Optimized YOLOv Algorithm for Fall Detection in Elderly Care Scenarios

Xuping Tang
Mechanical and Automotive Engineering College, Nanjing Tech University Pujiang Institute, Nanjing, 211200, China
E-mail: txp861108@outlook.com

Keywords: fall detection, real-time object detection algorithm, environmental adaptive fall detection model, multi-scale pooling, attention mechanism

Recievd: September 18, 2025

With the intensification of global population aging, falls among the elderly have become a major public health challenge. Traditional fall detection methods have problems like high false alarm rates, susceptibility to lighting, occlusion restrictions, and easy invasion of privacy. To optimize the detection accuracy and robustness, an optimized real-time object detection algorithm (You Only Look Once version, YOLOv) for elderly care scenarios was proposed. The research develops an environment adaptive fall detection model based on optimized YOLOv algorithm, which intelligently selects CSCD-YOLOv5 or MDF-YOLOv8 algorithm through "environment judgment". The CSCD-YOLOv5 algorithm utilizes the multi-scale pooling of Spatial Pyramid Pooling with Expanded Layer Aggregation Network to enhance feature expression, and combines context guidance to improve detection accuracy in complex environments. The MDF-YOLOv8 algorithm enhances the ability to recover details in low-light conditions through low-light enhancement, lightweight design, and attention mechanism. The experimental results on the Le2i Fall Detection Dataset showed that the area under the recall accuracy curve (AUC) of the research method was approximately 97.6%. At mAP@0.5, the maximum value was 97.4% when the light intensity was 400Lux. The inference speed decreased from 97.5 frames per second to 78.3 frames per second as the input image size increased from 160×160 to 960×960. When the light intensity was 400Lux, the average accuracy reached the maximum of 97.4%. The proposed method has good accuracy, robustness, generalization, and stability, effectively solving the insufficient accuracy and poor robustness of traditional methods, and enhancing the reliability of fall detection in elderly care scenarios.

Povzetek: Predlagana je okolju prilagodljiva metoda zaznavanja padcev starejših z optimiziranim YOLO algoritmom, ki doseže visoko natančnost (do 97,6 %) in dobro robustnost v realnem času.

1 Introduction

Due to global population aging, the risk of falls faced by the elderly due to physical decline has significantly increased [1]. Falling not only poses a direct threat to the life safety of the elderly, but may also lead to long-term health problems such as fractures, soft tissue injuries, and psychological trauma [2]. Monitoring the activity status of elderly people and quickly identifying falling behavior can effectively shorten rescue time, reduce the harm caused by falls, and ensure the life, health, and safety of elderly people [3]. The current fall detection algorithms have problems such as high similarity between fall categories, fixed monitoring angles but variable fall angles, and difficulty in feature extraction [4]. The real-time object detection algorithm series (You Only Look Once version, YOLOv) can divide the input image into multiple grid cells, each of which is used for predicting the bounding boxes and their category probabilities that fall within that region. It can also meet the requirements of fall detection in various complex environments [5]. Therefore, to optimize the accuracy and efficiency in elderly care scenarios, an innovative optimized YOLOv algorithm is proposed. Two optimizations are conducted on the YOLOv algorithm for complex behavioral environments and low-light environments. The former's CSCD-YOLOv5 algorithm combines the CSP Bottleneck with 3 Convolutions and Transformers (CBWT3), the Spatial Pyramid Pooling with Expanded Layer Aggregation Network (SPPELAN), the C3 with Context Guided Block (C3-CGB) and the Detection with Feature Refinement (Detect-FR). The latter's MDF-YOLOv8 algorithm combines C2f with Channel-Spatial attention and Multi-Scale (C2f-CSMS), Details Enhancement Feature (DEF), and C2f-Faster Block modules.

In addition, to enhance the adaptability of the algorithm in complex nonlinear scenarios, this study draws on adaptive control methods for uncertain systems in control theory, such as fractional order chaotic system synchronization and robust neural adaptive control, to improve the robustness of the model to dynamic environmental changes. In the future, research methods can be further combined with real-time safety mechanisms and personalized psychophysiological technologies to extend to intelligent rehabilitation systems for people in the environment, achieving more adaptive health monitoring for the elderly. In the process of building a fall detection model, the research attaches great importance to the privacy protection of the elderly. Three measures are adopted during the data collection phase to ensure privacy and security: (1) All image data are immediately processed for face blurring and identity information desensitization after collection. (2) The original video is not uploaded to the cloud, only detection results and alarm information are output. (3) Data access rights are strictly limited, and only authorized researchers can access the data. The above measures can effectively prevent privacy leakage and improve the credibility and acceptability of the system in elderly care scenarios. It is expected that the method can provide theoretical support for fall detection technology in elderly care scenarios.

2 Related works

With the increasing aging of the global population, health problems are becoming increasingly prominent. Therefore, research on fall detection technology is significant. Nooruddin S et al. built a method for fall detection and rescue system using single sensor and multi-sensor to address the health risks caused by accidental falls in the elderly. By systematically reviewing the architecture, sensor types, and performance indicators of existing technologies, the characteristics of two types of systems were classified and compared. The research method could significantly reduce the injuries and medical costs caused by falls [6]. Gaya-Morey et al. designed a computer vision method using deep learning for fall detection in elderly people living independently. A review analyzed 151 relevant studies from 2019 to 2023. The method had good privacy protection and practical deployment capabilities [7]. Durga Bhavani et al. built a fall classification method based on deep Convolutional Neural Network (CNN) to detect falls in the elderly. The research method had high accuracy and efficiency, and could reduce the health risks and economic burden caused by falls [8]. Qian et al. designed a wearable detection system using a multi-level threshold algorithm to address the poor operability, insufficient medical system interfaces, and high power consumption in fall detection. The research method had high accuracy, efficiency, sensitivity, and specificity [9]. Jain et al. built a pre-impact fall detection system based on CNN to address the insufficient response speed in fall detection. The transition window optimization algorithm was introduced during the research process. The research method had a low false negative rate and strong generalization ability [10].

Many domestic and foreign scholars have explored YOLOv. Xie et al. built a feature enhanced YOLO-based to address the low accuracy and high model complexity. The burden on the model was reduced by depthwise separable convolution and dense connections, and a new loss function and an anchor box optimization strategy were designed. The research method had high efficiency and accuracy [11]. Lian et al. proposed a pork freshness image recognition model based on YOLOv 8n to address the low accuracy and slow speed of pork freshness detection in the cold fresh meat industry chain. The research method had high accuracy and real-time performance [12]. Liu Z proposed an improved lightweight YOLOv8s algorithm to address the missed and false detections caused by scale and lighting changes in drone aerial photography. During the research process, SFPN and SDCN modules, model pruning, and a non-maximum suppression intersection ratio algorithm based on minimum point distance were designed. The results indicated that the research method significantly reduced parameters and computational complexity while effectively improving detection accuracy and robustness in infrared and visible light scenes [13]. Hao et al. proposed an automatic recognition model based on an improved target tracking network to address the low efficiency in manual supervision of motor vehicle traffic safety violations. The method could significantly improve recognition accuracy and stability, effectively assisting traffic management [14]. Zhang et al. built a lightweight MobileOne-YOLO for real-time detection of aircraft cargo hold fires. The MobileOne module was integrated into the YOLOv5 backbone to reduce the parameter count. The research method was efficient and real-time [15].

In recent years, research on video-based fall detection has gradually introduced temporal modeling architectures (such as LSTM, 3D-CNN, Transformer, etc.) to capture temporal features of actions. For example, Jain R et al. [10] combined CNN and LSTM to achieve pre-impact fall detection, but relied on wearable sensors. Butt et al. [2] used LSTM and transfer learning, but were limited by the real-time nature of temporal modeling. In terms of YOLO-based fall detection, some studies have attempted to apply YOLOv4 or YOLOv5 to human pose detection, but many have not considered complex environment adaptation and low-light enhancement (such as Gai R et al. [4] used YOLOv4 for fruit detection, without optimizing for fall scenarios). In contrast, the proposed environment adaptive fall detection model, by dynamically selecting CSCD-YOLOv5 or MDF-YOLOv8 algorithms, not only has stronger robustness in temporal behavior modeling, but also exhibits better detection accuracy and real-time performance in complex scenes such as low-light and occlusion. In summary, existing research has shown good performance in fall detection in elderly care settings, but it still faces challenges such as poor environmental adaptability, privacy breaches, and weak algorithm generalization ability. The YOLOv algorithm can unify target localization and classification tasks into a single neural network, thereby significantly reducing inference time while maintaining high detection accuracy. By fusing multi-scale features, it can optimize its detection ability for small targets in specific scenarios and enhance its detection performance. Therefore, the research proposes an optimized YOLOv algorithm for fall detection, hoping that it can meet the requirements when designing fall detection methods, improving the detection accuracy and generalization ability.

3 Fall detection method in elderly care scenarios based on improved YOLOv algorithm

3.1 Fall detection for elderly people in complex behavioral environments based on CSCD-YOLOv5

With the intensification of global aging, falls among the elderly have become a core health threat. Traditional visual detection methods are prone to misjudgment in complex behavioral environments. In low-light scenes, the missed detection rate may sharply increase due to image blurring [16]. Inspired by the adaptive inversion control method for a class of uncertain nonlinear systems, this study introduces a similar inversion mechanism in feature extraction and multi-scale fusion to enhance the modeling ability of the model for dynamic behavior changes. The CSCD-YOLOv5 complex behavior environment fall detection model can distinguish subtle differences in actions based on the similarity between fall behavior and daily actions through methods such as multi-classifier ensemble [17]. The MDF-YOLOv8 fall detection model in low-light environments can solve the low-light image noise and contrast degradation by introducing domain adaptation and other technologies [18]. Combining the two models can achieve all-weather and all scene coverage detection, enhancing the discriminative power of complex behaviors and the generalization ability of low-light scenes.

