Journal of Electronic Research and Application, 2026, Volume 10, Issue 5
http://ojs.bbwpublisher.com/index.php/JERA
Online ISSN: 2208-3510
Print ISSN: 2208-3502

Passenger Escalator Fall Detection Algorithm Based on SCGD-Yolo11m-Pose

Huidi Zhang, Shijie Jia*, Shaoyuan Xu

Dalian Jiaotong University, Dalian 116028, Liaoning, China

Corresponding author: Shijie Jia, jsj@djtu.edu.cn

Abstract: Escalator fall detection algorithms in subway stations are a crucial means of preventing passenger accidents. However, in a real scene, issues such as the loss of small objects due to scale changes and interference caused by complex backgrounds can lead to false positives and false negatives. This paper proposes a passenger escalator fall detection algorithm based on SCGD-Yolo11m-pose network. First, RFD module was introduced during the downsampling stage to improve the robustness of feature extraction. Second, in C2PSA, deformable attention was used and named C2DA. This enables the model to enhance its ability to perceive various falling poses in complex backgrounds. Finally, in neck network, Gold-Yolo structure replaces the PANet network to strengthen the recombination ability of multi-scale features and improves the accuracy of the model in complex background. Additionally, only four keypoints defined by COCO, both shoulders and both hips, are retained, which improves computational efficiency. Experimental results on our self-built subway escalator fall dataset show that the improved model is improved by 2.3% on AP50 and 2.5% on AP50:95. This validates the effectiveness and practicality of the proposed algorithm in ensuring the safety monitoring of subway passengers.

Keywords: Pose estimation; Escalator; Fall detectionYOLOv11-pose; Deformable attention; Gold-YOLO

Online publication: Jun 29, 2026

1. Introduction

As an important part of urban transit, the safety of subway passengers has received much attention. The construction and operation of intelligent subway safety monitoring technology are crucial to ensuring passenger travel safety. Escalator fall detection for passengers, as one of the key technologies in subway safety monitoring, can promptly warn of potential safety risks and alarm for dangerous behaviors that occur by recognizing and analyzing specific actions, thus reducing casualties as quickly as possible.

For algorithms recognizing passenger falls on escalators, deep learning-based methods are currently the absolute mainstream and cutting-edge direction, using neural networks to automatically learn features, greatly improving accuracy and robustness [1]. These methods can mainly be divided into two categories:

pose estimation-based methods and object detection-based methods [2–4]. Yu proposed a method combining OpenPose and YOLOv4 algorithms to achieve high-precision detection of human keypoints, and then used an MLP classifier to classify the human keypoints detected on escalators in an automated passenger safety detection system [5]. Yang proposed an escalator passenger fall detection algorithm using Swin Transformer as the backbone network, with a neck network adopting a YOLOX model fused with the CBAM attention mechanism [6]. Liang proposed a real-time escalator passenger monitoring algorithm, YOLOv5-CBAM-WBF, introducing an attention module into the neck network and using ShuffleNet v2 and GhostNet modules to lightweight the network [7]. Wang *et al.* proposed a lightweight real-time escalator passenger abnormal behavior detection algorithm, YOLO-STE, based on the YOLOv5s algorithm, and implemented deployment on Jetson Nano hardware [8]. Yang *et al.* proposed a method for recognizing abnormal escalator passenger behavior based on the human skeleton, using the Alphapose algorithm to extract skeletal keypoint coordinates, then using a spatiotemporal graph convolutional network to extract passenger skeleton information, and finally classifying abnormal actions through Softmax [9]. Hou *et al.* proposed a lightweight escalator passenger fall detection algorithm based on an improved YOLOv8 for regions of interest, shielding complex environmental interference in non-escalator background areas and effectively improving the detection rate of small targets [10].

Methods based on pose estimation doesn't focus on the pixels of the entire person; instead, they first locate the body's keypoints (such as the head, neck, shoulders, elbows, wrists, hips, knees, and ankles) and use logical judgments based on this keypoint information to detect falls. This method is computationally efficient and requires less data. It also has strong occlusion resistance; even if part of the body is obscured, a fall can still be detected as long as the keypoints remain visible. Object detection-based methods use object detection algorithms to directly identify human figures in a "falling" state within video frames. They require large datasets containing various falling and normal postures to train the model to recognize and box out the "falling" behavior. However, data collection and annotation costs are prohibitively high, and the model's generalization ability is weak for unusual falling postures not covered by the training data, potentially leading to false negatives.

