the ABM-based needs to calculate the IOU value between each anchor box and the ground truth box to assign its training label. It will increase the
computational cost and the complexity of the detection head. In recent years, the development of detection algorithms for the anchor-free mechanism (AFM) has accelerated, reducing the number of hyperparameters while maintaining similar detection accuracy. The anchor-free mechanism reduces the original prediction frame from three per pixel points to just one point laid down. The direct output of four predictions, including the upper left corner coordinates of the target box and the offset of the height and width of the detection box, reduces making three predictions for each feature point to one, which significantly reduces the number of parameters in the prediction layer as shown in Fig. 9. The input image scale is 640×640, and the feature tensor is downsampled by 32×, 16× and 8× and output at three scales. Unlike the baseline model, which predicts three boxes per feature tensor, the AFM predicts only one enclosing box per feature tensor, reducing the number of parameters predicted by two-thirds.
3.2.2 Decoupled head

In object detection tasks, it is a well-known problem that classification and regression tasks have different feature requirements for feature space, so decoupling detection heads are used in most target detectors to solve this conflict. With the evolution of the backbone network and feature pyramid, YOLO series detection algorithms have been using coupled detection heads since YOLOv3. The YOLOv5 algorithm follows the coupling head of YOLOv3 to complete the classification and regression in a $1 \times 1$ convolution, which limits the convergence speed and accuracy of the model. However, the decoupled detection head can accelerate the convergence speed of the model and improve the detection accuracy. At the same time, it will also bring little additional parameters and computational costs. The decoupled structure diagram is shown in Fig. 6. The decoupling experiments on the YOLOv3 model are carried out in reference [31]. The results show that the coupled detection head may damage the performance. Replacing the YOLO head with the decoupled YOLO head dramatically improves the convergence speed and improves the detection performance in the COCO dataset.

In the DFV-YOLOv5 algorithm proposed in this paper, the difference between the decoupled head and the YOLOv5 coupled head is that: in each layer of FPN features, a $1 \times 1$ Conv layer is used to reduce the feature channels to 256, and then two parallel branches are added, each with two $3 \times 3$ Conv layers for the classification and regression tasks, which are then finally integrated to predict the results. The regression branch includes the IoU branch and the Reg branch, where Reg(H, W, 4) is used to judge the regression parameters of a feature point, and the parameters are adjusted to obtain the prediction frame. Obj(H, W, 1) in the IoU regression task is used to judge whether a feature point contains an object. Cls(H, W, C) in the classification task indicates the category corresponding to the object contained in each feature point. The output after integration is Out(H, W, 4+1+C). The detailed network structure is shown in Table 10. Although the algorithm adds a small number of parameters, the performance of the decoupled head is significantly improved in the classification and regression tasks, and the decoupled detection head model is shown in the red dashed box in Fig. 5.

3.2.3 VariFocal loss

The loss function of the YOLOv5 model is based on Focal loss [50], which optimizes the class imbalance problem by assigning more weights to a small number of difficult samples and doing more attenuation to the weights of the majority of simple samples. To address the problem of poor performance of target detection models in small-range multi-target intensive detection tasks, VarifocalNet and the IoU-Aware Classification Score (IACS) are proposed to represent both target presence and location accuracy perception (or IoU [51] perception), as well as a new VariFocal loss function to train the dense object detector to do IACS regression. Rather than predicting an additional location accuracy score, it is combined into the classification score (IoU-aware centres). The predicted location-
VariFocal loss trains a dense target field within an irregular grid by stacking such deformable context information concerning a relatively large receptive region of convolution. Performing a convolutional operation on the difference between the bounding box and the actual box value. The classification loss and confidence loss are combined with the IoU-aware or IoU-aware classification score can represent both the presence of the target and the localization accuracy. The following are the calculations based on equation.

\[
VFL(p, q) = \begin{cases} 
-q(g \log(p) + (1 - q) \log(1 - p)), & \text{if } q < g; \\
-2p^q \log(1 - p), & \text{if } q = 0
\end{cases}
\]

where \( p \) is the predicted IACS, \( q \) is the object IoU score, for the positive sample \( q \) is the IoU between the prediction box and the ground truth box of the training label, and for the negative sample, \( q = 0 \). From the above equation, it can be seen that the number of negative samples is moderated using the \( x \) modulation factor and the negative samples are attenuated by the \( p^q \) parameter, thus making better use of the positive samples. In addition, using the \( q \) parameter weighting for the positive samples, when the positive samples and the true frame have a high cross-merge ratio, the contribution of the positive samples to the loss becomes larger, allowing the training to then focus on the high-quality positive samples.

The predicted value of \( p \) for IACS is estimated from a star-shaped representation of the boundary features. The yellow dots, as shown in Fig. 7 are nine fixed sampling points, and these sampling point features form a deformable convolutional bounding box. The method, therefore, incorporates the geometric information of the bounding box and the contextual information around the sampled points, facilitating a more accurate prediction of the offset between the bounding box and the actual box value. The deformable convolutional operation is a more generic form of convolution. Performing a convolutional operation on these sampling positions, the model can perceive more context information concerning a relatively large receptive field within an irregular grid by stacking such deformable convolution layers [52]. VariFocal loss trains a dense target detection model to predict IACS and uses the star-shaped bounding box representation described above to estimate \( p \)-values. The initial regression box (red box) was refined to a more accurate box (blue box).

In the DFV-YOLOv5 model, the Focal loss is replaced by the VariFocal loss function, and asymmetric weight decay is used when dealing with positive and negative complex samples. Using asymmetric attenuation strategy for positive and negative samples, VariFocal loss only reduces the loss contribution of negative cases \( (q = 0) \). Still, it does not reduce the weight of positive cases \( (q > 0) \) in the same way. The number of positive samples is smaller than that of negative samples, so it can better retain the learning information of positive samples. \( q \) is used to weight the positive sample, and if the positive sample has a high IoU, then the contribution of loss is greater so that the training can focus on those high-quality samples. In addition, to balance the positive and negative samples of the population, VariFocal loss also uses \( x \) to weight the negative samples. The weighting strategies of Focal loss and VariFocal loss for positive and negative samples are shown in Table 1. In Focal loss \( \gamma = 2.0, \alpha = 0.25 \), in VariFocal loss \( \gamma = 1.5, \alpha = 0.25 \).

