Visual State Space Model in YOLO for Safety Helmet Wearing Detection

Junming Huang, Lingyan He

Abstract—**It is essential for individuals involved in construction engineering to be adequately protected by the use of safety helmets. In response to the challenges associated with false positives and missed detections in helmet-wearing detection within complex environments, this study introduces a hybrid architectural algorithm termed Mamba-YOLO, which integrates a state-space model with Convolutional Neural Networks (CNNs). Initially, we propose the CDown downsampling module, which combines pooling downsampling with convolutional downsampling. This method is employed in Mamba-YOLO to efficiently reduce the model's parameters while simultaneously enhancing its learning capabilities. Furthermore, we introduce the Lightweight-C2f module within Mamba-YOLO, specifically designed to improve the model's ability to perceive object scales by reusing feature information and expanding receptive fields. Additionally, we propose the Mamba-Head, characterized by a hybrid architecture that incorporates a state-space model. The Mamba-Head facilitates a global receptive field through the cross-scan module, thereby enhancing the model's sensitivity to global contextual information. Following this, we conduct a series of experiments to evaluate the model's performance, which includes ablation studies, comparative experiments, and assessments against state-of-the-art models. The results indicate that Mamba-YOLO demonstrates significant efficacy in the task of helmet-wearing detection.**

Index Terms—**State space model, Vmamba, YOLO, Helmet-wearing detection**

I. INTRODUCTION

The development of urbanization has led to an increased demand for the construction of modern facilities. However, the diverse environments of construction sites expose numerous workers to a variety of occupational hazards, including falls from heights, mechanical injuries, and impacts from falling objects. Such accidents not only pose a significant risk to the physical health of workers but also place a considerable burden on their families. Safety helmets are widely regarded as essential components of personal protective equipment for ensuring the safety of workers in hazardous environments [1]. The use of safety helmets by workers during the execution of their duties has been shown to be an effective measure for preventing or reducing accidents [2]. Nevertheless, due to negligence and other factors, there are occasional instances of workers forgetting to wear helmets or wearing them incorrectly. Consequently, numerous scholars have conducted extensive research on safety helmet-wearing detection. Given the non-intrusive nature of computer vision technology, the majority of studies in this field have been based on it.

In 2004, Wen et al. [3] employed the Hough transform method to detect the arc contours of safety helmets and subsequently inferred whether a helmet was being worn by the worker. However, reliance on contour characteristics alone renders this method susceptible to false detections. Furthermore, this approach requires high-definition indoor images, limiting its applicability in complex environments such as construction sites. To reduce the noise present in the images, Cai et al. [4] utilized a combination of threshold segmentation and morphological operations, specifically the open and closed operators, to preprocess the background of the images. They then established empirical parameters to assess the ratio of candidate regions to the minimum circumscribing circle, thereby facilitating the detection of miners' helmet-wearing status. In the study by Shrestha et al. [5] on the detection of helmet-wearing status among construction workers, Haar features were utilized to perform facial detection first, which were then combined with edge detection algorithms to analyze the contour and color of safety helmets. In 2015, Park et al. [6] employed histogram of oriented gradients (HOG) and support vector machines (SVM) technologies to detect humans and safety helmets. Subsequently, the geometric and spatial relationships between the human and helmet were matched in order to ascertain whether workers were wearing helmets. Rubaiyat et al. [7] extracted frequency domain information from images that were segmented by a discrete cosine transform (DCT)-based Gaussian mixture model and then extracted HOG features from the DCT coefficients. Subsequently, they employed SVM to ascertain the presence of interest objects and utilized the feature extraction method of color and circular Hough transform (CHT) to determine the helmet-wearing status of construction workers. In addition, Doungmala et al. [8] also proposed a method for the detection of safety helmets based on Haar features and CHT. In the study by Kang et al. [9], the ViBe background modelling algorithm was employed to detect moving objects within substations, and the C4 real-time human classification framework was then utilized to accurately locate them. Furthermore, according to the positioning results, the helmet-wearing detection was achieved through the head position, color space transformation, and color feature discrimination. Wu et al. [10] developed a color-based hybrid descriptor using local binary patterns (LBP), Hu moments

Manuscript received January 28, 2025; revised July 20, 2025.

This work was supported by Guangxi Science and Technology Program (Grant No. AD23026010).

Junming Huang is a lecturer at the School of Mechanical and Electrical Engineering, Guangxi Vocational College of Water Resources and Electric Power, Nanning, Guangxi 530023, China (Email: 46242@qq.com).

Lingyan He is a lecturer at the School of Mechanical and Electrical Engineering, Guangxi Vocational College of Water Resources and Electric Power, Nanning, Guangxi 530023, China (corresponding author, Email: 497228377@qq.com).

invariants (HMI), and color histograms (CH) to extract features of helmets with different colors. Subsequently, a hierarchical support vector machine (H-SVM) classifier was constructed for the purpose of detecting helmets, which achieved an average recognition rate of 90.3% on their private dataset. Jin et al. [11] employed the deformable part model (DPM) algorithm to extract worker regions and then utilized color space conversion and color feature matching techniques to isolate the helmet area. Subsequently, a combination of HOG and SVM was applied within the identified region to detect helmet-wearing.

While the traditional image processing methods mentioned earlier can initially detect the wearing of safety helmets, they are limited in their ability to manage complex construction environments and objects of varying scales. To address these limitations and further enhance the reliability and practicality of safety helmet detection methods, many researchers have begun to focus on studies utilizing deep learning algorithms.

In the study on helmet-wearing state detection conducted by Li et al. [12], an enhanced Faster R-CNN algorithm was employed to identify both the helmet and its wearer. Subsequently, the geometric relationships between these elements were utilized to determine the state of helmet-wearing. Wang et al. [13] improved the backbone of YOLOv3 by integrating cross-stage partial networks (CSPNet) and spatial pyramid pooling (SPP) architectures, resulting in a significant increase in accuracy compared to the original algorithm. However, the uniform scale of the objects in the datasets used for training limited the model's sensitivity to multi-scale objects. In the research by Gu et al. [14], a three-point positioning method and skin color detection were used to identify the head regions of construction workers, while YOLOv4 was employed to detect the helmet regions. An evaluation of the helmet-wearing states among construction workers was subsequently conducted by examining the intersection of the helmet and head regions. Zhou et al. [15] proposed an attention mechanism-based helmet detection algorithm, referred to as AT-YOLO. First, channel attention modules and spatial attention modules were integrated into the backbone and neck networks of YOLOv3, thereby enhancing the network's feature perception. Second, the DIoU (Distance Intersection over Union) bounding box regression loss function was utilized to accelerate network training convergence while improving detection capabilities for small objects. The experimental results demonstrated that the improved algorithm achieved a high mean Average Precision (mAP). Jin et al. [16] further enhanced the YOLOv3 model for helmet detection. Initially, the K-means++ algorithm was applied to improve the size matching of prior anchor boxes. Subsequently, the depth-wise coordination attention (DWCA) mechanism was incorporated into the backbone network, enhancing the model's ability to distinguish between foreground and background. In a private dataset, the improved algorithm exhibited a 3% increase in mAP compared to YOLOv5. Based on the Single Shot Multibox Detector (SSD) framework, Han et al. [17] proposed an enhanced object detection algorithm that significantly improved the precision of helmet detection. This improved algorithm refined the feature information of target regions by applying spatial attention mechanisms to low-level features and channel attention mechanisms to high-level features, respectively. Additionally, a feature pyramid network (FPN) and a multi-scale perception module were introduced to bolster the algorithm's robustness in detecting multi-scale objects. An adaptive adjustment method for anchor boxes was also designed according to the scale distribution of anchors across layers. By regarding helmet-wearing detection as a task involving strong semantic keypoint detection, Song et al. [18] proposed a novel anchor-free object detection model, named as the reciprocal bidirectional feature pyramid detector (RBFPDet), which can achieve almost real-time detection in complex backgrounds. In the study of Liu et al. [19], a spatial position relation capsule network (SPRCapsNet) was employed to discern whether the helmet was properly positioned relative to the face and then to detect the helmet-wearing status. In order to reduce the computational cost, the algorithm implemented a segmentation of the deep feature maps into smaller patches, which were then transformed into vectors that served as the primary capsules. Subsequently, a dynamic routing algorithm was adopted to learn the spatial relationships between local image features. Finally, a decision optimization process was conducted based on the probability of different dimensions appearing in the output vector. Lee et al. [20] proposed a combined model, named YOLO-EfficientNet, which employed YOLOv5x for the purpose of detecting heads and utilized EfficientNet for the head state classification. The model could achieve high accuracy even with limited training data. Xiang et al. [21] constructed a safety helmet detection network based on a multi-scale Swin Transformer, and obtained a superior performance on the Pictor-v3 and SHWD datasets.