Regarding the current research on escalator passenger fall detection algorithms, the automated escalator operating environment is relatively complex, with severe background interference, numerous pedestrians, constantly changing passenger scales, and the detection of small distant passenger targets easily leading to missed and false detections [10]. This paper proposes a passenger escalator fall detection algorithm based on SCGD-Yolo11m-pose. The improved algorithm introduces robust feature downsampling SRFD and DRFD into the backbone network feature extraction for downsampling, replacing the traditional downsampling method that uses a stride-2 convolution to reduce feature map size, which solves the problem of fine-grained feature blurring or loss. A C2DA module is designed, improving the self-attention module of C2PSA module in the original Yolo11 into a deformable self-attention module (DAT), effectively capturing key information in the target area. Gold-Yolo is adopted to replace the Neck part of Yolo11m-pose, improving information loss during cross-layer fusion in the neck network. At the same time, the 17 keypoints defined in the COCO keypoint dataset are reduced to only 4 keypoints (both shoulders and both hips) for recognition, reducing computational complexity. By building a custom escalator monitoring viewpoint dataset, passenger escalator fall detection is achieved.

2. Improved network architecture design

2.1. SCGD-YOLO11m-pose network architecture

This study focuses on the task of passenger pose estimation on escalators. Based on the YOLO11m-pose algorithm model, improvements and optimizations were made so that the improved algorithm can meet the requirements for real-time monitoring and early warning of escalator passenger falls [11]. The YOLO11m-pose network structure is mainly composed of the following three parts:

(1) Backbone layer: A series of convolutional layers is used to convert original images into feature map;

(2) Neck layer: Perform feature fusion on the feature information output from the backbone;

(3) Head layer: Responsible for generating the final prediction from the fused features. In YOLO11m-pose, it is mainly divided into the object detection part and the pose estimation part. The object detection part outputs the bounding box coordinates and confidence; the pose estimation part analyzes the detected human targets and outputs the coordinates of each key point along with the corresponding visibility confidence.

As shown in **Figure 1**, this is the improved SCGD-Yolo11m-pose network structure diagram. SRFD (Shallow Robust Feature Downsampling) and DRFD (Deep Robust Feature Downsampling) modules are introduced in the downsampling stage of the backbone network. By fusing feature maps from different downsampling strategies, they effectively mitigate the issue of small target feature loss caused by traditional convolutional downsampling and enhance the robustness of feature extraction. In addition, to address target spatial scale changes in complex backgrounds, the deformable attention mechanism replaces self-attention mechanism in the C2PSA to construct C2DA module, which dynamically adjusts the shape and size of attention, better adapting to the spatial and scale variations of targets in complex backgrounds. This allows the model to adaptively focus on the key regions of the target, improving its ability to perceive falling postures. Finally, in the neck network part, the Gold-Yolo structure is used to replace the original PANet, utilizing its efficient global information fusion mechanism to strengthen the reorganization capability of multi-scale features and increase model's detection accuracy in complex backgrounds [12].

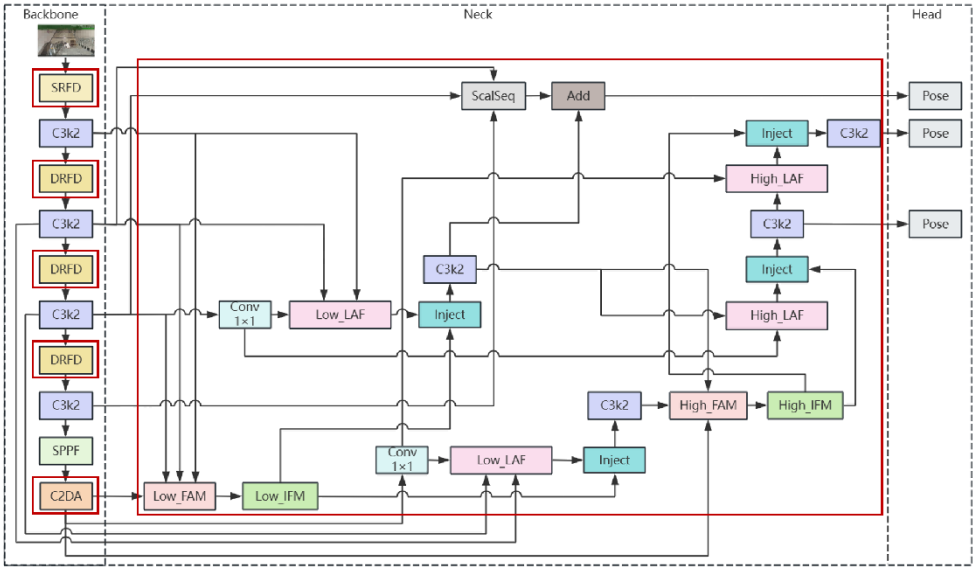


Figure 1. Network model structure of improved SCGD-YOLO11m-pose algorithm.