In the CSCD-YOLOv5 complex behavior environment fall detection model, the YOLOv5 algorithm uses a bounding box regression loss function to reflect the essential requirement of geometric consistency in object detection, which is the core of multi-scale object detection tasks. Its expression is shown in equation (1).

$$\begin{cases} \mathrm{L}_{\mathrm{CIoU}} = 1 - \mathrm{IoU} + \dfrac{\rho^2(b, b^{\mathrm{gt}})}{c^2} + \alpha v \\ v = \dfrac{4}{\pi^2}\left(\arctan\dfrac{w^{\mathrm{gt}}}{h^{\mathrm{gt}}} - \arctan\dfrac{w}{h}\right)^2 \end{cases} \quad (1)$$

In equation (1), $\mathrm{L}_{\mathrm{CIoU}}$ signifies the bounding box regression loss function. IoU signifies the intersection ratio. ρ signifies the Euclidean distance between the predicted box center point b and the real box center point b^{gt} . c signifies the diagonal length of the smallest bounding rectangle. α signifies the weight coefficient. v signifies the penalty for aspect ratio differences. w and h signify the width and height of the prediction box. w^{gt} and h^{gt} signify the width and height of the real box. However, the detection performance of YOLOv5 algorithm in complex behavioral environments in elderly care scenarios may decrease. Therefore, the study introduces CBWT3 module, SPPELAN module, C3-CGB module, and Detect-FR detection head to optimize YOLOv5. Figure 1 illustrates the overall structure of CSCD-YOLOv5.

In Figure 1, the CSCD-YOLOv5 algorithm introduces optimization modules such as SPPELAN and CBWT3 based on the YOLOv5 algorithm, and achieves efficient multi-scale detection through the collaborative architecture of Backbone-Neck-Head. Backbone focuses on feature extraction, Neck enhances feature diversity using C3-CGB, and Head improves small object detection accuracy through multi-scale prediction. The study dynamically fuses local and global features through adaptive pooling and gating mechanisms, as expressed in equation (2).

$$\begin{cases} \mathrm{F}_{\mathrm{CNN}} = \mathrm{Conv}_{3\times3}(X) \oplus \mathrm{DWConv}_{5\times5}(X) \\ F_{\mathrm{Trans}} = \mathrm{LayerNorm}\left(\mathrm{Softmax}\left(\dfrac{(XW_Q)(XW_K)^T}{\sqrt{d_k}}\right)(XW_V)\right) \\ F_{\mathrm{CBWT}} = \mathrm{AdaPool}(F_{\mathrm{CNN}}) \otimes \sigma(F_{\mathrm{Trans}}) \end{cases} \quad (2)$$

In equation (2), F signifies the branch feature map. X signifies the input feature. $\mathrm{Conv}_{3\times3}$ is the standard 3×3 convolution. DWConv is a depthwise separable convolution. $\oplus$ is element wise addition. W_Q , W_K , and W_V are projectable learning matrices. $\mathrm{Softmax}(\cdot)$ is the Softmax normalization in the row direction. $\mathrm{LayerNorm}(\cdot)$ is layer normalization. $\sqrt{d_k}$ is the scaling factor. $\mathrm{AdaPool}$ is adaptive pooling. σ signifies the Sigmoid activation function. $\otimes$ refers to element wise multiplication. Subsequently, this study introduces a Multi-Head Attention Mechanism (MHAM) to model long-range spatial correlations, as expressed in equation (3).

$$\begin{cases} \mathrm{head}_i = \mathrm{Softmax}\left(\dfrac{(XW_Q^i)(XW_K^i)^T}{\sqrt{d_k}}\right)(XW_V^i) \\ F_{\mathrm{MHSA}} = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h)W_O + X \end{cases} \quad (3)$$

In equation (3), Concat is the multi-head output concatenated along the channel dimension. W_O signifies the output projection matrix. h signifies the number of attention heads. Figure 2 presents the structure of the MHAM.

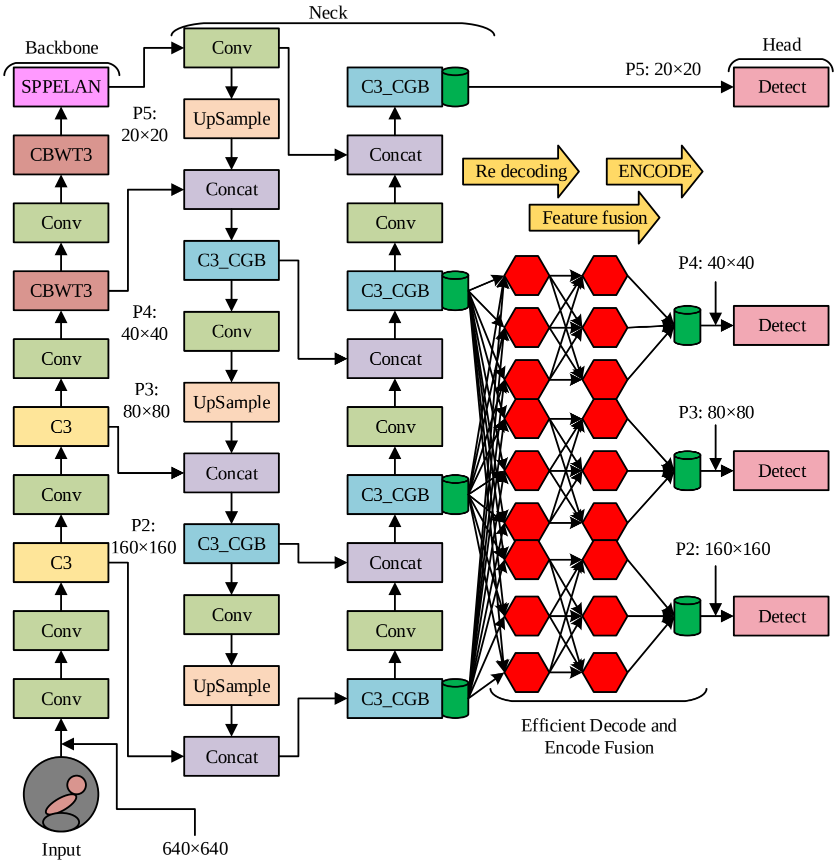


Figure 1: CSCD-YOLOv5 algorithm overall structure diagram

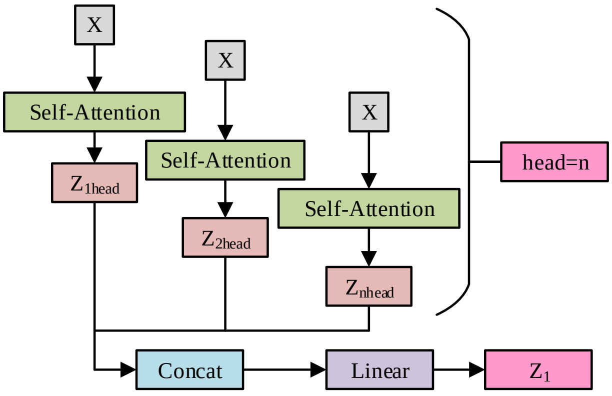


Figure 2: Structure diagram of MHAM

From Figure 2, the MHAM captures multi-dimensional features of input through parallel independent self-attention heads, and integrates heterogeneous information through concatenation and linear transformation to enhance the modeling ability for complex relationships. It is the core component of Transformer architecture. The SPPELAN mainly extracts multi-granularity contextual information through its SPP layer, solving the variable target scales in fall detection. Its expression is shown in equation (4).

$$\begin{cases} P_k = \mathrm{MaxPool}_{k\times k}(F_{\text{in}})(k\in\{5,9,13\}) \\ F_{\text{SPP}} = \mathrm{Concat}(F_{\text{in}}, P_5, P_9, P_{13}) \\ F_{\text{ELAN}} = \mathrm{Conv}_{1\times1}(F_{\text{SPP}}) \oplus \sigma(\mathrm{Conv}_{3\times3}(F_{\text{SPP}})) \\ F_{\text{SPPELAN}} = \mathrm{Conv}_{3\times3}(F_{\text{ELAN}}) \end{cases} \tag{4}$$

In equation (4), P_k is the feature of $k\times k$ after maximizing the pooling $\mathrm{MaxPool}$ · F_{in} is the input feature map. The importance of each channel feature is dynamically adjusted by the C3-CGB to optimize the robustness to complex environments, as expressed in equation (5).

$$\begin{cases} G_{\text{context}} = \mathrm{AvgPool}_G(F_{\text{in}}) \\ M_{\text{guide}} = \mathrm{MLP}(G_{\text{context}}) \\ F_{\text{CGB}} = \mathrm{Conv}_{3\times3}(F_{\text{in}} \otimes M_{\text{guide}}) \\ F_{\text{C3}} = \mathrm{Concat}(F_{\text{in}}, F_{\text{CGB}})W_{\text{fuse}} \end{cases} \tag{5}$$

In equation (5), G_{context} is the contextual feature. M_{guide} is the channel attention vector. $\mathrm{MLP}(\cdot)$ is the output of the multi-layer perceptron. W_{fuse} is 1×1 convolution fusion. The Detect-FR detection head is taken to decouple classification and regression tasks, as expressed in equation (6).

$$\begin{cases} F_{\text{reg}} = DW\text{Conv}(F_{\text{in}}) \\ F_{\text{cls}} = \text{MHSA}(F_{\text{in}}) \\ F_{\text{fuse}} = \text{Conv}_{1\times 1}(\text{Concat}(F_{\text{reg}}, F_{\text{cls}})) \\ F_{\text{recoded}} = \text{LayerNorm}(\text{Linear}(F_{\text{fuse}})) \end{cases} \quad (6)$$

In equation (6), F_{reg} signifies the classification feature map. F_{cls} signifies the classification feature map. F_{fuse} signifies the fused feature map. F_{recoded} is the output after re-encoding. MHSA is the multi-head attention operator. Linear is the fully connected layer. Finally, the composite loss of all module improvements is integrated, and its composite loss function expression is shown in equation (7).

$$L_{\text{total}} = \lambda_1 L_{\text{cls}} + \lambda_2 L_{\text{reg}} + \lambda_3 L_{\text{obj}} \quad (7)$$

In equation (7), L_{cls}, L_{reg}, L_{obj}, and L_{total} represent classification loss, regression loss, objective loss, and total loss, respectively. λ is a hyperparameter. In summary, the CSCD-YOLOv5 for elderly people fall detection in different behavioral environments is shown in Figure 3.