To improve the effectiveness of object detection algorithms in complex backgrounds, we propose a hybrid architecture algorithm called Mamba-YOLO, designed to detect helmet-wearing states. Mamba-YOLO is based on the YOLOv8 framework and integrates state space models with CNNs. The primary contributions of our research are as follows:

(1) A downsampling method, referred to as CDown, has been developed by integrating pooling downsampling and convolutional downsampling techniques. This CDown method effectively reduces the number of parameters in the model while simultaneously enhancing its learning capacity.

(2) A module named Lightweight-C2f has been developed from C2f to enhance the model's perception of object scales by reusing feature information and increasing receptive fields.

(3) A hybrid architecture-based head, called Mamba-Head, has been proposed by integrating the state space model into the detection head of the baseline. The Mamba-Head allows the model to possess a global receptive field, thereby improving its sensitivity to global contextual information.

II. Methods

YOLOv8 represents a notable advancement in the YOLO (You Only Look Once) series of object detection algorithms. In comparison to its predecessors, it exhibits enhanced accuracy while maintaining real-time performance [22]. As shown as Fig. 1, the YOLOv8 model is comprised of three principal components: Backbone, Neck, and Head. The Backbone is responsible for extracting image features, which

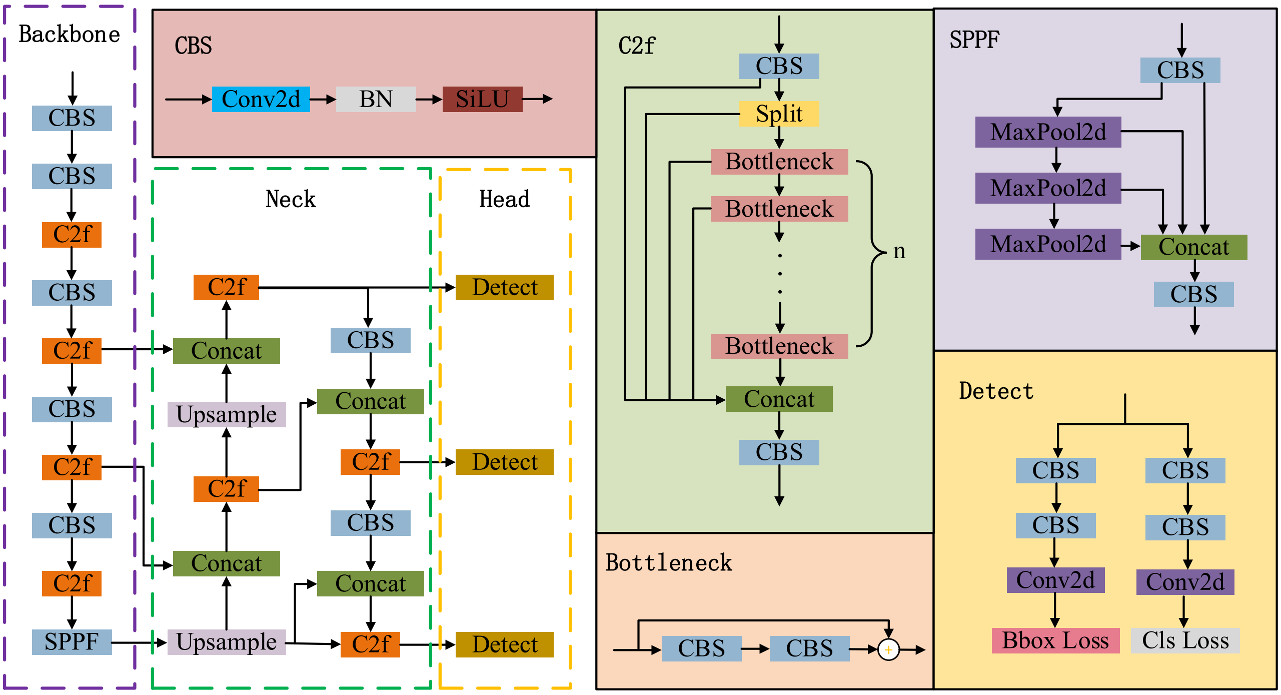


Fig. 1. YOLOv8 model architecture.

will then be subjected to fusion at different scales by the Neck. The function of the Head is to predict the categories and locations of objects. YOLOv8 is characterized by an efficient architectural design that facilitates enhanced speeds and more precise detection outcomes, accompanied by noteworthy generalization performance [23].

To further enhance the accuracy of safety helmet-wearing detection in complex backgrounds, YOLOv8 is employed as the baseline, and a series of improvement experiments will be conducted.

A. CDown

In deep learning networks, downsampling is commonly employed to improve feature hierarchies by decreasing the spatial resolution of feature maps [24]. The process of downsampling not only allows the model to concentrate on more abstract and high-level visual features but also significantly reduces the number of parameters and the computational burden in subsequent layers. Consequently, this approach enhances both the efficiency and performance of the network.

Downsampling methods used in deep learning networks can be classified into two categories: pooling downsampling and convolutional downsampling. Pooling downsampling is a non-linear dimensionality reduction technique that provides the advantage of translation invariance while reducing the number of network parameters, thereby decreasing the risk of overfitting [25]. However, this method may also lead to the loss of some spatial information. In contrast, convolutional downsampling reduces the size of feature maps by employing convolution operations with a stride greater than one. The advantage of convolutional downsampling is that it preserves more spatial information through a weight-learning process. Furthermore, by adjusting the weights of the convolutional kernels, the model can learn to perform feature selection and information compression optimally. Nevertheless, convolutional downsampling may increase model complexity and training difficulty.

To reduce the complexity and parameter count of the model while effectively preserving its learning capabilities, we propose a downsampling method called Collaborative Downsampling (CDown), as illustrated in Fig. 2.

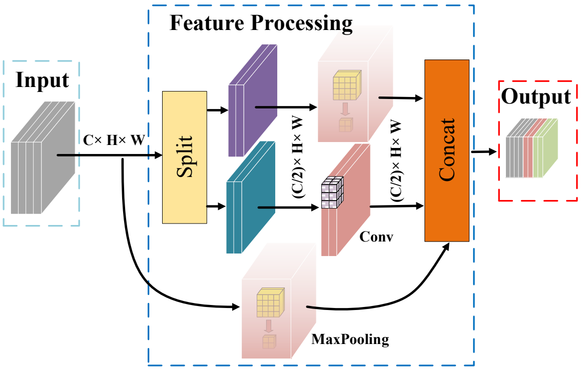


Fig. 2. Structure of the CDown module.