2.2. RFD (SRFD and DRFD)

In most backbone networks, although the convolution with stride of 2 is usually used to downsample and reduce the size of feature map to integrate local information, it has limited ability to retain fine-grained details. To address this issue, this paper adopts the Robust Feature Downsampling (RFD) structure proposed by Lu *et al.* [13]. Robust Feature Downsampling (RFD) is a new general-purpose downsampling module that includes Shallow Robust Feature Downsampling (SRFD) and Deep Robust Feature Downsampling (DRFD). By integrating features from multiple downsampling methods, it allows for the extraction of more reliable downsampled feature maps, enabling subsequent feature learning layers to capture key information more effectively and thereby improving the overall performance of the network.

When the input information received by the backbone network contains a large amount of redundant information, to improve performance by retaining significant features and filtering out shallow redundant information, shallow robust feature downsampling (SRFD) is used. The SRFD structure is shown in **Figure 2.**

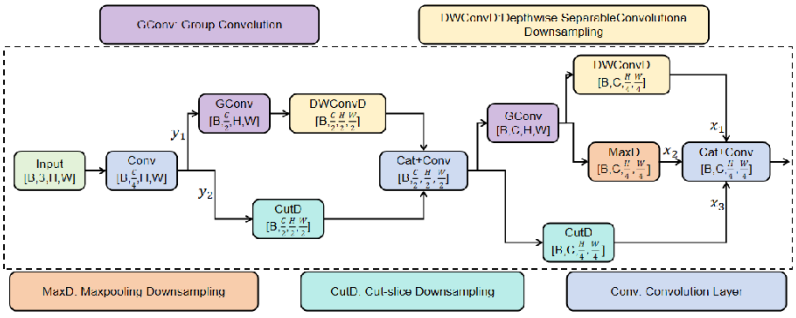


Figure 2. SRFD module structure.

First, a 7×7 convolution with stride 1 is applied to 640×640×3 input image to obtain a key feature map without redundant information in order to achieve feature enhancement. y_1 undergoes a 3×3 group convolution with a stride of 1 and then downsampled by a 3×3 depthwise convolution with a stride of 2, thus integrating local feature information, improving feature fusion and halving the feature size. By using grouped convolution and depthwise convolution, compared with traditional convolutional downsampling, this method reduces the network's floating-point operations. This method is named D_{conv}, as shown in **Equation (1)**. The y_2 branch undergoes slicing downsampling, named D_{cut}, as shown in **Equation (3)**. Then, y_1 and y_2 are concatenated and reduced the number of channels with 1×1 convolution, and three different methods are applied to the output features at this point to obtain output feature maps x_1,x_2 and x_3. First, x_1 is obtained through D_{conv}, achieving a 2× reduction in spatial dimensions; secondly, x'_2 is generated using grouped convolution with parameters shared with D_{conv}. After that, a 2×2 max pooling operation with stride 2 is applied to x'_2 to perform feature downsampling, resulting in x_2, which is named D_{max}, as shown in **Equation (4)**; x_3 is obtained by downsampling with D_{cut}. Finally, three branches x_1, x_2 and x_3 with the size of 160×160×128 pass through Concat and use 1×1 convolution to keep the number of 128 channels.

$$D_{conv} = BN\left(DWConvD\left(GConv\left(x\right)\right)\right) \tag{1}$$

$$fusion = BN\left(Conv\left(Concat\left(x,\ \ y,\ \ z\right)\right)\right) \tag{2}$$

$$D_{cut} = fusion\left(c_1,\ c_2,\ c_3,\ c_4\right) \tag{3}$$

$$D_{max} = BN\left(maxpooling\left(GConv\left(x\right)\right)\right) \tag{4}$$

D_{cut} uses cut slice as shown in **Figure 3**. Cut refers to dividing adjacent pixels into c_1, c_2, c_3, and c_4; then the feature maps c_1, c_2, c_3, and c_4 are connected through Concat; finally, a 1×1 convolution is used to reduce the number of channels from 4C to 2C. By slicing adjacent pixels, the size of the feature map is effectively reduced while preserving the original feature information, halving the original dimensions.

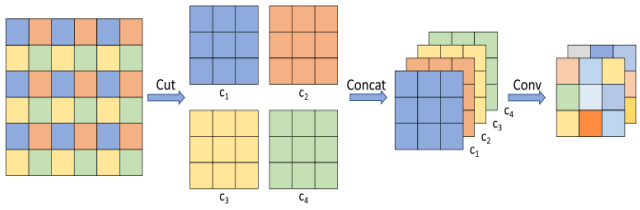


Figure 3. Cut slice.