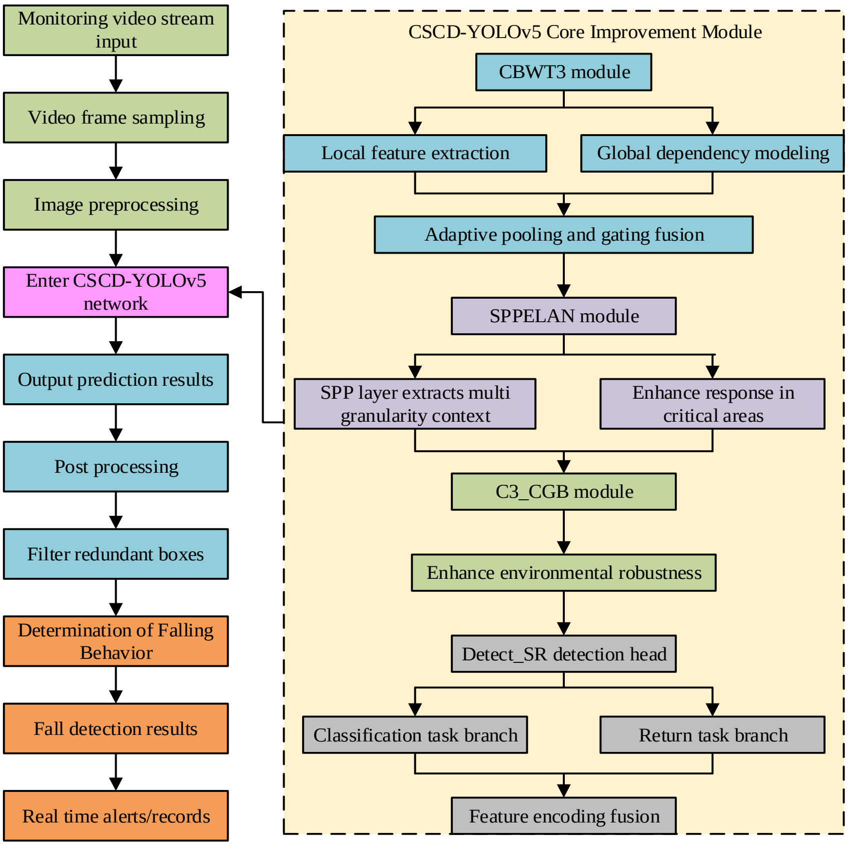

Monitoring video stream input
Video frame sampling
Image preprocessing
Enter CSCD-YOLOv5 network
Output prediction results
Post processing
Filter redundant boxes
Determination of Falling Behavior
Fall detection results
Real time alerts/records
CSCD-YOLOv5 Core Improvement Module
CBWT3 module
Local feature extraction
Global dependency modeling
Adaptive pooling and gating fusion
SPPELAN module
SPP layer extracts multi granularity context
Enhance response in critical areas
C3_CGB module
Enhance environmental robustness
Detect_SR detection head
Classification task branch
Return task branch
Feature encoding fusion

Figure 3: CSCD-YOLOv5 algorithm for elderly fall detection in different behavioral environments

Table 3.1.1: Detailed network structure of CSCD-YOLOv5

Module	Layer type	Input channel	Output channel	Kernel size/step size	Activation function	Parameter quantity (ten thousand)
Backbone	Conv	3	64	6*6/2	SiLU	1.2
	CBWT3	64	128	3*3/1	GELU	4.5
	SPPELAN	128	256	5*5/1	SiLU	7.8
Neck	C3-CGB	256	512	3*3/1	SiLU	12.3
Head	Detect-FR	512	6*(5+cls)	3*1/1	Linear	5.6

Table 3.1.2: Internal structure and hyperparameters of CSCD-YOLOv5 module

Module name	Layer sequence (sequential)	Key hyperparameters
CBWT3	Conv3x3 → BN → GELU → Transformer block (multi-head self-attention) → Residual join	The convolution kernel =3×3, the number of channels =256, the number of heads H=8, and the number of layers =1
SPPELAN	Input features → Parallel MaxPool (kernel size =5,9,13) → Feature concatenation → Conv1×1 → Output	Pooling kernel =[5,9,13], step size =1, number of channels =512
C3-CGB	Input features → Convolution group (Conv3×3, BN, SiLU) → Context-guided block (global average pooling + MLP) → Feature weighting	The number of convolution groups =3, the MLP hidden layer =128, and the activation function =SiLU
Detect-FR	Input Features → Shared convolution → Branch 1 (Classification: Conv1×1 + Softmax) → Branch 2 (Regression: Conv1×1 + Linear)	Convolution kernels =1×1, output channels = number of categories +4

From Figure 3, the CSCD-YOLOv5 algorithm constructs a complete process of "video input → preprocessing → detection → post-processing → alarm" for the elderly fall detection task. It introduces the CBWT3 module, SPPELAN module, and C3_CGB module, combined with the dual task branch of Detect-FR detection head, to improve detection accuracy in complex environments. The CSCD-YOLOv5 network architecture is shown in Table 3.1.1.

Table 3.1.1 describes the backbone, neck, and head structures of the CSCD-YOLOv5 model, including the type of each layer, input/output channels, kernel size, stride, activation function, and number of parameters, where CLS is the number of categories with a value of 1. Moreover, the core module architecture of CSCD-YOLOv5 algorithm is shown in Table 3.1.2.

Table 3.1.2 provides a detailed list of the hierarchical structure and hyperparameter configuration of CBWT3, SPPELAN, C3-CGB, and Detect-FR modules in the core module architecture of CSCD-YOLOv5 algorithm.

3.2 Fall detection method for elderly people in low-light environment based on MDF-YOLOv8

Drawing on the nonlinear optimal control concept driven by induction motors, a lightweight optimal feature selection strategy is introduced in the low-light enhancement module to maximize detail recovery while ensuring speed. The elderly fall detection in complex behavioral environments based on CSCD-YOLOv5 can achieve high-precision real-time recognition of elderly fall behavior in complex scenes, improving the detection accuracy in complex scenes. However, in elderly care settings, there are defects such as ineffective feature extraction, increased noise interference, and dependence on auxiliary lighting under low-light conditions [19]. The YOLOv8 algorithm can solve the core pain points of nighttime detection through multi-spectral fusion and low-light adaptive mechanism [20]. Therefore, to optimize the fall detection performance of the research method under low-light conditions in elderly care scenarios, the YOLOv8 algorithm is optimized. The YOLOv8 algorithm adopts the Anchor-Free mechanism, as presented in equation (8).

$$\begin{cases} \hat{x} = (2\cdot\sigma(\Delta x) - 0.5) + c_x \\ \hat{y} = (2\cdot\sigma(\Delta y) - 0.5) + c_y \\ \hat{w} = w_{\text{anchor}} \cdot \left(2\cdot\sigma(\Delta w)\right)^2 \\ \hat{h} = h_{\text{anchor}} \cdot \left(2\cdot\sigma(\Delta h)\right)^2 \end{cases} \quad (8)$$

In equation (8), $(\hat{x}, \hat{y})$ is the center coordinate of the decoded bounding box. Δx and Δy are the center point offsets predicted by the model. (c_x, c_y) is the coordinate of the upper left corner of the current grid. $\hat{w}$ and $\hat{h}$ signify the width and height of the decoded bounding boxes. Δw and Δh signify the width and height offsets predicted by the model. However, to solve the difficulty in distinguishing the contours and details of the elderly in low-light environments, as well as much noise masking the feature information of the elderly in low-light environments, the research introduces C2f-MSCS module, DEF module, and C2f-Faster Block to optimize the YOLOv8. Figure 4 presents the overall structure of the MDF-YOLOv8.

In Figure 4, MDF-YOLOv8 adopts a Backbone-Neck-Head three-stage architecture, and Backbone achieves efficient feature extraction through C2f-MSCS, DEF, and SPPF. Neck uses C2f-Faster, Upsample, and Concat for multi-scale feature fusion, and enhances representation ability using DEF. The Head adopts a three-detection head structure, combined with CBS and Detect modules to achieve multi-scale object detection. The study introduces the C2f-MSCS module into the backbone network of YOLOv8, highlighting details that are easily overlooked under low-light conditions. The expression is shown in equation (9).