The similarity between the DFV-YOLOv5 model and the YOLOv5 model is that it also uses CIoU as the bounding box regression loss function. The difference is that the model’s confidence loss and classification loss are based on the binary cross-entropy loss function of VariFocal loss. Because the images generated by data enhancement methods such as mosaic are sometimes far away from the target distribution of the natural scene, data enhancement is turned off in the last 15 epochs in the training process, and the \( L_1 \) loss function is added for regression. The experimental results show that this method can significantly improve the performance of the model on the verification set, so the loss function in DFV-YOLOv5 is defined as follows.

\[
L_{\text{total}} = L_{\text{conf}} + L_{\text{cls}} + L_{\text{CIoU}} + L_1
\]

\[
L_{\text{conf}} = \lambda_{\text{obj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} l_{ij}^{\text{obj}} \left[ -q(q \tilde{C}_i^j \log C_i^j) \\
+ (1 - q) (1 - \tilde{C}_i^j) \log (1 - C_i^j) \right] \\
- \alpha (C_i^j \tilde{C}_i^j (1 - \tilde{C}_i^j) \log (1 - C_i^j) \\
+ \lambda_{\text{obj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} l_{ij}^{\text{obj}} \left[ -q(q \tilde{C}_i^j \log C_i^j) \\
+ (1 - q) (1 - \tilde{C}_i^j) \log (1 - C_i^j) \right] \\
- \alpha (C_i^j \tilde{C}_i^j (1 - \tilde{C}_i^j) \log (1 - C_i^j))
\]

(15)
3.2.4 Ablation experiments

To verify the impact of each improvement on the detection performance, ablation experiments are carried out on VOC2012 datasets, CrowdHuman datasets and self-built datasets, and the experimental results are shown in Tables 2, 3, and 4. Where mAP$_{50}$ and mAP represent the mean average accuracy, Params represents the number of parameters of the network model, GFLOPs represents the computation amount of the neural network, Latency represents the single frame inference speed of the model, and FPS represents the number of model detection frames per second.

Experimental results show that when the anchor-free mechanism is adopted, the computation amount of prediction layer parameters and neural network is reduced by 2/3, and the Latency is reduced. The mAP value of VOC2012 and the self-built dataset is improved. When the anchor-free mechanism and decoupled head are used, the detection performance of the three datasets is significantly enhanced with the addition of a small number of calculation parameters (1.91M, 1.88M, and 1.88M, respectively). When VariFocal loss is used, the parameters and computation of the neural network are not increased, and the mAP values of the three datasets are also improved. As shown in
Table 4, the model index decreases after using the anchor-free mechanism in the CrowdHuman dataset. Because there are many overlapping targets in the dataset, a feature point in the AFM mechanism may correspond to multiple predictions, and it can only choose one of them as output, so the performance is affected to a certain extent.

The total loss function of DFV-YOLOv5 is shown in Equation (14), which consists of classification loss, logistic regression loss, confidence loss and $L_1$ loss. As some samples deviated from the target distribution of natural scenes after mosaic enhancement in the training process, all data enhancement was closed in the last 15 epochs to explore the ability of this training method to improve the detection accuracy of the verification set. The $L_1$ loss was only regression calculated in the epoch after closing data enhancement. To assess the effectiveness of each loss, ablation experiments were performed on each loss on three datasets. For a fair comparison, all the models were trained by setting the same hyper-parameters, e.g., setting the number of batch size to 32, the number of training epochs to 150.

The experimental results obtained from the above ablation experiments are shown in Tables 5, 6, and 7, it is clear that the detection results are optimal on all three datasets when all four losses are employed simultaneously.

Table 5 The effect of loss components on mAP

| $L_{cls}$ | $L_{conf}$ | $L_{CIoU}$ | $L_1$ | mAP  | $\Delta$ | mAP$_{50}$ | $\Delta$ |
|---------|-----------|------------|-----|-----|--------|----------|--------|
| ✓ | ✓ | ✓ | ✓ | 4.07 | -48.94 | 6.30 | -65.79 |
| ✓ | ✓ | ✓ | ✓ | 12.20 | -40.81 | 22.70 | -49.39 |
| ✓ | ✓ | ✓ | ✓ | 0.34 | -50.43 | 1.58 | -70.51 |
| ✓ | ✓ | ✓ | ✓ | 46.64 | -6.37 | 68.10 | -3.99 |
| ✓ | ✓ | ✓ | ✓ | 53.01 | -   | 72.09 | -   |

We trained the model on the VOC2012 dataset to turn off $L_1$ loss, CIoU loss, confidence loss and classification loss. Our baseline (last row) combines all losses

Table 6 The effect of loss components on mAP

| $L_{cls}$ | $L_{conf}$ | $L_{CIoU}$ | $L_1$ | mAP  | $\Delta$ | mAP$_{50}$ | $\Delta$ |
|---------|-----------|------------|-----|-----|--------|----------|--------|
| ✓ | ✓ | ✓ | ✓ | 49.5 | -5.8  | 82.5 | -3.8  |
| ✓ | ✓ | ✓ | ✓ | 14.1 | -41.2 | 33.7 | -52.6 |
| ✓ | ✓ | ✓ | ✓ | 2.1  | -53.2 | 12.9 | -73.4 |
| ✓ | ✓ | ✓ | ✓ | 54.5 | -0.8  | 86.6 | -0.2  |
| ✓ | ✓ | ✓ | ✓ | 55.3 | -    | 86.3 | -    |