The CDown downsampling method is employed as an alternative to the traditional convolutional downsampling method used in the baseline. As illustrated in Fig. 2, the input feature map is initially divided into two segments along the channel dimension. One segment is downsampled using the max pooling method, while the other segment is downsampled through a convolution operation to preserve the model's ability to learn the target features. Additionally, the original input feature map is directly processed with maximum pooling. Finally, the results of these three processes are concatenated along the channel dimension. In contrast to the traditional convolutional downsampling approach, the CDown method effectively reduces the model's parameter count by performing convolutional calculations on only a subset of the channels. Simultaneously, the direct application of the max pooling method to the input feature map, followed by its concatenation with the results of both the max pooling and convolutional downsampling methods, not only increases the number of channels but also promotes

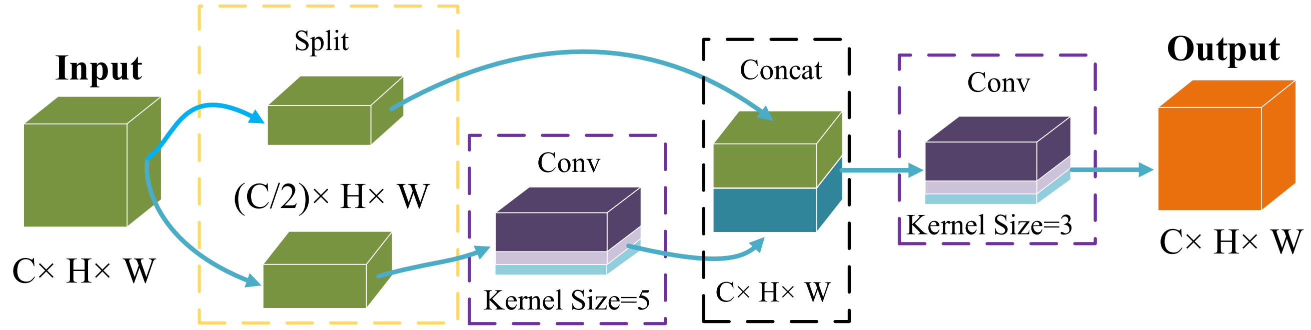


Fig. 3. Structure of the Lightweight-Bottleneck.

the efficient reuse of feature information, thereby enhancing the model's learning capacity.

B. *Lightweight-C2f*

As illustrated in Fig. 1, the C2f module represents a novel bottleneck layer design that is utilized in the baseline to enhance the model's efficiency and performance [26]. The C2f module improves the model's feature perception by concatenating the outputs of various bottleneck modules with the original feature map, thereby allowing the network to learn richer multi-scale information [27]. Additionally, the concatenation of multiple bottleneck modules enables the C2f to maintain a lightweight structure while acquiring more comprehensive gradient flow information, which facilitates faster model convergence.

To further streamline the model and reduce its parameter count and complexity, a Lightweight-C2f module derived from the C2f module is proposed as a replacement. The bottleneck modules of C2f have been modified based on the design philosophy of CSPNet. First, the input feature map is split along the channel dimension. Next, only a subset of the split feature map undergoes convolutional operations, while the remaining portion remains unprocessed. The unprocessed and convoluted portions are then concatenated along the channel dimension. Finally, a novel Lightweight-Bottleneck is created by replacing the first convolutional layer of the C2f bottleneck with this new structure, as illustrated in Fig. 3.

The Lightweight-Bottleneck significantly reduces redundant gradient information, thereby improving training efficiency and decreasing computational complexity. Additionally, by splitting and reusing feature map information, it enhances the network's ability to learn features. Notably, the convolution branch in the Lightweight-Bottleneck employs a kernel size of 5, whereas the maximum convolution kernel size of C2f is 3. As illustrated in Fig. 4, the larger convolution kernel provides an expanded receptive field, enabling neurons to capture information over a broader area and facilitating a more comprehensive understanding of the global structure and contextual information present in images [28].

C. *YOLO head with vision Mamba*

Mamba is a state-space model designed to effectively manage long sequence modeling tasks [29]. By integrating global receptive fields and dynamic weighting, Mamba overcomes the modeling limitations associated with CNNs and enhances the model's capabilities. To improve the model's ability to perceive variations in target scale, several convolutional layers in the YOLO head are replaced with the VSS block from the Mamba model [30]. Consequently, a hybrid Mamba-CNN architecture is proposed for the detection head, referred to as Mamba-Head, as illustrated in Fig. 5.

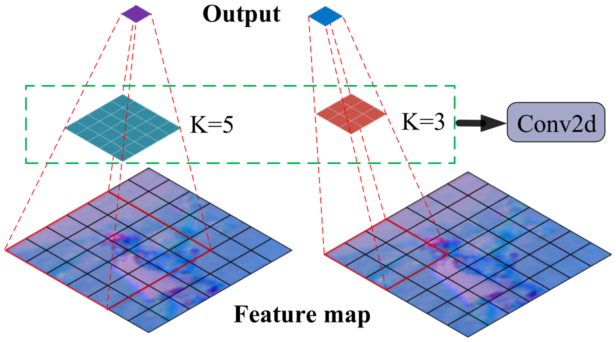


Fig. 4. The impact of different convolution kernel sizes on the receptive field.

A state space model is used to describe and analyze the behavior of a dynamic system [31]. It can map the system's input $x(t)\in R^L$ to the response $y(t)\in R^L$. Mathematically, the state space model is typically represented by a set of differential equations, as shown as (1).

$$\begin{aligned} h'(t) &= Ah(t) + Bx(t) \\ y(t) &= Ch(t) + Dx(t) \end{aligned} \tag{1}$$

where, $h(t)$ represents the system state vector, $x(t)$ represents the system input vector, $y(t)$ represents the system output vector, $A\in C^{N\times N}$ represents the state transition matrix, $B\in C^N$ and $C\in C^N$ represent the input matrix and output matrix respectively, $D\in C^1$ represents the direct transfer matrix, and N is the number of variables in the state space.

In the field of deep learning, state space models are employed to handle sequential data. By mapping sequential data into the state space, state space models can more effectively capture long-term dependencies in the data. In order to facilitate the processing of discrete sequential data by state space models, the input $x_k \in R^{L\times D}$ is treated as a sequence of length L with D-dimensional signals. This leads to the discretization of (1), and the results of discretization could be illustrated as (2).

$$\begin{gathered} h_k = \bar{A}h_{k-1} + \bar{B}x_k \\ y_k = \bar{C}h_k + \bar{D}x_k \\ \bar{A} = e^{\Delta A} \\ B = (e^{\Delta A} - I)A^{(-1)}B \\ \bar{C} = C \end{gathered} \tag{2}$$

where, $B \in R^{D\times N}$, $C \in R^{D\times N}$ and $\Delta \in R^D$.