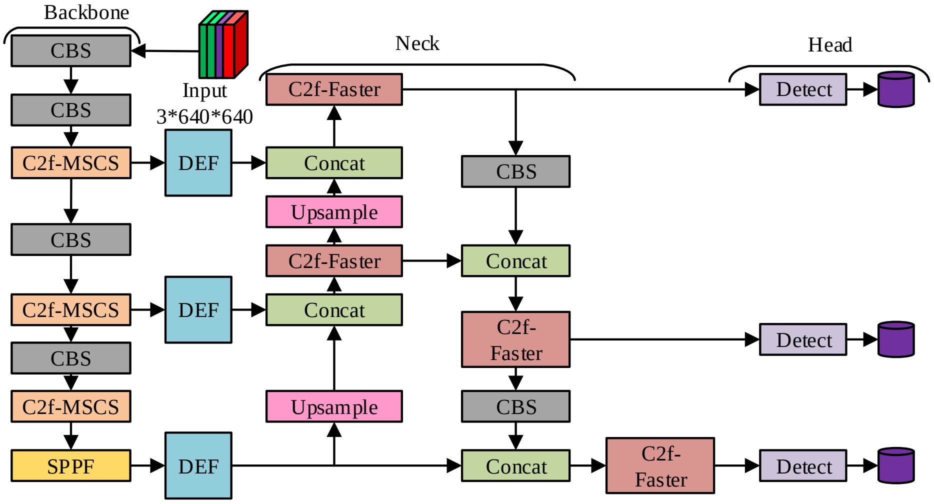


Figure 4: MDF-YOLOv8 detection algorithm model structure diagram

$$\begin{cases} F_{\text{ms}} = Concat(DWConv_{3\times3}(F_{in}), DWConv_{5\times5}(F_{in}), MaxPool(F_{in})) \\ F_{\text{ca}} = \sigma\left(\text{MLP}\left(\text{AvgPool}(F_{\text{ms}})\right) + \text{MLP}\left(\text{MaxPool}(F_{\text{ms}})\right)\right) \otimes F_{\text{ms}} \\ F_{\text{sa}} = \sigma\left(\text{Conv}_{7\times7}\left(\text{Concat}\left(\text{AvgPool}(F_{\text{ca}}), \text{MaxPool}(F_{\text{ca}})\right)\right)\right) \otimes F_{\text{ca}} \end{cases} \quad (9)$$

In equation (9), F_{ms} is a multi-scale feature. F_{ca} is the channel weighted feature. F_{sa} is a spatially weighted feature. Next, the DEF module is introduced into the YOLOv8 network to address the uneven image lighting distribution under low-light conditions, as expressed in equation (10).

$$\begin{cases} F_{\text{expand}} = \text{ReLU}\left(\text{Conv}_{1\times1}(F_{\text{in}})\right) \\ F_{\text{detail}} = \text{Conv}_{3\times3}\left(\text{GroupNorm}(F_{\text{expand}})\right) \oplus F_{\text{in}} \\ F_{\text{DEF}} = \text{Conv}_{1\times1}\left(\text{ELU}(F_{\text{detail}})\right) \end{cases} \quad (10)$$

In equation (10), F_{expand} is the feature map after channel expansion. ReLU is the modified linear unit activation function. F_{detail} is the intermediate feature map for detail enhancement. GroupNorm is group normalization. ELU is the exponential linear unit activation function. Subsequently, the research replaces the C2f module with the C2f-Faster Block, effectively reducing computational complexity, as expressed in equation (11).

$$F_{\text{PConv}} = \text{Conv}_{k\times k}\left(F_{\text{in}} \otimes M\right) \oplus \left(\text{Conv}_{1\times1}(F_{\text{in}}) \otimes (1-M)\right) \quad (1$$
1)

In equation (11), M is the channel selection mask. The C2f-Faster Block module, which combines PConv's sparse convolution with MLP's global modeling, can achieve lightweight feature extraction, as expressed in equation (12).

$$\begin{cases} F_{\text{part}} = \text{PConv}_{3\times3}(F_{\text{in}}) \\ F_{\text{mlp}} = \text{MLP}\left(\text{GELU}(F_{\text{part}})\right) \\ F_{\text{Faster}} = \text{Conv}_{1\times1}\left(\text{LayerNorm}(F_{\text{mlp}} \oplus F_{\text{in}})\right) \end{cases} \quad (12)$$

In equation (12), F_{part} signifies the feature map output by partial convolution. GELU signifies the Gaussian error linear unit activation function. To suppress low-light noise, different scales of semantic information output by the C2f-MSCS module are fused, and its expression is shown in equation (13).

$$\begin{cases} F_{\text{pyramid}} = \text{Concat}\left(F_{\text{MSCS}}, \text{Upsample}(F_{\text{MSCS}} \downarrow_2), \text{Upsample}(F_{\text{MSCS}} \downarrow_4)\right) \\ F_{\text{refined}} = \text{CBAM}\left(\text{Conv}_{3\times3}(F_{\text{pyramid}})\right) \end{cases} \quad (13)$$

In equation (13), F_{pyramid} is the multi-scale pyramid feature map. Upsample is bilinear interpolation upsampling. $\downarrow_s$ is downsampled s times. F_{refined} is the refined feature map. CBAM is the module that concatenates channel and spatial attention. To solve the boundary blurring under low-light conditions, an edge consistency loss is added to the regression loss, as shown in equation (14).

$$L_{\text{low-light}} = \gamma_1 \cdot \text{CIoU}(b_{\text{pred}}, b_{\text{gt}}) + \gamma_2 \cdot \| \varphi(F_{DEF}) - \varphi(F_{\text{gt}}) \|_2 \quad (14)$$

In equation (14), $L_{\text{low-light}}$ is the total loss of low-light adaptive. γ is the weight coefficient.

$\text{CIoU}(b_{\text{pred}}, b_{\text{gt}})$ is the localization regression loss. $\varphi(\cdot)$ is the edge feature extraction operator. F_{gt} is the grayscale feature of the original image within the real frame area. Finally, the study defines the data flow between optimization module, which is expressed as equation (15).

$$F_{\text{out}} = \text{C2f\_FasterBlock}\left(\text{DEF}\left(\text{C2f\_MSCS}(F_{\text{in}})\right) \oplus F_{\text{in}}\right) \quad (15)$$

In equation (15), F_{out} is the output feature map. $\oplus F_{\text{in}}$ is the residual connection. In summary, the fall detection process for elderly people in low-light environments based on MDF-YOLOv8 is shown in Figure 5.

From Figure 5, the fall detection model for elderly people in low-light environments based on MDF-YOLOv8 first preprocesses the low-light original image, and outputs the results through multi-scale feature processing using the MDF-YOLOv8 model. Post-processing determines whether a human body has been detected. If detected, human body region features are extracted for fall detection. If a fall is detected, an alarm is triggered and relevant personnel are notified. Otherwise, the image is re-entered. The MDF-YOLOv8 network architecture is shown in Table 3.2.1.

Table 3.2.1 describes the backbone, neck, and head structures of the MDF-YOLOv8 model, where CLS is the number of categories with a value of 1. Moreover, the core module architecture of CSCD-YOLOv5 algorithm is shown in Table 3.2.2.

Table 3.2.2 provides a detailed list of the hierarchical structure and hyperparameter configuration of the C2f-CSMS, DEF, and C2f-Faster Block in the module architecture of the MDF-YOLOv8 algorithm. To improve the usability of images under low-light conditions, contrast limited adaptive histogram equalization is used for image enhancement before inputting the MDF-YOLOv8 model. The specific parameter settings are: clipLimit=2.0 and tileGridSize=(8,8). This preprocessing step is uniformly applied during both training and inference processes to maintain data distribution consistency. The preprocessing and detection model adopts a cascaded pipeline, which first enhances the original image and then inputs it into the network for feature extraction and fall recognition, without using end-to-end joint training. In summary, the specific process of the fall detection method in elderly care scenarios based on optimized YOLOv algorithm is shown in Figure 6.

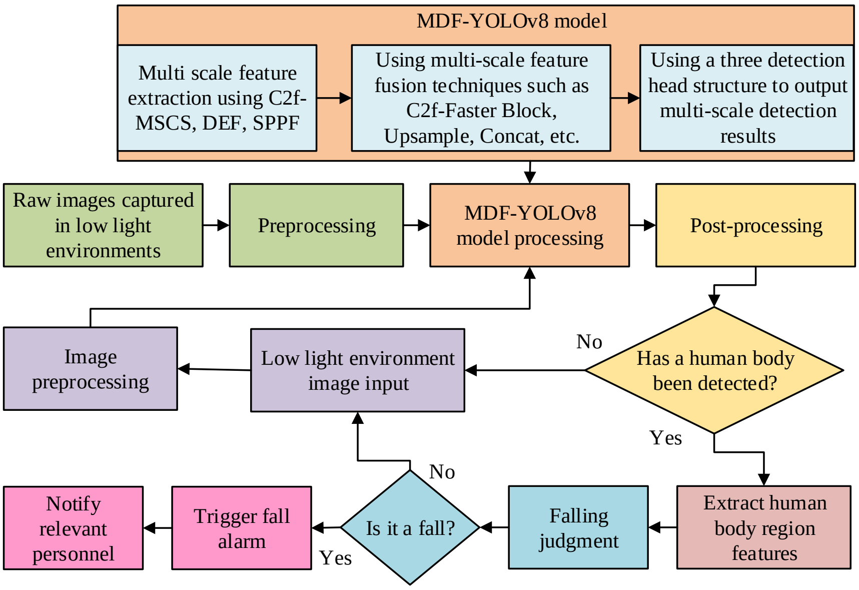

MDF-YOLOv8 model
Multi scale feature extraction using C2f-MSCS, DEF, SPPF
Using multi-scale feature fusion techniques such as C2f-Faster Block, Upsample, Concat, etc.
Using a three detection head structure to output multi-scale detection results
Raw images captured in low light environments
Preprocessing
MDF-YOLOv8 model processing
Post-processing
Image preprocessing
Low light environment image input
No
Has a human body been detected?
Yes
No
Notify relevant personnel
Trigger fall alarm
Is it a fall?
Yes
Falling judgment
Extract human body region features

Figure 5: Fall detection in low-light environments based on MDF-YOLOv8

Table 3.2.1: Internal structure and hyperparameters of CSCD-YOLOv5 module

Module name	Layer sequence (sequential)	Key hyperparameters
CBWT3	Conv3x3 → BN → GELU → Transformer block (multi-head self-attention) → Residual join	The convolution kernel =3×3, the number of channels =256, the number of heads H=8, and the number of layers =1
SPPELAN	Input features → Parallel MaxPool (kernel size =5,9,13) → Feature concatenation → Conv1×1 → Output	Pooling kernel =[5,9,13], step size =1, number of channels =512
C3-CGB	Input features → Convolution group (Conv3×3, BN,	The number of convolution groups =3, the

	SiLU) → Context-guided block (global average pooling + MLP) → Feature weighting	MLP hidden layer =128, and the activation function =SiLU
Detect-FR	Input Features → Shared convolution → Branch 1 (Classification: Conv1×1 + Softmax) → Branch 2 (Regression: Conv1×1 + Linear)	Convolution kernels =1×1, output channels = number of categories +4

Table 3.2.2: Internal structure and hyperparameters of MDF-YOLOv8 module

Module name	Layer sequence (sequential)	Key hyperparameters
C2f-CSMS	Input features → C2f base block → Channel attention (SE module) → Spatial attention (Conv7×7 + Sigmoid) → Feature fusion	The number of attention heads =4, the convolution kernels =7×7, and the number of channels =256
DEF	Input Features → Conv1×1 → ReLU → Group Normalization → ELU → Output	The number of groups =8, the activation function =ELU, and the channel expansion rate =2
C2f-Faster	Input features → PConv (Partial convolution) → GELU → MLP (Fully connected Layer) → Residual connection	The PConv selection ratio is 0.5, the MLP hidden layer is 512, and the activation function is GELU

From Figure 6, an environment adaptive fall detection model based on optimized YOLOv algorithm intelligently selects CSCD-YOLOv5 or MDF-YOLOv8 algorithm for detection through initial "environment judgment". CSCD-YOLOv5 utilizes gate-controlled fusion of CBT3 module and multi-scale pooling of SPPELLAN to enhance feature expression, combined with C3-CGB and Detect-SR to improve detection accuracy in complex environments. MDF-YOLOv8 optimizes its detail recovery ability under low-light conditions through DEF, PConv, and attention mechanisms. Both paths achieve high-precision fall recognition through bounding box optimization, post-processing, and behavior judgment, and ultimately link with an alarm system.