We trained the model on the CrowdHuman dataset to turn off $L_1$ loss, CIoU loss, confidence loss and classification loss. Our baseline (last row) combines all losses

When the $L_1$ loss is not used, there is a slight decrease in mAP values on all three datasets, thus also validating that turning off data enhancement at the appropriate time is effective in improving the model’s performance. When the regression loss $L_{CIoU}$ is not used, the detection performance of the model decreases substantially, with mAP values close to zero on all three datasets, thus validating the importance of this loss for model training. When the confidence loss $L_{conf}$ is not applied, the model performance drops substantially, and therefore this loss is critical for target localization. When the classification loss $L_{cls}$ is turned off, the mAP decreases less on the self-built dataset and the CrowdHuman dataset ($\Delta \leq 10$), however, the mAP is close to zero on the VOC2012 dataset since the first two datasets are single-class datasets. In contrast, the VOC2012 is a multi-class dataset relying more on the classification loss to achieve correct classification.

3.3 Transformer encoder and BIFPN block

Attention mechanism helps to quickly screen out high-value and practical information from a large amount of data, which significantly improves the efficiency and accuracy of information processing in the visual system [54]. Google first proposed the concept of Transformer in 2017 [42]. It was initially used in natural language processing (NLP) tasks, using Attention modules to stack neural networks, known as Transformer, to complete NLP tasks without LSTM and RNN. After the success in the field of NLP, Transformer has been applied to the area of computer vision by researchers [55–58]. Especially in image classification, the difference of the target image usually focuses on the details of some areas. At the same time, the attention mechanism can effectively select the target area and focus on the spot to obtain more features of the target. In 2020, Google proposed a Vision Transformer (ViT) [42] network for image classification, which consists of two sub-layers: the first is a Multi-Head Attention Layer (MHL), and the second is a Fully Connected Layer (MLP).
with residual connections between each sub-layer. The transformer module improves the ability to record multiple types of location data. It employs a self-attentive approach to investigate the potential of feature representation, demonstrating that it outperforms public datasets on densely occluded objects. In this study, the ViT network replaced the C3 module in the baseline in a comparative experiment to investigate the functionality of the module and its impact on the model parameters. The network structure is shown in Fig. 8.

BIFPN is a weighted bi-directional feature pyramid network that allows simple and fast multi-scale feature fusion with the aim of pursuing a more efficient way of multi-scale fusion. Whereas previous feature fusion treated different scale features equally, BIFPN introduces weights to better balance feature information at different scales, and the contrasting network structure is shown in Fig. 8. In the Bifp-YOLOv5s6 model, the improved YOLOv5 model replaces PANET by the BIFPN module with Concat at layers 14, 18, 22, 25, 28 and 31.

### 3.4 DFV-YOLOv5 output-logistic

The output layer of the DFV-YOLOv5 detection network uses logistic output for prediction, which can support the detection of multi-tag targets. The detection network maps it to one different output tensor on self-built datasets, indicating the probability that various objects exist in each location in the image. As shown in Fig. 9, we enter a picture of 640×640, different from YOLOv5 set 3 boxes
for each of the size metrics. Our proposed algorithm adopts anchor-free mechanism only set 1 box for each of the metrics of each size, for a total of $20/C_2^2 = 10240/400$ predictions. Each prediction is a vector of $(4+1+1=6)$ dimensions, including four-position coordinates, border confidence, and the probability of 1 object class.

3.5 Self-built datasets

Public datasets in target detection, such as MSCOCO, PASCAL VOC, ImageNet, etc., are for general life scenarios. In contrast, occupancy detection in buildings often suffers from occlusion due to the angle of camera installation, and public fields or office areas often have a high density of people, for which such datasets are currently lacking. We used a university classroom as the research scenario to address the above issues. We collected data on the classroom status of classroom personnel at different times using monitoring facilities installed at different angles of the classroom. The dataset comprises different periods of away states, divided into classroom state, recess state, daytime classroom, nighttime classroom, high staff density, and low density. 11,367 data were collected and labelled built a classroom dataset with the image resolution of $1920/C_2^2 = 1080$ and $2560/C_2^2 = 1440$. The dataset’s composition for different angles and person densities is shown in Fig. 10.

4 Experimental results and analysis

In this paper, the performance of the target detection model was tested using VOC2012, CrowdHuman dataset and a self-built classroom dataset. Firstly, the latest detection algorithms and models proposed in this paper are compared in terms of mean Average Precision (mAP), mAP$_{50}$, Precision, Recall, detection speed and the number of parameters. Secondly, ablation experiments are conducted on the benchmark YOLOv5 model and the improved DFV-YOLOv5 model, and the performance of the different improved methods is tested.

4.1 Experimental dataset

The VOC2012 dataset was able to be used for model evaluation due to a large number of dense target images, inter-and intra-class overlap masking, and the number of person samples in the dataset exceeded 50% of the total sample size. In addition, the model was further evaluated on the larger public dataset, the CrowdHuman dataset, to assess better the performance of the dense human occupancy model in buildings. This dataset contains many indoor scenes, with 15,000 in the training set, 5000 in the test set and 4370 in the validation set. There are 470K instances in the training and validation sets, containing approximately 23 people per image, along with a variety of occlusions. The three datasets in this paper are used to evaluate the model proposed in this study and calculate mAP and mAP$_{50}$. The self-built datasets are also divided
into a training set, a validation set and a test set, with a ratio of 8:1:1 between the validation and test sets. 9094 samples are included in the training set, 1136 samples in the validation set and 1137 samples in the test set.