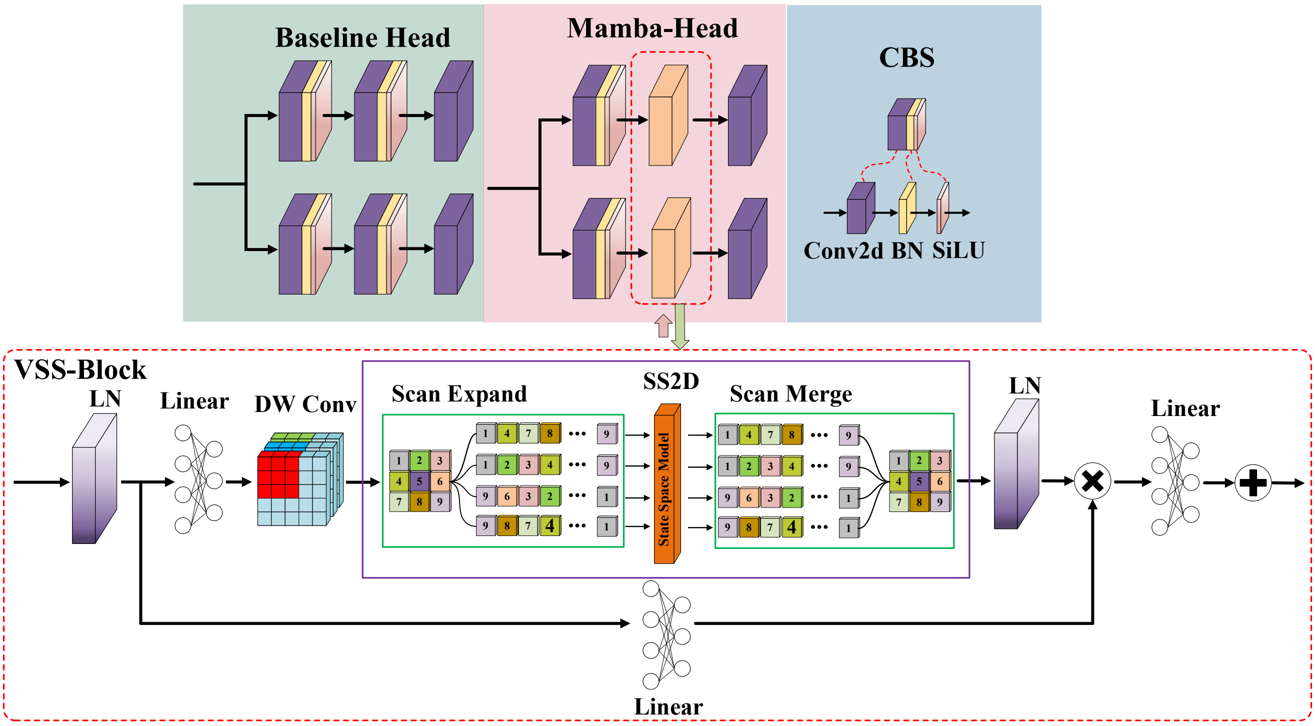


Fig. 5. Structure of the Mamba-Head.

Actually, the $\overline{B}$ in (2) is usually linearly approximated by a first-order Taylor expansion, as illustrated in (3).

$$\overline{B} = (e^{\Delta A} - I)A^{-1}B \approx (\Delta A)(\Delta A)^{-1}\Delta B = \Delta B \tag{3}$$

The information processed by CNNs is typically represented as 2D feature maps, which do not naturally align with the sequential data processing of state space models [32]. Therefore, it is necessary to serialize the information contained in these 2D feature maps. However, 2D feature maps exhibit non-causal characteristics, and directly flattening them for sequential scanning can lead to a loss of the global receptive field. To address this challenge and enable the model to maintain a global receptive field, a 2D selective scanning method known as the Cross Scan Module (CSM) is employed in the Mamba model [33]. The CSM initiates a scan from the four corner pixels of an image and then proceeds to move in various directions. Subsequently, the results of the cross-scanning are serialized, facilitating selective scanning through a state space model. Finally, the scanned data is reconstructed into an image, as illustrated in Fig. 6.

III. Experiment Results

A. Datasets

To validate the effectiveness of the presented work, the model was trained and evaluated using the open-source SHWD dataset. This dataset provides data for the purposes of safety helmet-wearing and human head detection. It consists of 7,581 images, with 9,044 instances depicting individuals wearing safety helmets and 111,514 instances showing heads without helmets. The distribution of samples across different scales within various subsets is presented in TABLE I and TABLE II, respectively.

TABLE I
THE DISTRIBUTION OF SAMPLE NUMBERS

Label	Train	Val	Test
hat	6419	747	1878
person	79778	9178	22558

TABLE II
THE NUMBER OF SAMPLES ACROSS DIFFERENT SCALES WITHIN VARIOUS SUBSETS

Scale	Train	Val	Test
small(area≤32×32)	59845	7069	17027
medium(32×32＜area≤96×96)	19014	2141	5462
large(area＞96×96)	7338	715	1947

To improve the model's generalization ability and robustness, the data augmentation strategies used in the baseline are also applied in this study. These strategies include mosaic augmentation, random horizontal flipping, random vertical flipping, and color jittering. Notably, the

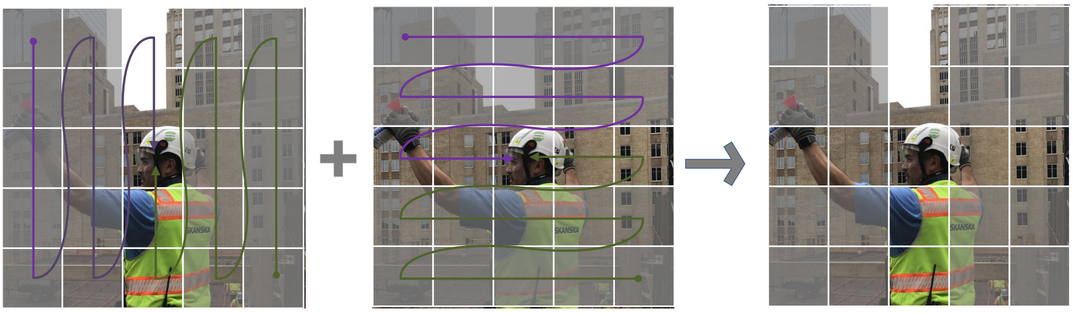


Fig. 6. 2D selective scanning method.

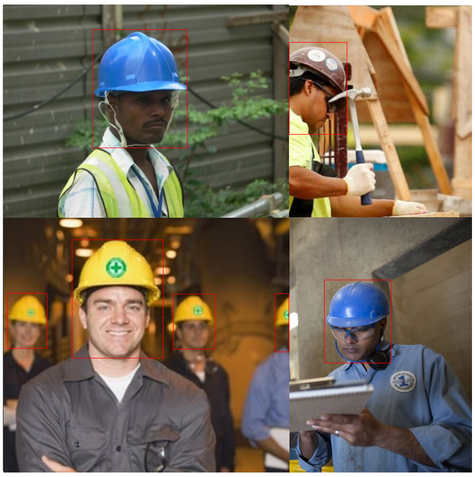


Fig. 7. Mosaic data augmentation strategy.

mosaic data augmentation technique combines four randomly selected images into a single composite image, as illustrated in Fig. 7. This method significantly enhances the background of the targets and balances the distribution among targets of varying scales.

B. *Platform and trainning*

In order to ensure the reproducibility and impartiality of the experimental results, a comprehensive list of the hardware and software environments used in the experiments is presented in TABLE III.