Furthermore, to enhance the system's warning capability before falls occur, a temporal behavior prediction mechanism based on output feedback is introduced. This mechanism models the sequence of human postures in consecutive video frames, predicts future behavioral trends in several frames, and identifies dangerous actions before a fall occurs. The specific steps are as follows: First, CSCD-YOLOv5 or MDF-YOLOv8 are utilized to extract the coordinates of human keypoints (such as head, torso, and limbs) for each frame. Subsequently, a temporal sequence is constructed. Then, lightweight LSTM or Transformer encoders are introduced to encode the pose sequence and output pose predictions for future frames. Finally, the stability index between the current posture and the predicted posture is calculated, and an alert is triggered if it exceeds the threshold. This warning mechanism can be embedded into existing detection models to form a "detection prediction feedback" loop. Although this study is based on single frame image for fall detection and does not explicitly introduce a time modeling module, in practical deployment, a short-term time filtering strategy is introduced through the post-processing stage to smooth the detection results and reduce instantaneous misjudgments. In the future, time modeling methods such as optical flow or 3D convolution will be considered to further enhance the discriminative ability for continuous actions.

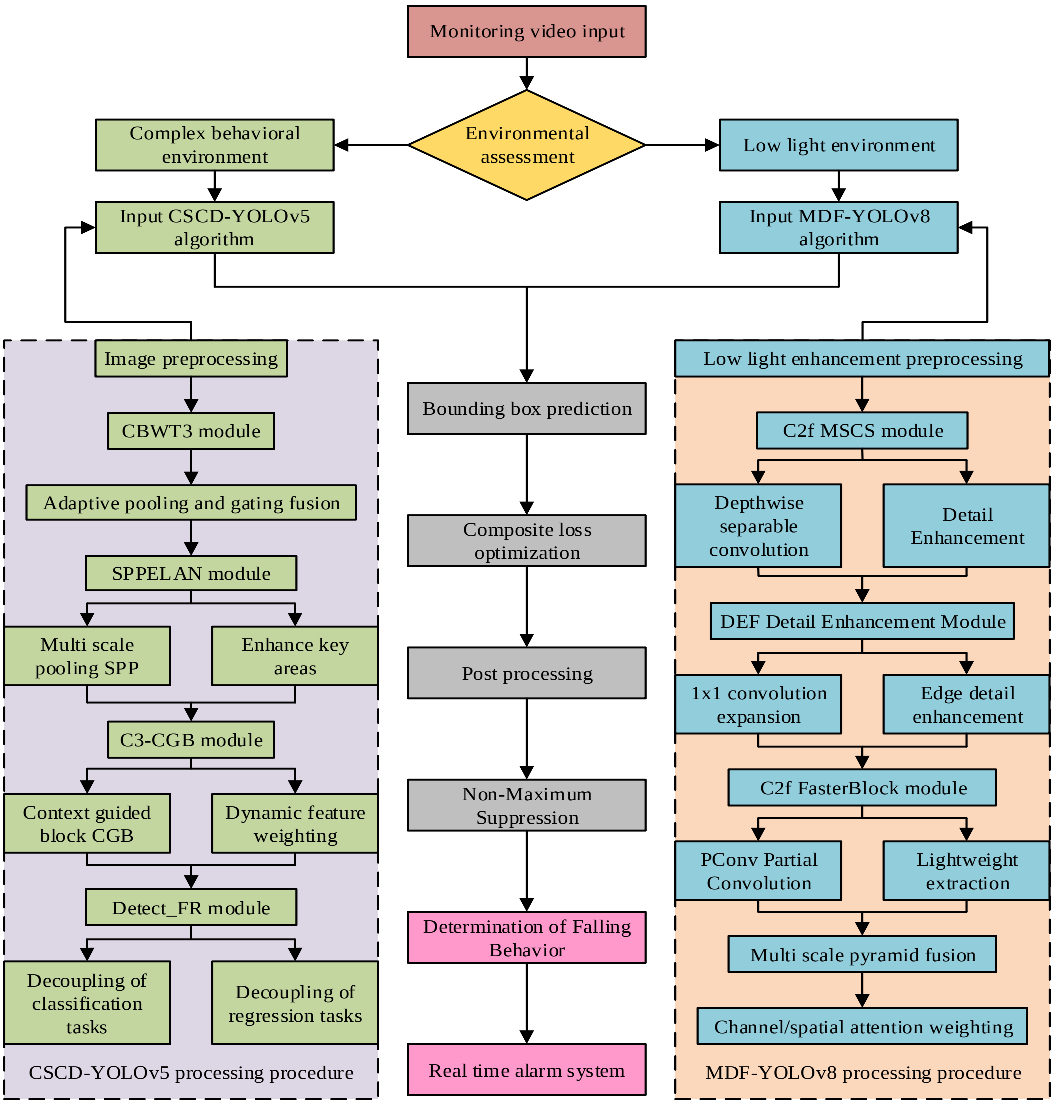

Monitoring video input
Environmental assessment
Complex behavioral environment
Low light environment
Input CSCD-YOLOv5 algorithm
Input MDF-YOLOv8 algorithm
Image preprocessing
CBWT3 module
Adaptive pooling and gating fusion
SPPELAN module
Multi scale pooling SPP
Enhance key areas
C3-CGB module
Context guided block CGB
Dynamic feature weighting
Detect_FR module
Decoupling of classification tasks
Decoupling of regression tasks
CSCD-YOLOv5 processing procedure
Bounding box prediction
Composite loss optimization
Post processing
Non-Maximum Suppression
Determination of Falling Behavior
Real time alarm system
Low light enhancement preprocessing
C2f MSCS module
Depthwise separable convolution
Detail Enhancement
DEF Detail Enhancement Module
1x1 convolution expansion
Edge detail enhancement
C2f FasterBlock module
PConv Partial Convolution
Lightweight extraction
Multi scale pyramid fusion
Channel/spatial attention weighting
MDF-YOLOv8 processing procedure

Figure 6: Fall detection in elderly care scenarios based on optimized YOLOv algorithm

Overall, the proposed environment adaptive switching strategy shares similarities in ideology with adaptive control methods in nonlinear system theory. The latter adjusts the controller parameters in real-time to cope with dynamic changes in the system, while this study dynamically selects CSCD-YOLOv5 or MDF-YOLOv8 algorithms through the "environment judgment" module to achieve autonomous adaptation to different lighting and behavioral scenes, thus achieving similar control logic in the visual detection system and improving the system's environmental robustness.

4 Validation of fall detection method in elderly care scenarios based on optimized YOLOv algorithm

4.1 Performance testing of fall detection methods in elderly care scenarios

To verify the fall detection method in elderly care scenarios based on optimized YOLOv algorithm, a simulation model is built. Table 4.1.1 presents the experimental environment. As shown in Table 4.1.1, the specific configurations in the table were used for performance testing. To ensure model reproducibility, the hyperparameter configuration used in the training process is AdamW optimizer. The initial learning rate is 0.001. The learning rate plan is CosineAnnealingLR, and Tmax=100. The batch size is 16. The number of iterations is 100 epochs. The weight decay is 0.05. The data augmentation methods include random flipping, rotation (± 15°), and brightness contrast adjustment. If the validation set loss does not decrease for 10 consecutive epochs, the early stop time will be terminated. The random seed is 42. The research method is compared with Computer Vision-based Fall Detection (CV-FD) and Wearable Sensor-based Fall Detection (WS-FD). Among them, CV-FD adopts background subtraction and morphological feature analysis, and the implementation method is derived from reference [3]. WS-FD is based on three-axis accelerometer data and uses threshold to determine falls. The implementation method is based on reference [9]. All methods are trained and tested on the same Le2i Fall Detection Dataset dataset, with hyperparameters following the recommended settings in the original paper to ensure the fairness of the comparison. The study uses the Le2i Fall Detection Dataset, which consists of 2,800 video clips covering various types of falls and different lighting conditions. The dataset is randomly divided into training set, validation set, and testing set in a ratio of 7:2:1 [21]. All comparison methods are trained and tested on the same dataset partition to ensure fair comparison. The Le2i Fall Detection Dataset is a publicly available benchmark dataset for fall detection, which includes video sequences of falls and daily behavior in various indoor scenarios. All videos in the dataset are collected at a frame rate of 25 frames per second, with a resolution of 320 × 240 pixels. Frame by frame bounding box annotation is provided, and the annotation protocol clearly distinguishes between "falling" and "non falling" behaviors.