### 4.2 Comparative methods and parameter setting

This research uses the Pytorch 1.8.0 deep learning framework to train 200 epochs for each model on the VOC2012 dataset and a self-built large-scene classroom dataset for the occupancy detection problem, with a warm-up of 3 epochs. The adam [59] optimizer is used for parameter optimization, and the stochastic gradient descent (SGD) and backpropagation algorithms are used to learn the network parameters, with the momentum set to 0.937, the weight decay set to 0.0005, the initial learning rate set to 0.01, the image input size set to 640×640, the batch size set to 32, and the IoU threshold set to 0.2. The operating system of this experimental platform is Ubuntu 18.04 and Windows 10, and the processor Intel(R) Core(TM) i7-9800X CPU, the main frequency is 3.80 GHz, the memory is 32 GB, and the GPU is NVIDIA RTX 2080ti to train the detection model. During the training process, as the improved algorithm model DFV-YOLOv5 is shared with the YOLOv5 backbone network (blocks 0–9), the weights of the training model can use the pre-training weights YOLOv5, and using and training weights can significantly reduce the training time.

### 4.3 The occupancy detection performance evaluation indices

To quantitatively evaluate the model’s performance, Precision, Recall, AP, and mAP are used as evaluation indicators.

\[
\text{Precision} = \frac{TP}{TP + FP} \tag{18}
\]

\[
\text{Recall} = \frac{TP}{TP + FN} \tag{19}
\]

TP indicates that it is judged to be a positive sample, which is a positive sample, and FP suggests that it is considered a positive sample but a negative sample. FN indicates that it is judged as a negative sample but a positive example.

\[
AP = \int_0^1 P(R)dR \tag{20}
\]

\[
mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{21}
\]

The AP values consider precision and recall and can be expressed as the area under the P–R curve. The mAP represents the average AP value for all classified objects. AP50 represents the measured value of AP when the IoU threshold is 0.5, AP75 represents the measured value of AP when the IoU threshold is 0.75, APS represents the measured value of AP of the target frame whose pixel area is less than 32², AMP represents the measured value of AP of the target frame with pixel area of 32² – 96², and APS represents the measured value of AP of the target frame whose pixel area is greater than 96².

### 4.4 Experiments and comparisons

#### 4.4.1 Experiment on Pascal-VOC2012 dataset

The Pascal-VOC2012 dataset is a well-known public object detection dataset that has been used to test the performance of most image object detection and image segmentation techniques. It includes 11,355 large-scale, accurately manually annotated, well-aligned images of visible natural scenes, divided into four main categories, namely people, common animals, traffic vehicles, indoor furniture, and so on, with 20 subcategories, including a variety of regular street and natural urban scenes. Because the detection target of this paper is human, and the personnel samples in the VOC2012 dataset are more than half of the total samples, the dataset is persuasive for evaluating the occupancy detection model.

To better demonstrate the improvements in the described decoupling, anchor-free mechanism, and VariFocal loss techniques for the occupancy detection models, the improved models are compared with the mainstream algorithmic models YOLOv3, YOLOv4, and YOLOv5. The YOLOv5 model is ablated for experiments with recent research results such as the attention mechanism Transformer and bidirectional feature pyramids. The technical composition of each model is shown in Table 8. The variables in the ablation experiments include:

- The small target detection layer (s6).
- The loss function VariFocal Loss.
- The Bifpn or transformer module.
- The decoupled head.
- The anchor-free mechanism.

The DF-YOLOv5 model uses a decoupling head and uses the anchor-free mechanism, while DFV-YOLOv5 uses VariFocal loss on this basis. The performance comparison results of the experiment are shown in Table 9. The YOLOv5s6 model with only a small target detection layer improves 4.47% on mAP50 compared with the baseline model and improves 6.02% on mAP after replacing the...
original Focal loss with the VariFocal loss function, and achieves the best performance among all models. The Transformer module is embedded in layer 11 of the YOLOv5s6 backbone network, and the performance mAP on the VOC2012 dataset is 73.07%. The Bifpn module is embedded in layers 14, 18, 22, 25, 28, and 31, and the mAP50 value is 72.7%. When using the Bifpn module and VariFocal loss simultaneously, the mAP50 value is 72.75%, which has no significant improvement compared with the use of the two modules alone.

Although our improved model described above offers significant performance improvements over the current mainstream detection models, it also introduces many parameters, a larger model, and slower model inference. The DFV-YOLOv5 algorithm proposed in this paper uses an anchor-free mechanism, decoupled detection header, and VariFocal loss without increasing the additional computational burden of the small object detection layer. Compared with the baseline model YOLOv5, the leading indicators such as mAP50 and Precision are improved by 4.6% and 5.3%, respectively, while only 1.8M increases
Fig. 11 Training curves for YOLOv5 and its variants on the public dataset VOC2012

(a) mAP (IoU=0.5).
(b) mAP_{0.5:0.95} (AP of 10 IoU thresholds with an average of 0.5-0.95).

(c) Precision curve.
(d) Recall curve.

(e) Location loss curve.
(f) Classification loss curve.

(g) Confidence loss curve.
the calculation parameters. To better evaluate the model’s performance, mAP is used to calculate the average detection accuracy under ten different IOU thresholds. As shown in Table 9, the mAP of DFV-YOLOv5 is 53.01%, which is 6.05% higher than that of the YOLOv5 model. The size of the weight model is almost the same as the baseline YOLOv5. Compared with other improved models, it not only ensures the speed and accuracy of reasoning but also maintains the model’s size, which is more conducive to the model migration of mobile devices.

To provide a more visual representation of the performance of each model, we visualize the change in metrics for each model during the training phase. Figure 11 shows the training curves of several indicators of the baseline model YOLOv5s, YOLOv5s6, bifpn-YOLOv5s6, VFL-YOLOv5s6, and Trans-YOLOv5s6 model. Figure 11a shows the mAP training curve for 20 classes of targets when the threshold value of IoU is 0.5. Figure 11b shows the average training curve of mAP for ten different thresholds when the IoU is 0.5–0.95. The x-axis represents the Epoch number, and the y-axis represents the accuracy. From Fig. 11a, b, it can be seen that when the Epoch number is around 100, the curve changes slowly, and the model approaches convergence. As the curves show, by adding the small target detection
model, Bifpn, and Transformer modules, and using the VariFocal loss strategy, the detection performance can be significantly improved. Using the VariFocal loss and Transformer modules alone gives the best results, with the best accuracy metrics and the lowest loss values. The more minor training losses in Fig. 11e–g indicate better performance. However, the Parames and Weight metrics in Table 9 show that this improved approach introduces many parameters, resulting in an oversized model.