TABLE III
THE HARDWARE AND SOFTWARE ENVIRONMENTS

Options	Configuration
OS	Ubuntu 18.04
CPU	Intel(R) Xeon(R) Platinum 8352V
GPU	Nvidia RTX 4090 24G
Framework	Pytorch 2.0
Language	Python 3.8

To ensure the effectiveness of model training, this study adjusted several key hyperparameters throughout the training process. The hyperparameters employed in this research are presented in TABLE IV. Notably, the early stopping patience is set to 50 epochs, in accordance with the early stopping strategy. This means that if the model's accuracy does not show improvement within the specified 50 epochs, the training will automatically conclude. This approach not only prevents the model from overfitting but also effectively conserves computational resources.

TABLE IV
MODEL TRAINING HYPERPARAMETER SETTINGS

Training options	Setting
Input image size	640*640
Lr0	0.01
Lrf	0.01
Lr scheduler	LinearLR
Momentum	0.937
Batch size	16
Optimizer	SGD
Epochs	300
Early stopping patience	50

In order to evaluate the effectiveness of model training, three types of loss values are typically employed: classification loss, bounding box regression loss, and distribution focal loss [34]. These three types of loss can be derived from equations (4) to (6).

$$\mathrm{VFL}(p,q)=\begin{cases}-q(q(log(p)+(1-q)log(1-p)) \\ \qquad -\alpha p^{\gamma} log(1-p) \\ \qquad\qquad q>0 \\ \qquad\qquad q=0\end{cases} \tag{4}$$

where, q represents the true class of the samples, p denotes the predicted probability, α is the weight factor for positive samples, and γ is the modulating factor.

$$\mathcal{L}_{CIoU}=1-IoU+\frac{\rho^2(b,b^{gt})}{c^2}+\alpha v \tag{5}$$

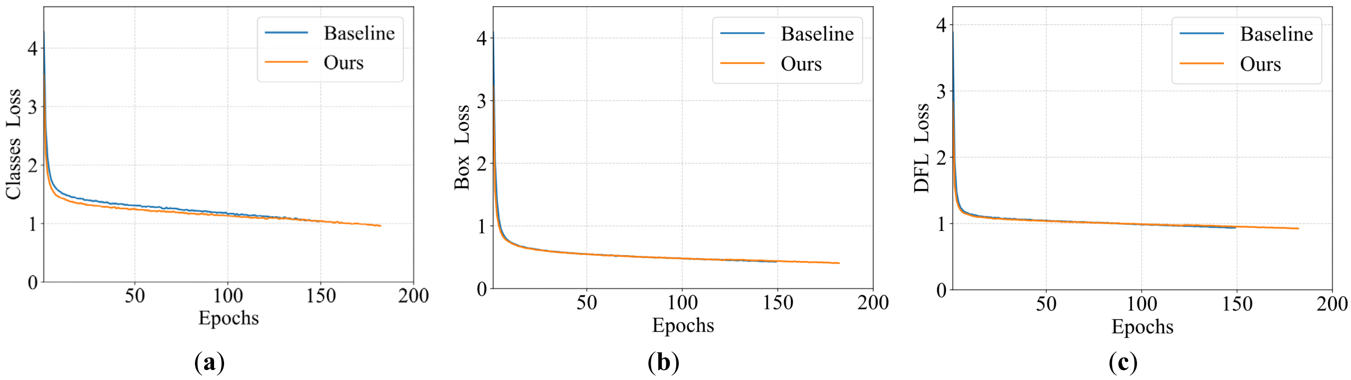
where, IoU is the intersection over union of the ground truth box and the predicted box, ρ is the Euclidean distance, b and b^{gt} respectively represent the distances from the centers of the ground truth and predicted boxes, and v is a measure of the consistency of the aspect ratios.

$$DFL(S_i,S_{i+1})=-((y_{i+1}-y)log(S_i)+(y-y_i)log(S_{i+1}))$$

$$S_i=\frac{y_{i+1}-y}{y_{i+1}-y_i},S_{i+1}=\frac{y-y_i}{y_{i+1}-y_i} \tag{6}$$

where, S_i and S_{i+1} respectively represent the predicted values output by the network and the adjacent predicted value; y, y_i, and y_{i+1} respectively represent the actual label value, the integrated label value, and the adjacent integrated label value.

The training results are presented in Fig. 8. It is evident that the three types of loss values for the model decrease rapidly before gradually stabilizing. The iterative process ultimately concluded due to the implementation of the early stopping strategy.



(a)

(b)

(c)

Fig. 8. Comparison of training loss curves between Mamba-YOLO and baseline. (a) classification loss, (b) bounding box regression loss, (c) distribution focal loss.

C. Main results

Evaluation metrics

The evaluation metrics, including precision, recall, and mean Average Precision at IoU 0.5 (mAP_{50}), are employed to rigorously assess the model's performance. Precision is defined as the ratio of the area predicted by the algorithm to the actual detection area, while recall represents the proportion of correctly predicted categories out of the total number of required categories. The mAP_{50} indicates the average precision across all samples, where the overlap between the predicted bounding box and the ground truth bounding box is at least 50% of the total area of the two boxes. A higher mAP_{50} value signifies greater prediction accuracy. These metrics are crucial for comparing the performance of different models, as they reflect the reliability of the models from various perspectives. The evaluation metrics are illustrated in equation (7).

$$
\begin{aligned}
Precision &= \frac{TP+FP}{TP} \\
Recall &= \frac{TP+FN}{TP} \\
F1 &= 2\times\frac{Precision\times Recall}{Precision+Recall} \\
AP &= \int_0^1 Precision(r)dr \\
mAP_{50} &= \sum_{i=1}^{c} AP_i
\end{aligned}
\qquad (7)
$$

where, TP represents the number of true positive samples, FP represents the number of false positive samples, FN represents the number of false negative samples, AP represents the average precision, r denotes recall, C is the total number of classes, and i indicates a specific class.

Comparison of ablation experiments

Ablation research evaluates the influence of each component on overall performance by progressively removing or modifying different parts of the model. This approach not only validates the effectiveness of individual modules but also provides a solid practical foundation for future research. To gain a deeper understanding of the specific contributions of the various improvement modules proposed in this work to the model's performance, ablation experiments were conducted using the SHWD open-source dataset. The results are detailed in Table V.

The implementation of the CDown module in isolation results in a reduction of model parameters and computations by 4.5 M and 14.2 GFLOPs, respectively, compared to the baseline model. Although there is a slight decline in the F1 score, the mAP50 metric shows a 0.1% improvement. This outcome indicates that the CDown module can decrease the computational complexity and parameter quantity of the model while maintaining the model's feature learning capability. Furthermore, the integration of the Lightweight-C2f module results in a substantial enhancement in both the mAP50 and F1 score comparison to the baseline. This enhancement is accompanied by a reduction in model parameters by 2.9 M and a decrease in computational resources by 14.7 GFLOPs. This observation suggests that the feature reuse strategy employed by the C2f module not only contributes to the reduction in model complexity, but also leads to a notable enhancement in the model's feature extraction capability. Furthermore, the integration of the Mamba-Head module into the model alone led to substantial enhancements in both mAP50 and F1 score, despite a marginal increase in parameters and computational cost. This suggests that the Mamba-Head module effectively enhances the model's capacity to discern target features by preserving the global receptive field.

When any two of these modules are randomly combined and applied to the model, the mAP_{50} of the enhanced model demonstrates a notable improvement over the baseline model, while simultaneously reducing both the number of parameters and computational costs. The simultaneous application of all three proposed modules to the model, referred to as Mamba-YOLO, yields the highest mAP_{50} score. Additionally, both the computational costs and the number of parameters are significantly decreased. In summary, the improved method presented in this paper not only enhances the feature learning capability of the model but also effectively reduces its complexity.