In addition, the dataset provides temporal contextual information of consecutive frames, which is suitable for fall behavior analysis based on video sequences. This dataset contains video clips of various daily activities and fall behaviors, covering different environments, lighting, and perspectives. A subset of categories including "walking", "sitting and lying", "forward leaning falls", "backward leaning falls", "lateral falls", and "compound posture falls" is selected, and the human bounding boxes and behavior categories in each frame of the image are manually annotated. The labeling follows the PASCAL VOC format, and the criteria for determining falling behavior are that the angle between the human torso and the ground is less than 45° or the height of the human center of gravity drops sharply. The precision-recall curve and the area under the curves of the three methods are compared. In the study, the average accuracy is calculated using an IoU threshold of 0.5. For each category, the precision and recall rates at different confidence thresholds are calculated, the precision-recall curve is plotted. The 11-point interpolation is taken to calculate the area under the curve as the average precision for that category. The final mean Average Precision (mAP) is obtained by taking the average of AP values for all categories. The final result is shown in Figure 7.

Table 4.1.1: Test environment and specific configuration

Test environment	Specific configuration
GPU	NVIDIA RTX 4090 (24GB video memory)
CPU	Intel i9-13900K
Edge deployment device	Jetson AGX Orin
Auxiliary equipment	NtelCare 4D radar
Deep learning framework	PyTorch 1.13 + CUDA 11.7
Operating system	Ubuntu 20.04 LTS

From Figure 7 (a), the research method exhibited a high recall-precision trend overall. When the recall was 98.5%, its precision was still 93.8%, and the area under the recall-precision curve was close to a rectangle, with an area value of about 97.6%. As shown in Figure 7 (b), the recall-precision curve of the CV-FD method showed a significant downward trend. When the recall was 92.3%, the precision was 50.6%, and the area was also close to a rectangle, with an area value of about 77.4%. From Figure 7 (c), the recall-precision curve of the WS-FD method continued to steadily decline. When the recall was 83.5%, its precision was only 18.9%, and the area was close to a triangle, with an area value of about 32.9%. Overall, compared to comparative methods, the research method has better accuracy, stability, and robustness. In addition, the study provided confusion matrices and Average Precision per class (AP) to further evaluate model performance. The confusion matrix showed that on the test set, the True Positive Rate (TP) was 95.2%, the False Positive Rate (FP) was 2.1%, the True Negative Rate (TN) was 97.8%, and the False Negative Rate (FN) was 4.8%. The AP values for each category were as follows: the forward leaning fall AP was 96.5%, the backward leaning fall AP was 97.2%, the lateral fall AP was 94.8%, and the compound posture AP was 92.1%. These results indicate that the model has high accuracy on all types of falls. The inference speed of the three methods under different input image sizes and the correct detection rate under different occlusion areas are compared, as illustrated in Figure 8.

From Figure 8 (a), the safety inference speed threshold of the system at each input image size was 50 frames per second, and the inference speed of each method decreased with the increase of input image size. The inference speed of the research method decreased minimally with the increase of input image size, and was generally above the safety threshold. When the input image size increased from 160*160 to 960*960, the inference speed of the research method decreased from 97.5 frames/s to 78.3 frames/s, with a decrease of only 19.2 frames/s. However, the other two methods showed a significantly larger decrease than the research method, among which the inference speed of the CV-FD method showed an abnormal decrease when the input image size was 800*800. From Figure 8 (b), the correct detection rate of different methods demonstrated a decreasing trend with the increase of human occlusion area. The correct detection rate of the research method was 99.6% when the human occlusion area was 20%, 97.8% when the occlusion area was 40%, 95.2% when the occlusion area was 60%, and 93.4% when the occlusion area was 80%. The correct detection rates of the other two methods were significantly lower than those of the research method in terms of the occlusion area of each human body. To further validate the significant performance advantages of the research method (optimized YOLOv) compared to baseline methods (CV-FD and WS-FD), the paired t-test (α=0.05) was conducted. In multiple key indicators such as mAP, inference speed, missed detection rate, and false positive rate, the research method significantly outperformed the two baseline methods (p<0.01). In addition, the bootstrap method calculated a 95% confidence interval, and the results showed that the performance gain interval of the research method did not include zero, further confirming its statistical significance.

Overall, compared to comparison methods, the research method has better computational efficiency, detection accuracy, robustness, and generalization. Overall, the fall detection method for elderly care scenarios based on optimized YOLOv algorithm has high detection accuracy, stability, robustness, computational efficiency, and generalization ability. In addition, to evaluate the actual deployment performance of the research method on edge devices, end-to-end latency was tested on the Jetson AGX Orin platform. The experimental results showed that the research method had an average end-to-end latency of 18.3 milliseconds when the input image size was 640 × 640, which meets the real-time detection requirements. Meanwhile, the average power consumption during the inference process was 14.2 watts, demonstrating a good energy efficiency ratio.

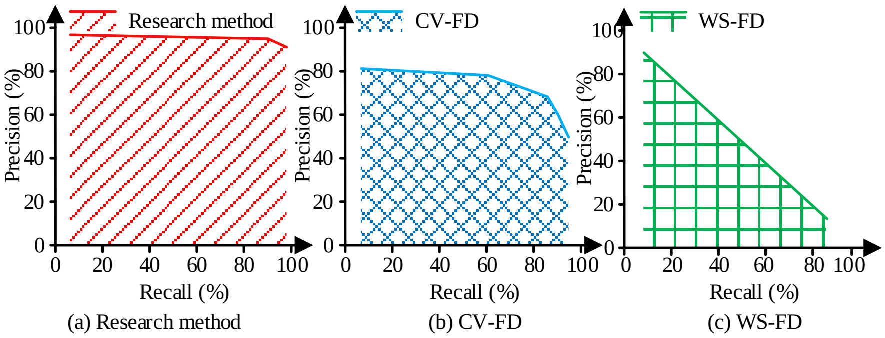


Figure 7: Recall-precision curve and analysis of area under the curve

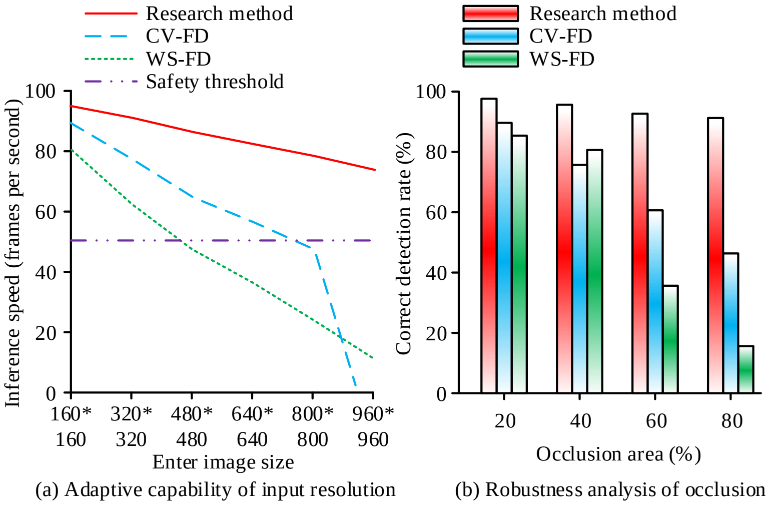
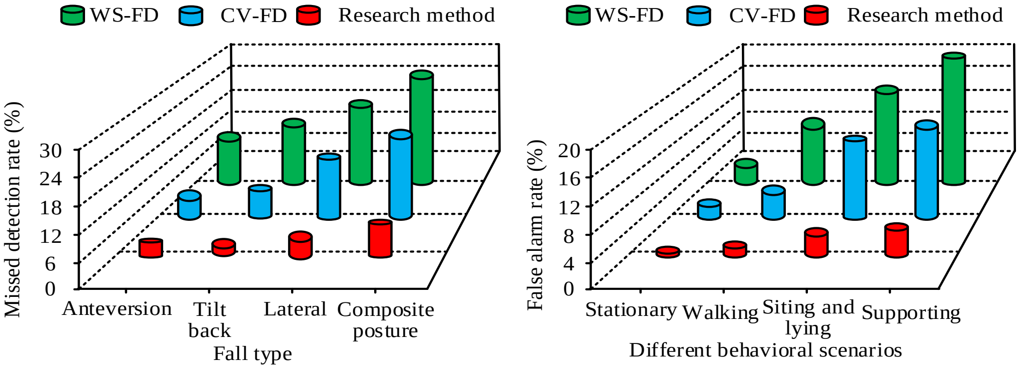


Figure 8: Analysis of adaptive ability and occlusion robustness of input resolution

4.2 The practical application effect of fall detection methods in elderly care scenarios

After verifying the fall detection method in elderly care scenarios based on optimized YOLOv algorithm, the detection effect of the research method under different lighting conditions is verified. The study recorded 80% of unsupervised falls at night, including the time and duration of posture retention, as a dataset, and built an intelligent hospital bed behavior monitoring platform. In research, "composite posture" refers to a complex posture that involves multiple directions or continuous movements during a fall, such as first tilting and then forward tilting. "Assisting behavior" refers to the act of caregivers or assistive devices providing physical support to elderly people. The ground truth annotation is completed by three independent annotators, and the final label is determined by a majority vote to ensure consistency in annotation. The nighttime unsupervised falls data used in the study were derived from anonymized surveillance records in partnership with a hospital/institution. For the protection of subject privacy and data security, this portion of data will not be made public. The data is used only for the research and there are no plans to make it public in the future. The designed method is compared with CV-FD and WS-FD. The missed detection rates of three methods under different types of falls and the false alarm rates under different behavioral scenarios are compared, as illustrated in Figure 9.