The DF-YOLOv5 technique suggested in this research is an enhanced multi-target identification algorithm in a real scenario. The performance is greatly improved compared to the current mainstream target detection algorithms YOLOv3, YOLOX, YOLOv4, Scaled-YOLOv4, and YOLOv5. In contrast, the detection speed, parameters, and model complexity remain unchanged, and the detection performance indicators mAP50 and Precision are improved by 4.22% and 5.3%, respectively, especially the mAP index. Detailed training curves

| From | Layer (type)          | Output shape                  | Param  |
|------|-----------------------|--------------------------------|--------|
| 0    | -1                    | Models.common.Focus [3, 32, 3, 1, None, 1, 'relu'] | 3520   |
| 1    | -1                    | models.common.Conv [32, 64, 3, 2, None, 1, 'silu'] | 18560  |
| 2    | -1                    | models.common.C3 [64, 64, 1, True, 1, 0.5, 'relu'] | 18816  |
| 3    | -1                    | models.common.Conv [64, 128, 3, 2, None, 1, 'silu'] | 73984  |
| 4    | -1                    | models.common.C3 [128, 128, 3, True, 1, 0.5, 'relu_silu'] | 156928 |
| 5    | -1                    | models.common.Conv [128, 256, 3, 2, None, 1, 'silu'] | 295424 |
| 6    | -1                    | models.common.C3 [256, 256, 3, True, 1, 0.5, 'relu_silu'] | 625152 |
| 7    | -1                    | models.common.Conv [256, 512, 3, 2, None, 1, 'silu'] | 1180672|
| 8    | -1                    | models.common.SPP [512, 512, [5, 9, 13], 'silu'] | 656896 |
| 9    | -1                    | models.common.C3 [512, 512, 1, False, 1, 0.5, 'relu_silu'] | 1182720|
| 10   | -1                    | models.common.Conv [512, 256, 1, 1, None, 1, 'silu'] | 131584 |
| 11   | -1                    | Upsample [None, 2, 'nearest'] | 0      |
| 12   | [-1, 6]               | models.common.Concat [1] | 0      |
| 13   | -1                    | models.common.C3 [512, 256, 1, False, 1, 0.5, 'relu_silu'] | 361984 |
| 14   | -1                    | models.common.Conv [256, 128, 1, 1, None, 1, 'silu'] | 33024  |
| 15   | -1                    | Upsample [None, 2, 'nearest'] | 0      |
| 16   | [-1, 4]               | models.common.Concat [1] | 0      |
| 17   | -1                    | models.common.C3 [256, 128, 1, False, 1, 0.5, 'relu_silu'] | 900880 |
| 18   | -1                    | models.common.Conv [128, 128, 3, 2, None, 1, 'silu'] | 147712 |
| 19   | [-1, 14]              | models.common.Concat [1] | 0      |
| 20   | -1                    | models.common.C3 [256, 256, 1, False, 1, 0.5, 'relu_silu'] | 296448 |
| 21   | -1                    | models.common.Conv [256, 256, 3, 2, None, 1, 'silu'] | 590336 |
| 22   | [-1, 10]              | models.common.Concat [1] | 0      |
| 23   | -1                    | models.common.C3 [512, 512, 1, False, 1, 0.5, 'relu_silu'] | 1182720|
| 24   | 17                    | models.common.Conv [128, 128, 1, 1, None, 1, 'silu'] | 16640  |
| 25   | 20                    | models.common.Conv [256, 128, 1, 1, None, 1, 'silu'] | 33024  |
| 26   | 23                    | models.common.Conv [512, 128, 1, None, 1, 'silu'] | 65792  |
| 27   | 24                    | models.common.Conv [128, 128, 3, 1] | 295424 |
| 28   | 24                    | models.common.Conv [128, 128, 3, 1] | 295424 |
| 29   | 25                    | models.common.Conv [128, 128, 3, 1] | 295424 |
| 30   | 25                    | models.common.Conv [128, 128, 3, 1] | 295424 |
| 31   | 26                    | models.common.Conv [128, 128, 3, 1] | 295424 |
| 32   | 26                    | models.common.Conv [128, 128, 3, 1] | 295424 |
| 33   | [-27, 28, 29, 30, 31, 32] | DetectX [20, 1, [128, 128, 128, 128]] | 1      |

Model summary: 360 layers, 8945035 parameters, 8945035 gradients
Fig. 13 Performance comparison of the improved DFV-YOLOv5 model and the baseline model in the VOC2012 training set.

(a) mAP(IoU=0.5).

(b) mAP_{0.5:0.95}(AP of 10 IoU thresholds with an average of 0.5-0.95).

(c) Precision curve.

(d) Recall curve.

(e) Confidence loss curve.

(f) Classification loss curve.

(g) II loss curve.

(h) Location loss curve.
Fig. 14 Detection results of the proposed DFV-YOLOv5 algorithm in the test set

Fig. 15 Detection results of the proposed DFV-YOLOv5 algorithm in the self-built data test set
before and after the improvements are depicted in Fig. 13. Where the DFV-YOLOv5_rslu model and DF-YOLOv5_rslu model are used, we use a hybrid activation function to explore its effect on inference speed. The mAP 50 value of DFV-YOLOv5 while utilizing VariFocal loss is 71.79%, and the value of mAP is 53.01%, which is 4.6% higher than the baseline model and 6.05% higher than the baseline model, respectively.