In order to visually assess the improvement in target feature perception capabilities of our work compared to the baseline, we conducted a visual comparison of the inference results, as illustrated in Fig. 9. The first set of comparison images shows that Mamba-YOLO can accurately detect heavily occluded targets, whereas the baseline fails to register two instances of such targets. As demonstrated by the second and third sets of comparison images, the baseline model generates several false positives for small targets in complex backgrounds. In contrast, the Mamba-YOLO model effectively distinguishes between targets and the background, successfully completing the detections.

TABLE V
RESULTS OF ABLATION EXPERIMENTS

Model *				P (%)	R (%)	mAP_{50} (%)	F1	Parameters (M)	GFLOPs
B	C	L	M						
√				92.6	90.8	94.6	91.7	43.0	165.7
√	√			92.0	90.6	94.7	91.3	38.5	151.5
√		√		92.8	90.8	95.2	91.8	40.1	151.0
√			√	92.8	91.2	95.1	92.0	45.4	171.4
√	√	√		92.7	90.9	95.2	91.8	34.8	136.8
√	√		√	91.8	91.1	94.7	91.5	40.1	157.2
√		√	√	92.8	90.6	94.7	91.7	41.8	156.6
√	√	√	√	91.4	92.0	95.4	91.7	36.4	142.0

*: B stands for the baseline model; C stands for the CDown module; L stands for the Lightweight-C2f module; M stands for the Mamba-Head module.

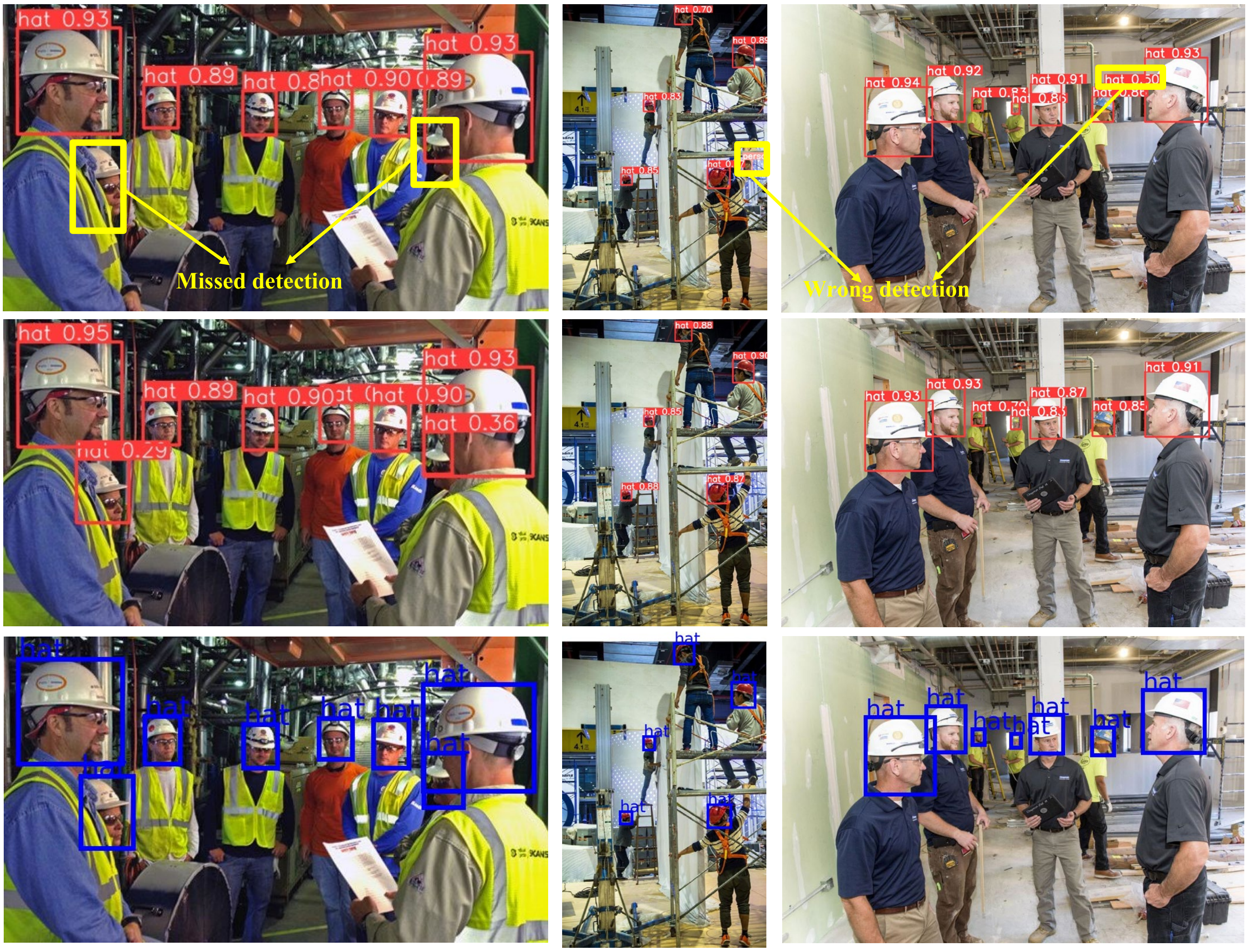


Fig. 9. Visual comparison of inference results between baseline and Mamba-YOLO. The first line represents the baseline inference results, the second line represents the Mamba-YOLO inference results, and the third line represents the manually labeled labels.

Experiment with multiple datasets

In order to further validate the generalization ability and robustness of Mamba-YOLO across various datasets, Mamba-YOLO was also trained and evaluated on the GDUT-SHWD and Helmet Detection datasets following the ablation experiments on the SHWD dataset. By testing on a diverse array of datasets, we can conduct a more comprehensive assessment of the model's performance and ensure its effectiveness in a wide range of practical applications. The results of the experiments conducted with Mamba-YOLO and the baseline across different datasets are presented in Table VI.

As demonstrated in Table VI, compared to the baseline, Mamba-YOLO achieves improvements of 1.5% and 1% in mAP_{50} on the GDUT-SHWD and Helmet Detection datasets, respectively. Furthermore, the F1 score increases by 0.9 and 0.7 on these two datasets, respectively. The experimental results across multiple datasets indicate that the proposed method performs well on individual datasets and also exhibits strong generalization performance and robustness across diverse datasets.

In order to comprehensively assess the model's genuine capacity for generalization, cross-domain training, validation, and testing were performed utilizing open-source datasets from diverse fields. This multi-domain evaluation experiment aims to identify the model's susceptibilities to variations in real-world environments. Specifically, datasets from three unrelated domains were utilized for experimentation: agronomy, zoology, and aeronautical science. The experimental results are summarized in Table VII.