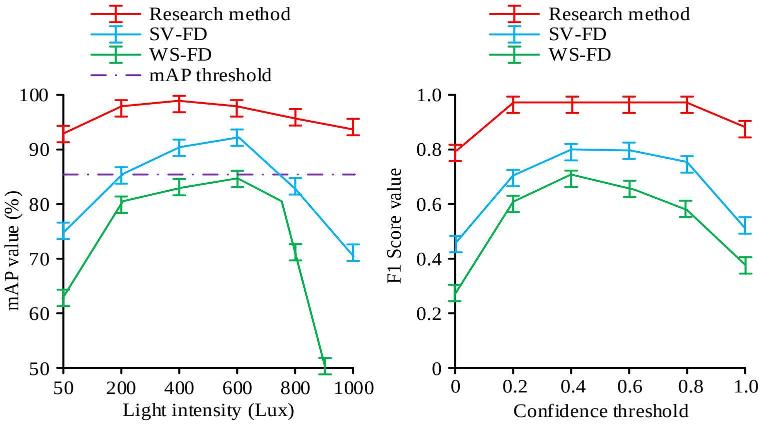
From Figure 9, the three methods had significant differences in missed detection rates under different types of falls and false alarm rates under different behavioral scenarios. From Figure 9 (a), the missed detection rate of the research method was 2.3% in anteversion falls, 1.5% in tilt back falls, 4.5% in lateral falls, and 7.2% in composite postures. The missed detection rates of the other two methods were significantly higher than those of the research method in various types of falls. As shown in Figure 9 (b), the false alarm rates of the three methods showed an increasing trend with the complexity of the behavioral scene. The false alarm rate of the research method was 0.8% under stationary behavior, 1.5% under walking behavior, 3.2% under sitting and lying behavior, and 3.9% under supporting behavior. The false alarm rates of the other two methods were significantly higher in non-stationary behavior scenarios than in the research method. Overall, compared to the comparative methods, the research method has better detection robustness, stability, and anti-interference ability. The mAP values of the three methods under different lighting conditions, as well as the F1 Score values changing with confidence threshold, are compared, as illustrated in Figure 10.



(a) Missed detection rates under different types of falls

(b) False alarm rate in different behavioral scenarios

Figure 9: Missed detection rate under different types of falls and false alarm rate under different behavioral scenarios

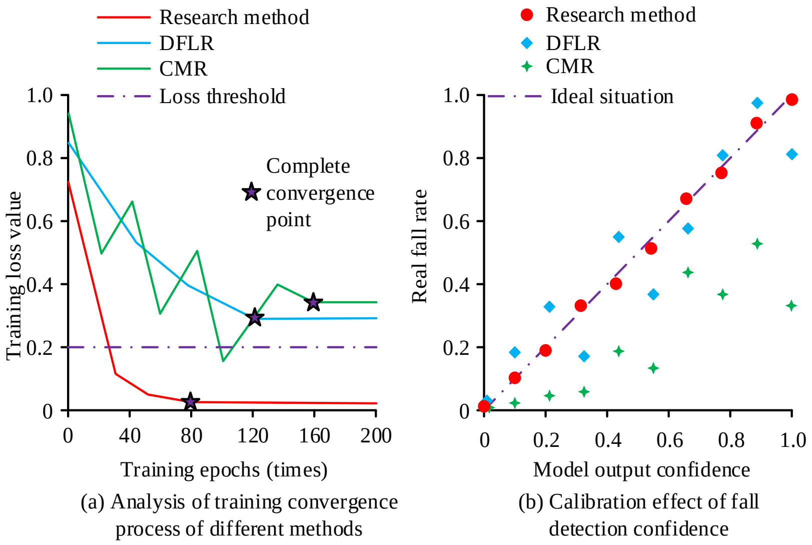


(a) mAP values under different lighting conditions

(b) The trend of F1 Score value changing with confidence threshold

Figure 10: The variation trend of mAP value with light intensity and F1 Score value with confidence threshold

In Figure 10 (a), the results shown in the figure are the mean of 5-fold cross validation, and the error bars represent the standard deviation. The mAP threshold of the system was 85 ± 0.5% under different light intensities. The mAP value of the research method was generally above the threshold with increasing light intensity. The mAP value of the research method was 92.3 ± 0.5% at a light intensity of 50Lux, and reached its maximum value of 97.4 ± 0.5% at 400Lux. Then, it gradually decreased with increasing light intensity, and decreased to 93.6 ± 0.5% at 1,000Lux. The mAP values of the other two methods were significantly lower than those of the research method under various light intensities. In Figure 10 (b), in the model output stage, a dynamic confidence threshold strategy is adopted to filter the detection results by setting different confidence thresholds (range 0.0~1.0, step size 0.1). There were significant differences in the F1 Score values of the three methods as the confidence threshold changed. The overall F1 Score value of the research method at each confidence threshold was greater than 0.8 ± 0.05, and it remained stable at 0.98 ± 0.05 within the confidence threshold range of 0.2-0.8, ultimately decreasing to 0.85 ± 0.05 at the confidence threshold of 1.0. The F1 Score values of the other two methods were significantly lower than those of the research method at each confidence threshold. In practical application scenarios, the performance of three methods under different lighting and behavioral scenarios was subjected to repeated ANOVA. The results showed that the research method was significantly better than CV-FD and WS-FD in mAP, F1 Score and other indicators (p<0.05), and maintained stable performance even under changes in lighting and increased behavioral complexity. Overall, compared to the comparative methods, the research method has better illumination robustness, feature extraction ability, generalization, and discrimination clarity. The convergence of the training loss values of the three methods with training epochs and the calibration effect of fall detection confidence are analyzed, and the results are shown in Figure 11.



(a) Analysis of training convergence process of different methods

(b) Calibration effect of fall detection confidence

Figure 11: Analysis of training convergence process and confidence calibration effect of fall detection

From Figure 11 (a), the training loss threshold of the system was 0.2. The training loss value of the research method rapidly decreased when the training epochs were less than 30, and then the downward trend slowed down. Finally, it fully converged at the 80th training epochs, with a stable value of 0.02 at full convergence. The stable loss values of the other two methods were both above the loss threshold when they fully converged. The WS-FD method fluctuated violently during the convergence process and finally fully converged at the 160th training session. From Figure 11 (b), the ideal distribution of the true fall ratio of the system in the model output confidence was a diagonal distribution. The actual fall rate of the research method was distributed around the diagonal as a whole under the confidence level. The true fall rate of the other two methods under the confidence level is significantly further away from the diagonal than the research method, and the distribution of the WS-FD method is generally lower than the diagonal. Overall, compared to comparative methods, the research method has better convergence, training efficiency, reliability, and risk controllability. Overall, the fall detection method for elderly care scenarios based on optimized YOLOv algorithm has good robustness, stability, generalization, convergence, and training efficiency.

To further verify the reliability of the system in actual deployment, a network constrained environment was simulated to test the behavior of alarm transmission. Under the bandwidth limitation of 1 Mbps, the successful transmission rate of system alarms reached 98.7%, with an average transmission latency of 210 milliseconds, indicating that the research method still has good alarm response capability in network limited scenarios. Finally, to evaluate the contribution of each optimization module to model performance and efficiency, ablation experiments were conducted on CSCD-YOLOv5 and MDF-YOLOv8, respectively. All tests were conducted on Jetson AGX Orin with an input image size of 640 × 640. (1) CSCD-YOLOv5 module analysis: The baseline YOLOv5 model parameters are 7.0M, FLOPs are 16.2G, latency is 12.3ms, model size is 13.5MB, and memory usage is 1.2GB. After adding the CBWT3 module, the parameters increase by 0.8M, FLOPs increase by 2.1G, latency increases by 1.5ms, model size increases to 14.8MB, and memory usage increases to 1.4GB. After introducing the SPPELAN module, the parameters increase by 1.2M, FLOPs increase by 3.5G, latency increases by 2.1ms, model size increases to 16.9MB, and memory usage increases to 1.6GB. Combining the C3-CGB and Detect-FR modules, the total parameters reach 9.5M, FLOPs are 23.8G, latency is 17.9ms, model size is 18.3MB, and memory usage increases to 1.6GB. The usage is 1.9GB. (2) MDF-YOLOv8 module analysis: The baseline YOLOv8 model parameters are 3.2M, FLOPs are 8.7G, latency is 9.8ms, model size is 6.1MB, and memory usage is 0.9GB. After introducing the C2f-MSCS module, the parameters increase by 0.6 M. FLOPs increase by 1.8G, latency increases by 1.2ms, model size increases to 7.2MB, and memory usage increases to 1.1GB. After adding the DEF module, the parameters increase by 0.9M, FLOPs increase by 2.4G, latency increases by 1.7ms, model size increases to 8.5MB, and memory usage increases to 1.3GB.

To quantify the contribution of each module to detection performance, C2f-Faster Block is taken to replace the original C2f module, reducing parameters by 0.3M, FLOPs by 0.9G, latency by 0.8ms, model size by 7.9MB, and memory usage by 1.2GB. Using YOLOv5 as the baseline model, CBWT3, SPPELAN, C3-CGB, and Detect-FR modules are gradually added, and their mAP, accuracy, recall, and inference FPS changes are recorded on the test set. Among them, the results of the baseline YOLOv5 module are: mAP of 89.2%, accuracy of 90.1%, recall of 88.5%, and inference FPS of 102.3 frames per second. The results of the CBWT3 module show that mAP has increased to 91.5%, accuracy has increased to 91.8%, recall has increased to 90.2%, and inference FPS has slightly decreased to 98.7 frames per second, indicating that the module has enhanced its feature extraction capability. The results of the SPPELAN module show that mAP is further improved to 93.7%, accuracy is 93.4%, recall is 92.1%, and inference FPS is 95.4 frames per second, verifying the adaptability of multi-scale pooling to complex scenes. The results of the C3-CGB module show that mAP reaches 95.3%, accuracy is 95.0%, recall is 93.8%, and inference FPS is 91.2 frames per second, indicating that the context guided mechanism improves robustness. The results of the Detect-FR detection head show that the final mAP reaches 97.4%, accuracy is 96.9%, recall is 96.5%, and inference FPS is 87.6 frames per second, proving that decoupling the detection head effectively improves classification and localization accuracy. The results indicate that each module contributes positively to performance improvement, with SPPELAN and Detect-FR showing the most significant improvement in mAP, while CBWT3 and C3-CGB mainly enhance the model's discriminative ability under complex behavior.