The training loss plays a vital role in the training process and reflects the relationship between the true and predicted values. The smaller the loss and the closer to the actual value, the better the model’s performance. The improved model in this paper uses the strategy of anchor-free prediction target to decouple the coupled detection head and calculates the loss during training in the decoupled head, firstly by decoupling the DFV-YOLOv5 head shown in Fig. 9 to the three branches ofCls, Reg, and IoU, and then merging to calculate the network coordinates on the feature map.

4.4.2 Experiment on self-built classroom dataset

To further illustrate the superiority of the model proposed in this paper on dense datasets containing multiple occlusion targets, comparative experiments will be carried out.

| Methods            | Backbone | Size | Epoch | APval (%) | APval_{50} (%) | AP_{50} (%) | AP_{75} (%) | AP_{L} (%) | AP_{S} (%) | Latency (ms) |
|--------------------|----------|------|-------|-----------|---------------|-------------|-------------|-------------|-------------|--------------|
| RetinaNet [50]     | ResNet50 | 512  | 86.7  | 59.3      | 71.8          | 64.0        | 33.9        | –           | –           | 31.7         |
| Faster-Rcnn [21]   | ResNet50 | 600  | 98.7  | 68.0      | 80.4          | 70.2        | 51.0        | –           | –           | 79.2         |
| CenterNet [30]     | ResNet50 | 512  | 97.8  | 68.8      | 83.7          | 71.5        | 52.4        | –           | –           | 13.7         |
| EfficientDet [60]  | EfficientNet-B2 | 768  | 93.3  | 68.8      | 84.5          | 71.6        | 57.8        | –           | –           | 75.5         |
| SSD300 [25]        | VGG16    | 300  | 87.7  | 53.7      | 58.9          | 59.8        | 13.5        | –           | –           | 9.8          |
| SSD512 [25]        | VGG16    | 512  | 98.3  | 71.6      | 86.9          | 74.3        | 54.7        | –           | –           | 16.9         |
| YOLOv3 [24]        | D53      | 640  | 88.1  | 86.9      | –             | –           | –           | –           | –           | –            |
| YOLOv4 [26]        | CSPD53   | 640  | 94.2  | 84.2      | –             | –           | –           | –           | –           | 38.2         |
| YOLOX [31]         | D53      | 640  | 90.8  | 80.9      | –             | –           | –           | –           | –           | 12.4         |
| Scaled-YOLOv4 [29] | CSP-P7   | 640  | 98.4  | 86.9      | –             | –           | –           | –           | –           | 24.2         |
| YOLOv5 [27]        | C3D53    | 640  | 96.4  | 87.5      | –             | –           | –           | –           | –           | 6.1          |
| DF-YOLOv5          | C3D53    | 640  | 99.5  | 90.9      | –             | –           | –           | –           | –           | 6.9          |
| DF-YOLOv5_rslu     | C3D53    | 640  | 99.3  | 90.2      | –             | –           | –           | –           | –           | 6.5          |
| DFV-YOLOv5         | C3D53    | 640  | 99.7  | 93.9      | –             | –           | –           | –           | –           | 6.9          |

Bold indicates the best value in each column of indicators and in the Methods column indicates the algorithm model proposed in this paper. (AP refers to AP at IoU = 0.50:0.05:0.95. AP_{50} and AP_{75} indicate AP at IoU = 0.50 and AP at IoU = 0.75, respectively. APS, APM and APL mean AP for small objects (area < 32^2), AP for medium objects (32^2 < area < 96^2) and AP for large objects (area > 96^2))

Table 12 Data performance of each model on CrowdHuman datasets

| Methods        | Backbone | Size | Epoch | APval (%) | APval_{50} (%) | AP_{50} (%) | AP_{75} (%) | AP_{L} (%) | AP_{S} (%) | AP_{L} (%) |
|----------------|----------|------|-------|-----------|---------------|-------------|-------------|-------------|-------------|------------|
| YOLOv3 [24]   | D53      | 640  | 150   | 38.7      | 77.1          | 39.4        | 78.7        | 81.2        | 81.7        | 82.9       |
| YOLOv4 [26]   | CSPD53   | 640  | 150   | 43.6      | 79.8          | 44.7        | 81.2        | 81.7        | 82.7        | 82.7       |
| YOLOX [31]    | D53      | 640  | 150   | 47.5      | 81.6          | 48.2        | 81.7        | 85.4        | 85.4        | 86.1       |
| YOLOv5 [27]   | C3D53    | 640  | 150   | 51.5      | 83.6          | 50.1        | 82.9        | 82.7        | 82.7        | 82.7       |
| YOLOv5s6      | C3D53    | 640  | 150   | 50.8      | 83.3          | 50.1        | 82.9        | 82.7        | 82.7        | 82.7       |
| F-YOLOv5      | C3D53    | 640  | 150   | 50.6      | 83.4          | 49.6        | 82.7        | 82.7        | 82.7        | 82.7       |
| DF-YOLOv5     | C3D53    | 640  | 150   | 54.5      | 85.7          | 54.6        | 85.4        | 85.4        | 85.4        | 86.1       |
| DFV-YOLOv5    | C3D53    | 640  | 150   | 55.3      | 86.3          | 55.1        | 86.1        | 86.1        | 86.1        | 86.1       |

Bold indicates the best value in each column of indicators and in the Methods column indicates the algorithm model proposed in this paper.
on self-built datasets. Detailed comparison results are shown in Table 11.