As demonstrated in Table VII, the performance of the Mamba-YOLO model and the baseline model shows significant variations across datasets from different domains. Mamba-YOLO exhibits a substantial improvement in performance when evaluated on the WeedCrop and Aircraft Detection datasets, as evidenced by notable increases in mAP_{50}. The Mamba-YOLO achieves enhancements of 1.4%

TABLE VI
RESULTS OF EXPERIMENT WITH MULTIPLE DATASETS

Datasets	Model	P (%)	R (%)	mAP_{50}(%)	F1
SHWD	Baseline	92.6	90.8	94.6	91.7
	Mamba-YOLO	91.4	92.0	95.4	91.7
GDUT-SHWD	Baseline	89.3	81.2	88.1	85.0
	Mamba-YOLO	89.4	82.7	89.6	85.9
Helmet Detection	Baseline	92.2	84.3	89.7	88.1
	Mamba-YOLO	92.9	85.1	90.7	88.8

TABLE VII
RESULTS OF EXPERIMENT WITH MULTIPLE DATASETS

Datasets	Model	P (%)	R (%)	mAP_{50}(%)	F1
WeedCrop	Baseline	71.1	75.8	72.9	73.4
	Mamba-YOLO	79.4	68.6	74.3	73.6
African Wildlife	Baseline	92.8	90.0	96.4	91.4
	Mamba-YOLO	95.9	90.4	96.6	93.1
Aircraft Detection	Baseline	83.4	70.7	77.7	76.5
	Mamba-YOLO	94.5	73.2	82.7	82.5

TABLE VIII
PERFORMANCE COMPARISON OF MAMBA-YOLO AND OTHER ALGORITHMS

Models	P (%)	R (%)	mAP_{50} (%)	F1	Parameter (M)	GFLOPs
Rt-Detr	87.9	85.0	90.5	86.4	32.8	108.0
yolov9c	92.7	90.5	94.5	91.6	25.3	102.3
yolov6l	92.0	90.0	93.9	91.0	110.8	391.2
yolov5	92.0	91.0	94.7	91.5	53.2	135.3
yolov3	93.1	90.2	94.4	91.6	103.6	283.0
Mamba-YOLO	91.4	92.0	95.4	91.7	36.4	142.0

and 5.0% in mAP_{50} for these two datasets, respectively. In contrast, no significant difference is observed between the two methods on the African Wildlife dataset regarding mAP_{50}. With regard to the F1 score, the Mamba-YOLO consistently yields higher results than the baseline across all three datasets. The enhancement exhibited by the WeedCrop dataset is negligible, as evidenced by a mere 0.2 increase. It is important to note that the Mamba-YOLO exceeds the baseline by 1.7 and 6.0, respectively, on the African Wildlife and Aircraft Detection datasets. The collective results of these experiments indicate that Mamba-YOLO demonstrates superior generalization ability compared to the baseline across various open-source datasets from diverse domains.

Compared with different algorithms

To further evaluate the performance of Mamba-YOLO, five additional algorithms were selected for comparative experiments: YOLOv3, YOLOv5, YOLOv6, RT-DETR, and YOLOv9. The comparative experiments utilized the SHWD open-source dataset and were conducted on the same hardware for both training and evaluation. All models underwent comprehensive training, and the optimal weights were chosen for testing. The comparison data are presented in Table VIII. Mamba-YOLO achieves the highest score in the mAP_{50} metric, outperforming YOLOv3, YOLOv5, YOLOv6, RT-DETR, and YOLOv9 by 1%, 0.7%, 1.5%, 0.9%, and 4.9%, respectively. Furthermore, Mamba-YOLO also attains the highest F1 score, with improvements of 5.3, 0.1, 0.7, 0.2, and 0.1 over the aforementioned models, respectively. Through comprehensive comparisons with multiple state-of-the-art models, Mamba-YOLO demonstrates significant advantages in the mAP_{50} metric, underscoring its generalizability and practical value.

In contrast to traditional downsampling methods, the CDown module is more lightweight and effectively reduces the computational and parameter costs of the model. Furthermore, a Lightweight-C2f module has been introduced to decrease the number of parameters and computations while increasing the model's receptive field, owing to its capacity for feature information reuse. Additionally, state space models have been integrated into the CNNs, resulting in a hybrid architecture detection head known as Mamba-Head. This integration enables the model to possess a global receptive field, thereby enhancing its ability to perceive multi-scale target information. The main conclusions drawn from a series of experiments are as follows:

(1) Ablation experiments on the SHWD dataset were conducted to evaluate the performance of the proposed CDown, Lightweight-C2f, and Mamba-Head modules. Compared to the baseline, these modules demonstrated a range of improvements in the mAP_{50} metric. Additionally, the Mamba-YOLO algorithm shows an increase of 0.8% in the mAP_{50} metric relative to the baseline.

(2) To validate the generalization performance and assess the practicality of the model, Mamba-YOLO is compared with the baseline across multiple datasets. The experiments demonstrate that Mamba-YOLO achieved increases of 1.5% and 1% in the mAP_{50} metric on the GDUT-SHWD and Helmet Detection datasets, respectively, compared to the baseline.

(3) To validate the generalization ability and assess the practical utility of the proposed method, Mamba-YOLO is evaluated against the baseline on multi-domain datasets. Experimental results indicate that Mamba-YOLO achieves improvements of 1.4%, 0.2%, and 5.0% in mAP_{50} on the WeedCrop, African Wildlife, and Aircraft Detection datasets, respectively, compared to the baseline.

(4) To further validate the effectiveness of our work, we conducted a comparative analysis with state-of-the-art algorithms. The experiments demonstrated that our approach outperformed YOLOv3, YOLOv5, YOLOv6, Rt-Detr, and YOLOv9c by 1%, 0.7%, 1.5%, 0.9%, and 4.9%, respectively, in the mAP_{50} metric.

The experimental results demonstrate that the proposed hybrid architecture algorithm, Mamba-YOLO, which integrates a state space model with CNNs, can more accurately detect the status of safety helmet usage in complex backgrounds. In summary, Mamba-YOLO exhibits significant potential for application in safety helmet-wearing detection.

IV. Conclusion

In this paper, we propose the Mamba-YOLO model to enhance safety helmet-wearing detection accuracy in complex construction scenarios by integrating a hybrid architecture of convolutional neural networks and state-space models. Key innovations include the CDown downsampling method for parameter reduction, the Lightweight-C2f module for multi-scale feature perception, and the Mamba-Head with cross-scan modules for global context modeling. Comprehensive experiments, including ablation studies, multi-dataset comparisons, and multi-algorithm evaluations, demonstrate that Mamba-YOLO significantly improves detection robustness while reducing false positives and missed detections. This framework provides an effective solution for real-world safety helmet-wearing detection, with potential applications extending to broader industrial safety systems. Future research will focus on integrating this technology with industrial IoT platforms for real-time analytics and proactive risk prevention.

Data Availability

GDUT-SHW: https://github.com/yudaprama/hardhat-wearing-detection;
HelmetDetect: https://aistudio.baidu.com/datasetdetail/50329;
SHWD: https://github.com/njvisionpower/Safety-Helmet-Wearing-Dataset;
Code is available at: https://github.com/Carter007gx/MambaYOLO.git

REFERENCES

[1] Xudong Song, Tiankai Zhang, and Weiguo Yi, "An improved YOLOv8 safety helmet wearing detection network," Scientific Reports, vol. 14, no. 1, p. 17550, 2024.

[2] Yayun Wang, Ye Tao, Wenhua Cui, and Lijia Shen, "A steel surface defect detection model based on YOLOv7-tiny," IAENG International Journal of Computer Science, vol. 51, no. 12, pp. 2074-2082, 2024.

[3] Che-Yen Wen, "The safety helmet detection technology and its application to the surveillance system," Journal of Forensic Sciences, vol. 49, no. 4, pp. 770-780, 2004.

[4] Limei Cai, and Jiansheng Qian, "A method for detecting miners in underground coal mine videos," 2009 Second International Symposium on Computational Intelligence and Design, Changsha, China, vol. 2, pp. 127-130, 2009.

[5] Kishor Shrestha, Pramen P. Shrestha, Dinesh Bajracharya, and Evangelos A. Yfantis, "Hard‐hat detection for construction safety visualization," Journal of Construction Engineering, vol. 2015, no. 1, p. 721380, 2015.