Although the inference speed slightly decreases with the increase of modules, it still remains above the real-time detection threshold. Sensitivity analysis is conducted to evaluate the impact of hyperparameters on model performance in low-light enhancement (DEF module) and attention mechanism (C2f-MSCS module). The experiment is conducted under low-light conditions (50 Lux), with the mAP as the evaluation index, and the following hyperparameters are analyzed in detail: The number of attention heads is set to 4, 8, and 16. The DEF module channel expansion factor values are 1, 2, and 4. When the number of attention heads is 8 and the expansion factor is 2, the mAP is highest (93.1%). Too many (16) or too few (4) attention heads can lead to performance degradation, indicating that a moderate number of attention heads can balance feature capture and computational efficiency. When the channel expansion factor is 2, the performance is optimal. If the expansion factor is too large (4), it is easy to introduce redundant features. In summary, the study ultimately selects 8 attention heads and 2 expansion factors as hyperparameter configurations. In addition, the research method mainly relies on single frame images for detection, and does not fully utilize the motion information between consecutive frames. Therefore, there may be discrimination limitations when dealing with continuous behaviors similar to falling postures, such as slow squatting and standing up. Subsequent research will introduce temporal modeling modules (such as RNN or Video Transformer) to enhance the ability to model behavioral continuity.

5 Summary

Aiming at the low compliance, high false alarm rate, and insufficient reliability of existing fall detection methods in daily fall detection, an innovative optimized YOLOv was proposed. The research aimed to optimize the YOLOv5 and YOLOv8 algorithms to enhance detection performance in complex behavioral environments and low-light conditions. The recall-precision curve of the research method exhibited an overall high recall-precision. When the recall was 98.5%, the precision was 93.8%. When the input image size increased from 160*160 to 960*960, the inference speed of the research method only decreased by 19.2 frames/s. The correct detection rate of the research method was 93.4% when the human occlusion area was 80%. In practical applications, the F1 Score remained stable at 0.98 within the confidence threshold range of 0.2-0.8, and ultimately decreased to 0.85 when the confidence threshold was 1.0. The training loss value of the research method converged completely at the 80th training session, and its stable value at complete convergence was 0.02. The above results indicate that the proposed model has good robustness, stability, generalization, convergence, and training ability.

Subsequent research will further introduce advanced control methods such as fractional order chaotic system synchronization control and output feedback projection lag synchronization to enhance the modeling ability of nonlinear dynamics in continuous motion sequences to improve detection accuracy and system stability in extreme environments. In addition, neural adaptive control methods in control theory, such as neural adaptive control of nonlinear systems, are also committed to maintaining system stability and reliability in complex and uncertain environments. This type of method responds to system dynamic changes and external disturbances by adjusting controller parameters online, which is conceptually similar to the environment adaptive fall detection model proposed in this study. Both emphasize that the system should have self-regulation ability to cope with uncertainty: Neural adaptive control ensures system convergence through Lyapunov stability theory, while this study achieves robust response to environmental changes through multi-scale feature fusion, attention mechanism, and low-light enhancement module. However, neural adaptive control focuses more on the design of control laws for continuous dynamic systems, while this study focuses on feature extraction and classification in discrete visual perception tasks.

In future research, it is possible to integrate uncertainty quantification methods in neural adaptive control into the loss function or network structure of the YOLOv algorithm to further enhance the model's generalization ability under extreme noise and occlusion conditions. Moreover, to achieve faster intervention response, real-time scheduling and inference acceleration strategies can be introduced in subsequent work to ensure that fall detection is completed within a fixed time window. For example, through model lightweight, hardware acceleration or edge computing optimization, the detection latency is controlled at the millisecond level, so that an alarm or linkage rescue system can be triggered in a very short time after a fall, minimizing the injury. To achieve the application of psychological and physiological personalization in the human environment system, further research can be combined with psychological and physiological signals to construct a personalized fall risk assessment and rehabilitation intervention system. For example, in robot assisted rehabilitation scenarios, the system can dynamically adjust the rehabilitation training intensity or provide personalized warnings based on the user's real-time physiological state and behavior patterns. In an intelligent nursing environment, a fall detection model that integrates psychological states can distinguish abnormal behaviors caused by emotional fluctuations, enhancing the system's situational understanding and humanized interaction capabilities. In addition, the proposed environment adaptive detection framework has strong transfer potential and can be extended to multiple fields such as robot environment perception, medical behavior monitoring, and intelligent security. For example, in service robots, it can be used for recognizing abnormal human posture, and in ward monitoring, it can achieve intelligent monitoring of patient behaviors such as falls and getting out of bed, which has good engineering application prospects.

References

[1] Mekruksavanich S, Jitpattanakul A. Fallnext: A deep residual model based on multi-branch aggregation for sensor-based fall detection. ECTI Transactions on Computer and Information Technology (ECTI-CIT), 2022, 16(4): 352-364. DOI: 10.37936/ecti-cit.2022164.248156.

[2] Butt A, Narejo S, Anjum M R, Yonus M U, Memon M, Samejo A A. Fall detection using LSTM and transfer learning. Wireless Personal Communications, 2022, 126(2): 1733-1750. DOI: 10.1007/s11277-022-09819-3.

[3] Kaur N, Rani S, Kaur S. Real-time video surveillance based human fall detection system using hybrid haar cascade classifier. Multimedia Tools and Applications, 2024, 83(28): 71599-71617. DOI: 10.1007/s11042-024-18305-w.

[4] Gai R, Chen N, Yuan H. A detection algorithm for cherry fruits based on the improved YOLO-v4 model. Neural Computing and Applications, 2023, 35(19): 13895-13906. DOI: 10.1007/s00521-021-06029-z.

[5] Haifeng Z, Xinchun L U, Bo F, Jin Y. Marine fish species recognition based on improved YOLOv 5s. Food and Machinery, 2025, 40(8): 84-92. DOI: 10.13652/j.spjx.1003.5788.2023.81080.

[6] Nooruddin S, Islam M M, Sharna F A, Alhetari H, Kabir M N. Sensor-based fall detection systems: a review. Journal of Ambient Intelligence and Humanized Computing, 2022, 13(5): 2735-2751. DOI: 10.1007/s12652-021-03248-z.

[7] Gaya-Morey F X, Manresa-Yee C, Buades-Rubio J M. Deep learning for computer vision-based activity recognition and fall detection of the elderly: a systematic review. Applied Intelligence, 2024, 54(19): 8982-9007. DOI: 10.1007/s10489-024-05645-1.

[8] Durga Bhavani K, Ferni Ukrit M. Design of inception with deep convolutional neural network-based fall detection and classification model. Multimedia Tools and Applications, 2024, 83(8): 23799-23817. DOI: 10.1007/s11042-023-16476-6.

[9] Qian Z, Lin Y, Jing W, Ma Z, Liu H, Yin R., et al. Development of a real-time wearable fall detection system in the context of Internet of Things. IEEE internet of things journal, 2022, 9(21): 21999-22007. DOI: 10.1109/JIOT.2022.3181701.

[10] Jain R, Semwal V B. A novel feature extraction method for preimpact fall detection system using deep learning and wearable sensors. IEEE Sensors Journal, 2022, 22(23): 22943-22951. DOI: 10.1109/JSEN.2022.3213814.

[11] Xie Y, Hu W, Xie S, He L. Surface defect detection algorithm based on feature-enhanced YOLO. Cognitive Computation, 2023, 15(2): 565-579. DOI: 10.1007/s12559-022-10061-z.

[12] Lian W, Jun L I U, Jie P I, Daoying W. Image recognition algorithm for pork freshness based on YOLOv 8n. Food and Machinery, 2025, 41(5): 98-104. DOI: 10.13652/j.spjx.1003.5788.2024.80882.

[13] Liu Z. Unmanned Aerial Vehicles General Aerial Person-Vehicle Recognition Based on Improved YOLOv8s Algorithm.Computers, Materials & Continua, 2024, 78(3):3787-3803. DOI: 10.32604/cmc.2024.048998.

[14] Hao Z. Method for Identifying Motor Vehicle Traffic Violations Based on Improved YOLOv Network. Scalable Computing: Practice and Experience, 2023, 24(3): 217-228. DOI: 10.12694/scpe.v24i3.2335.

[15] Zhang W, Wang K, Zhou X, Shi L, Song X. MobileOne-YOLO: An Improved Real-Time Fire Detection Algorithm for Aircraft Cargo Compartments. Journal of Aerospace Information Systems, 2025, 22(6): 447-456. DOI: 10.2514/1.I011514.

[16] Chen S, Yang W, Xu Y, Geng Y, Xin B, Huang L. AFall: Wi-Fi-based device-free fall detection system using spatial angle of arrival. IEEE Transactions on

Mobile Computing, 2022, 22(8): 4471-4484. DOI: 10.1109/TMC.2022.3157666.

[17] Du Z, Feng X, Li F, Xian Q, Jia, Z. A Lightweight UAV Visual Obstacle Avoidance Algorithm Based on Improved YOLOv8. Computers, Materials & Continua, 2024, 81(2):2607-2627. DOI: 10.32604/cmc.2024.056616.

[18] Lv S, Tao C, Hao Z, Ni H, Hou Z, Li X, et al. Research on strip surface defect detection based on improved YOLOv5 algorithm. Ironmaking & Steelmaking, 2024, 51(10): 1046-1064. DOI: 10.1177/03019233241260922.

[19] Liu Z. Unmanned Aerial Vehicles General Aerial Person-Vehicle Recognition Based on Improved YOLOv8s Algorithm.Computers, Materials & Continua, 2024, 78(3):3787-3803. DOI: 10.32604/cmc.2024.048998.

[20] Du Z, Feng X, Li F, Xian Q, Jia, Z. A Lightweight UAV Visual Obstacle Avoidance Algorithm Based on Improved YOLOv8.Computers, Materials & Continua, 2024, 81(2):2607-2627. DOI: 10.32604/cmc.2024.056616.

[21] Charfi I, Miteran J, Dubois J, Atri M, Tourki R. Definition and performance evaluation of a robust SVM based fall detection solution. In: 2012 Eighth International Conference on Signal Image Technology and Internet Based Systems. 2012: 218-224. DOI: 10.1109/SITIS.2012.84.