The intermediate outputs of the neural network are visualized by further comparing the performance of the models through feature maps. A feature map shows the output features of each convolution and pooling layer in the network (the output of this layer is often referred to as the activation of that layer, i.e. the output of the activation function). The maps of the elements are visualized in three main dimensions: width, height, and depth (32 channels in total). Each channel corresponds to a relatively independent feature, so the correct way to visualize these feature images is to plot the contents of each channel as a two-dimensional image. We visualize the feature maps for 32 layers of the proposed model and 23 layers of the baseline model. Each layer of the feature map has 32 channels. The feature maps of the coupled predictor head of the baseline model are distributed in layers 17, 20, and 23. The decoupled predictor head of our proposed model corresponds mainly to layers 27 to 32. As shown in Fig. 12, Fig. 12b shows the feature map of the C3 module after 16-fold downsampling of the baseline model, which is used to couple the detection head for classification, regression, and IoU tasks. Figure 12c shows the feature map of the convolutional layer after the 16-fold downsampling decoupling of our proposed model, which is used for the classification task. Figure 12d shows the feature map of the convolutional layer of our proposed model after 16-fold downsampling decoupling, which is used for the regression and IoU tasks. We can see from the comparison of the feature maps that the decoupled convolutional layer's feature mapping focuses on different feature representations for various tasks, and the feature representation is more transparent and more conducive to classification and IoU tasks resulting in significantly improved decoupled model detection performance. For occupancy detection of indoor people, we focused on the feature representation of the model’s eight-fold downsampled small target detection layer due to the higher density of people, whose feature maps are shown in Fig. 12f–h. The feature representation in Fig. 12g, h is more precise and more prosperous than that in the baseline model in Fig. 12f. Figure 12a, e shows that the proposed algorithm (left panel) has finer prediction

![Fig. 16 The detection results of the proposed DFV-YOLOv5 algorithm are verified in the CrowdHuman data test set. The red box represents the improved model, and the pink box represents the baseline model test results.](image)
boxes and higher confidence scores than the baseline algorithm (right panel).

On the self-built dataset, the DFV-YOLOv5 algorithm is compared with the mainstream detection models Retina-Net, Faster-Rcnn, CenterNet, EfficientDet, SSD, YOLOv3, YOLOv4, Scaled-YOLOv4, YOLOX and YOLOv5s in terms of average detection accuracy and detection speed, and the advantage of the algorithm in share detection is verified. As shown in Table 11, in the target-dense classroom dataset, DF-YOLOv5 with the anchor-free strategy performs better detection accuracy and speed, with better generalization ability, as the classroom scenes are more open, with larger target sizes in front and smaller targets in the back. Among them, the AP50 value is the test result of the verification set during the training, which is 3.1% higher than that of the baseline model YOLOv5, reaching 99.5%, and the AP on the test set is 3.4% higher than that of YOLOv5, reaching 90.9%. On this basis, when using VariFocal loss, the AP50 value of the DF-YOLOv5 model comes 99.7%. The AP value goes 93.89%. Compared with the baseline model, it increased by 3.3% and 6.39%, respectively. In terms of detection time, the average single frame processing time of the proposed algorithm DF-YOLOv5 model is 6.9ms, which is higher than the baseline model YOLOv5 detection time of 0.8ms per frame.

To further improve the speed of model detection, this paper also explores the use of mixed activation functions of ReLU and SiLU to improve the reasoning time of the model. All the convolution layers with stride 2 in the DF-YOLOv5 model, the SPP module, and the last convolution layer of the C3 module adopt the SiLU activation function and other convolution standard modules use the ReLU activation function. The detailed architecture of the DF-YOLOv5_rslu model is shown in Table 10, where the leftmost number is the number of layers, From indicates the number of layers connected to that layer, where -1 represents the previous layer. The Layer column indicates the type of network in each layer, OutPut Sharp suggests the dimension of the feature map and the type of activation function used in each layer, and Param indicates the number of parameters in each layer. The single-frame detection time of the DF-YOLOv5_rslu model with mixed activation function on the test set is 0.4ms lower than that of the DF-YOLOv5 model and decreases by 0.2% and 0.7% in AP50 and AP, respectively. The detection effect of the DFV-YOLOv5 algorithm proposed in this paper in the VOC2012 dataset test set is shown in Fig. 14. The detection effect of the test set in the self-built classroom dataset is shown in Fig. 15.

4.4.3 Experiment on CrowdHuman dataset

CrowdHuman [61] is a benchmark dataset to better evaluate detectors in crowd scenarios. The CrowdHuman dataset is large, rich-annotated and contains high diversity. CrowdHuman contains 15,000, 4370, and 5000 images for training, validation, and testing, respectively. There are a total of 470K human instances from train and validation subsets and 23 persons per image, with various kinds of occlusions in the dataset. Each human instance is annotated with a head bounding-box, human visible-region bounding-box and human full-body bounding-box. The dataset has a strong relationship with the scene studied in this paper, so it is convincing to evaluate the performance of the occupancy detection model in this paper.

The improved DFV-YOLOv5 algorithm in this paper is compared with YOLOv3, YOLOv4, YOLOX, and baseline model YOLOv5 on AP50 and AP indicators on the verification set and test set of CrowdHuman. The experimental results are shown in Table 12, the AP50 and AP of the improved model DFV-YOLOv5 on the validation set are 2.7% and 3.8% higher than that of the baseline model YOLOv5. Compared with baseline model YOLOv5, the AP50 and AP of the improved model DFV-YOLOv5 on the test set improved by 3.2% and 5.0%, respectively. When the IoU threshold is set to 0.6, and the NMS threshold is 0.45, the result of the test on the CrowdHuman test set is shown in Fig. 16. Compared with the baseline model (pink box), the improved DFV-YOLOv5 model (red box) can significantly improve the detection effect in densely populated scenes.

5 Conclusion

To improve the generalization ability, model convergence speed and densely occluded target detection performance and reduce the complexity of the modular model coupled detection head, we propose a novel decoupled anchor-free, VariFocal loss-based convolutional network algorithm DFV-YOLOv5. This method uses AFM to reduce the number of design parameters that require heuristic adjustments. To reduce the model’s coupling, accelerate the model’s convergence speed, and improve the performance of model detection, the detection head is decoupled to solve the conflict between classification and regression tasks. We use VariFocal loss to assign more weights to positive samples to optimize the class imbalance problem.
and reassemble the total loss function. In addition, we build a classroom dataset to verify the occupancy detection performance of the model. To illustrate the effectiveness and feasibility of the DFV-YOLOv5 approach, it is compared with multiple mainstream models on mAP, inference time, and model parameters on three datasets.