[6] Man-Woo Park, Nehad Elsafty, and Zhenhua Zhu, "Hardhat-wearing detection for enhancing on-site safety of construction workers," Journal of Construction Engineering and Management, vol. 141, no. 9, p. 04015024, 2015.

[7] Abu H. M. Rubaiyat, Tanjin T. Toma, Masoumeh Kalantari-Khandani, Syed A. Rahman, Lingwei Chen, Yanfang Ye, and Christopher S. Pan, "Automatic detection of helmet uses for construction safety," 2016 IEEE/WIC/ACM International Conference on Web Intelligence Workshops (WIW), Omaha, NE, USA, pp. 135-142, 2016.

[8] Pathasu Doungmala, and Katanyoo Klubsuwan, "Helmet wearing detection in Thailand using Haar like feature and circle hough transform on image processing," 2016 IEEE International Conference on Computer and Information Technology (CIT), Nadi, Fiji, pp. 611-614, 2016.

[9] Kang Li, Xiaoguang Zhao, Jiang Bian, and Min Tan, "Automatic safety helmet wearing detection," 2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Honolulu, HI, USA, pp. 617-622, 2017.

[10] Hao Wu, and Jinsong Zhao, "An intelligent vision-based approach for helmet identification for work safety," Computers in Industry, vol. 100, pp. 267-277, 2018.

[11] Miao Jin, Jun Zhang, Xiwen Chen, Quan Wang, Bing Lu, and Wei Zhou, "Safety helmet detection algorithm based on color and hog features," 2020 IEEE 19th International Conference on Cognitive Informatics & Cognitive Computing (ICCICC)*, Beijing, China, pp. 215-219, 2020.

[12] Ning Li, Xin Lyu, Shoukun Xu, Yaru Wang, Yusheng Wang, and Yuwan Gu, "Incorporate online hard example mining and multi-part combination into automatic safety helmet wearing detection," IEEE Access, vol. 9, pp. 139536-139543, 2020.

[13] Haikuan Wang, Zhaoyan Hu, Yuanjun Guo, Zhile Yang, Feixiang Zhou, and Peng Xu, "A real-time safety helmet wearing detection approach based on CSYOLOv3," Applied Sciences, vol. 10, no. 19, p. 6732, 2020.

[14] Yuwan Gu, Yusheng Wang, Lin Shi, Ning Li, Lihua Zhuang, and Shoukun Xu, "Automatic detection of safety helmet wearing based on head region location," IET Image Processing, vol. 15, no. 10, pp. 2441-2453, 2021.

[15] Qingyang Zhou, Jiaohua Qin, Xuyu Xiang, Yun Tan, and Neal N. Xiong, "Algorithm of helmet wearing detection based on AT-YOLO deep mode," Computers, Materials & Continua, vol. 69, no. 1, pp. 159-174, 2021.

[16] Zhang Jin, Peiqi Qu, Cheng Sun, Meng Luo, Yan Gui, Jianming Zhang, and Hong Liu, "DWCA‐YOLOv5: an improve single shot detector for safety helmet detection," Journal of Sensors, vol. 2021, p. 4746516, 2021.

[17] Guang Han, Mengcheng Zhu, Xuechen Zhao, and Hua Gao, "Method based on the cross-layer attention mechanism and multiscale perception for safety helmet-wearing detection," Computers & Electrical Engineering, vol. 95, p. 107458, 2021.

[18] Renjie Song, and Ziming Wang, "RBFPDet: An anchor-free helmet wearing detection method," Applied Intelligence, vol. 53, no. 5, pp. 5013-5028, 2023.

[19] Jun Liu, Xuhua Xian, Zhenjie Hou, Jiuzhen Liang, and Hao Liu, "Safety helmet wearing correctly detection based on capsule network," Multimedia Tools and Applications, vol. 83, no. 2, pp. 6351-6372, 2024.

[20] Ju-Yeon Lee, Woo-Seok Choi, and Sang-Hyun Choi, "Verification and performance comparison of CNN-based algorithms for two-step helmet-wearing detection," Expert Systems with Applications, vol. 225, p. 120096, 2023.

[21] Xiang, Changcheng, Duofen Yin, Fei Song, Zaixue Yu, Xu Jian, and Huaming Gong, "A fast and robust safety helmet network based on a multiscale Swin transformer," Buildings, vol. 14, no. 3, p. 688, 2024.

[22] Jiayuan Wang, Q. M. Jonathan Wu, and Ning Zhang, "You only look at once for real-time and generic multi-task," IEEE Transactions on Vehicular Technology, vol. 73, no. 9, pp. 12625-12637, 2024.

[23] Shiquan Gao, and Ying Tian, "Research on steel surface defects detection algorithms by YOLOv8 based on attention mechanism," IAENG International Journal of Computer Science, vol. 51, no. 9, pp. 1309-1315, 2024.

[24] Ding-Xuan Zhou, "Theory of deep convolutional neural networks: downsampling," Neural Networks, vol. 124, pp. 319-327, 2020.

[25] Alexandros Stergiou, and Ronald Poppe, "Adapool: exponential adaptive pooling for information-retaining downsampling," IEEE Transactions on Image Processing, vol. 32, pp. 251-266, 2022.

[26] Jin Zhu, Tao Hu, Linhan Zheng, Nan Zhou, Huilin Ge, and Zhichao Hong, "YOLOv8-C2f-Faster-EMA: an improved underwater trash detection model based on YOLOv8," Sensors, vol. 24, no. 8, p. 2483, 2024.

[27] Shi Liu, Meng Zhu, Rui Tao, and Honge Ren, "Fine-grained feature perception for unmanned aerial vehicle target detection algorithm," Drones, vol. 8, no. 5, p. 181, 2024.

[28] Qi Yan, Yajing Zheng, Shanshan Jia, Yichen Zhang, Zhaofei Yu, and Feng Chen, "Revealing fine structures of the retinal receptive field by deep-learning networks," IEEE Transactions on Cybernetics, vol. 52, no. 1, pp. 39-50, 2020.

[29] Hao Ding, Bo Xia, Weilin Liu, Zekai Zhang, Jinglin Zhang, Xing Wang, and Sen Xu, "A novel mamba architecture with a semantic transformer for efficient real-time remote sensing semantic segmentation," Remote Sensing, vol. 16, no. 14, p. 2620, 2024.

[30] Hanwei Zhang, Ying Zhu, Dan Wang, Lijun Zhang, Tianxiang Chen, Ziyang Wang, and Zi Ye, Tongxiang Chen, Ziyu Wang, and Zhen Ye, "A survey on visual mamba," Applied Sciences, vol. 14, no. 13, p. 5683, 2024.

[31] Qinfeng Zhu, Yuanzhi Cai, Yuan Fang, Yihan Yang, Cheng Chen, Lei Fan, and Anh Nguyen, "Samba: semantic segmentation of remotely sensed images with state space model," Heliyon, vol. 10, no. 19, p. e38495, 2024.

[32] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou, "A survey of convolutional neural networks: analysis, applications, and prospects," IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 12, pp. 6999-7019, 2021.

[33] Qiaohong Chen, and Jing Li, "Dual triple attention guided CNN-VMamba for medical image segmentation," Multimedia Systems, vol. 30, no. 5, p. 275, 2024.

[34] Hao Yi, Bo Liu, Bin Zhao, and Enhai Liu, "Small object detection algorithm based on improved YOLOv8 for remote sensing," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 17, pp. 1734-1747, 2024.