The main contributions of our work are as follows: Firstly, to reduce the problem of poor anchor-based generalization, we use the anchor-free technique in combination with the YOLOv5 model for training from feature map pixel points. Secondly, we use two parallel branches for the decoupled probe heads, positioned for classification, to speed up training convergence. Third, to improve the detection accuracy of densely occluded targets, a VariFocal loss technique is used to further balance the positive and negative sample weights. Fourth, we build a classroom-intensive person dataset to compensate for the lack of a public dataset. Finally, experiments conducted on PASCAL VOC2012, CrowdHuman and self-built datasets validate the better performance of the DFV-YOLOv5 model for in-building occupancy detection through detailed comparisons with the YOLOv3, YOLOv4, Scaled-YOLOv4, and YOLOv5 models. In future research, further pruning of the proposed DFV-YOLOv5 model and improvement of the suppression strategy of the prediction frame is needed to facilitate its practical application.

**Author Contributions** All authors contributed equally.

**Funding** This study is partly supported by the National Natural Science Foundation of China (62003191), the Natural Science Foundation of Shandong Province (ZR2022QF072).

**Data Availability** The data that support the findings of this study are available from the corresponding author upon reasonable request.

**Declarations**

**Conflict of interest** None.

**Ethics approval** Confirm.

**Consent to participate** Confirm.

**Consent for publication** Confirm.

**References**

1. Beiter P, Elchinger M, Tian T (2017) 2016 renewable energy data book. Technical report, National Renewable Energy Lab (NREL), Golden
2. Xu Q (2019) Research on new energy saving technology of building lighting system. Ph.D. thesis, Suzhou University of Science and Technology
3. Petersen S, Pedersen TH, Nielsen KU, Knudsen M (2016) Establishing an image-based ground truth for validation of sensor data-based room occupancy detection. Energy Build 130:787–793
4. Zou J, Zhao Q, Yang W, Wang F (2017) Occupancy detection in the office by analyzing surveillance videos and its application to building energy conservation. Energy Build 152:385–398
5. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05), vol 1. IEEE, pp 886–893
6. Platt J (1999) Sequential minimal optimization: a fast algorithm for training support vector machines.
7. Zaveri S, Ross S, Erickson V, Kamthe A, Cerpa A (2011) Building energy management systems actuated using wireless camera sensor networks. ACM, New York
8. Li NJ, Weng CF, Wang WJ, Chen HC, Lee PJ (2013) The people number estimation based on the embedded dsp system with surveillance camera. In: International conference on system science and engineering
9. Liu D, Guan X, Du Y, Zhao Q (2013) Measuring indoor occupancy in intelligent buildings using the fusion of vision sensors. Meas Sci Technol 24(7):074023
10. Yang J, Pantazaras A, Chaturvedi KA, Chandran AK, Santamouris M, Lee SE, Tham KW (2018) Comparison of different occupancy counting methods for single system-single zone applications. Energy Build 172:221–234
11. Sun K, Zhao Q, Zhang Z, Hu X (2022) Indoor occupancy measurement by the fusion of motion detection and static estimation. Energy Build 254:11593
12. Zheng Y, Bao H, Meng C, Ma N (2020) A method of traffic police detection based on attention mechanism in natural scene. Neurocomputing
13. Wei Y, Zhang Z, Wang Y, Xu M, Yang Y, Yan S, Wang M (2021) Deraincyclegan: rain attentive cyclegan for single image deraining and rainmaking. IEEE Trans Image Process 30:4788–4801
14. Yang Y, Zhang W, He Z, Li D (2020) High-speed rail pole number recognition through deep representation and temporal redundancy. Neurocomputing 415:201–214
15. Zhang T, Tang Z, Wang Y, Zhang Z, Zhan C, Zha Z, Wang M (2021) Dense residual network: enhancing global dense feature flow for character recognition. Neural Netw 139:77–85
16. Lin TY, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, Perona P, Ramanan D, Zitnick CL, Dollár P (2014) Microsoft COCO: common Objects in Context. arxiv:1405.0312 Comment: 1) updated annotation pipeline description and figures; 2) added new section describing datasets splits; 3) updated author list
17. Everingham M, Gool LV, Williams CKI, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. Int J Comput Vis 88(2):303–338
18. Ahmed A, Tangri P, Panda A, Ramani D, Karmakar S (2019) Vfnet: a convolutional architecture for accent classification. IEEE Pattern Anal Mach Intell 39(6):1137–1149
19. Zhou X, Koltun V, Krhenbühl P (2021) Probabilistic two-stage detection
20. Girshick R (2015) Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp 1440–1448
21. Ren S, He K, Girshick R, Sun J (2017) Faster r-cnn: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149
22. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR), pp 779–788
61. Shao S, Zhao Z, Li B, Xiao T, Yu G, Zhang X, Sun J (2018) Crowdhuman: a benchmark for detecting human in a crowd. arXiv:1805.00123

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Chao Wang1 · Yunchu Zhang1 · Yanfei Zhou1 · Shaohan Sun1 · Hanyuan Zhang1 · Yepeng Wang2,3

Yunchu Zhang
yczhang@sdjzu.edu.cn
Chao Wang
2019080110@stu.sdjzu.edu.cn
Yanfei Zhou
yanfeizhou98@163.com
Shaohan Sun
ssh18366101826@163.com
Hanyuan Zhang
zhanghanyuan18@sdjzu.edu.cn
Yepeng Wang
wangyepeng89@126.com

1 Shandong Key Laboratory of Intelligent Buildings Technology, School of Information and Electrical Engineering, Shandong Jianzhu University, Jinan 250101, China
2 Shandong Transportation Planning and Design Institute Group Co. Ltd, Jinan 250098, China
3 Beijing Institute of Technology, Beijing 100081, China