To prevent convolution from losing rich semantic information during the downsampling process, a deep robust feature downsampling(DRFD) structure is adopted. The DRFD structure is shown in **Figure 4**. The input feature map y is downsampled through D_{conv}, D_{cut}, and D_{max} to get the output feature maps x_1, x_2, and x_3, respectively. x_1 is obtained through D_{conv} and simultaneously activated by GELU, sownsampling with D_{max} yields x_2, downsampling with D_{cut} produces x_3 [14]. The three operations respectively reduce the feature size by half and increase the number of channels from C to 2C. Finally, x_1, x_2, and x_3 are concatenated and passed through a 1×1 convolution, first increasing the number of channels of feature map to 6C and then reducing it to 2C. The feature map of the backbone network improved by RFD module is the same as that of the original network, and P2~P5 are still 160×160×128, 80×80×256, 40×40×512 and 20×20×1024.

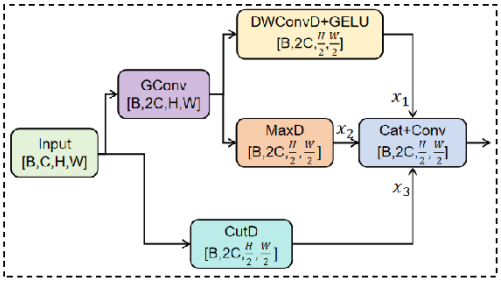


Figure 4. DRFD module structure.

2.3. C2DA

Due to the wide field of view of surveillance cameras, background-irrelevant areas occupy the majority of images in the dataset. This phenomenon directly leads to missed detections and false detections of escalator passengers' postures, affecting the model's detection accuracy. Traditional attention mechanisms usually calculate weights using position-fixed attention models. Although this can improve feature extraction to some extent, it is difficult to adapt to spatial transformations and size variations of different targets in complex image backgrounds. In contrast, deformable attention mechanisms can dynamically adjust the shape and

size of attention, better adapting to characteristics of different tasks and inputs, and especially effectively capturing key information of target areas in complex backgrounds.

Therefore, this paper introduces a simple yet effective Deformable Attention module (Deformable Attention) into the Yolo11m network. DAttention is used to enhance the self-attention mechanism of the C2PSA module in the Yolo11m architecture, and the improved module is named C2DA. The C2DA module first expands the channel into 2×512 by using 1×1 convolution, and then splits the channel into two parts with a separation ratio of 0.5. One part passes through a DABlock module, while the other part is concatenated with the processed branch along the channel dimension via a shortcut, and finally, a 1×1 convolution is applied to restore the original number of channels. The DABlock module serves as the core computational unit, each containing a DAttention module and a feedforward network (FFN), both connected through shortcuts. The FFN consists of two consecutive 1×1 convolutional layers. The first convolutional layer doubles the number of channels, and the second layer restores the channel count to its original dimension, thereby providing additional complex non-linear fitting capability, which helps the model learn richer feature representations. The C2DA structure is illustrated in **Figure 5**.

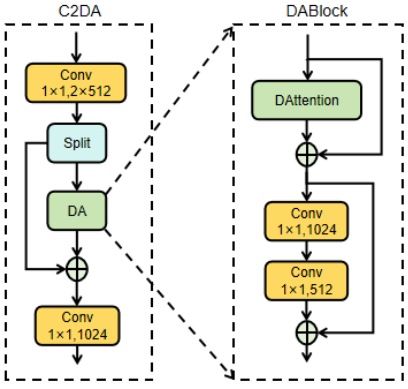


Figure 5. C2DA module structure.

The deformable attention mechanism, DAttention, was proposed by Xia *et al.* [15]. It is a novel deformable attention module that improves the traditional attention mechanism through deformable convolutions. It can select keypoints and value pairs in self-attention according to the needs of the data, enabling the model to focus on relevant regions and capture more informative features. This module enhances the model's feature extraction capability while adding almost no extra parameters.

Figure 6 illustrates a deformable attention module. At the beginning, a group of reference points are uniformly placed on the feature map and the offset of these points is learned from the query through the offset network. These offsets indicate the positional variations of the reference points, enabling the model to flexibly attend to information at different locations. Then, as shown on the right side, based on these deformed reference points, the keys and values in the feature map are dynamically sampled and adjusted through projection, resulting in deformed key-value pairs. In this manner, the model can focus on key targets across different regions of the feature map while ignoring irrelevant background information. There are only four reference points showing in the figure, but in fact, more points will actually help the model capture richer information [16].

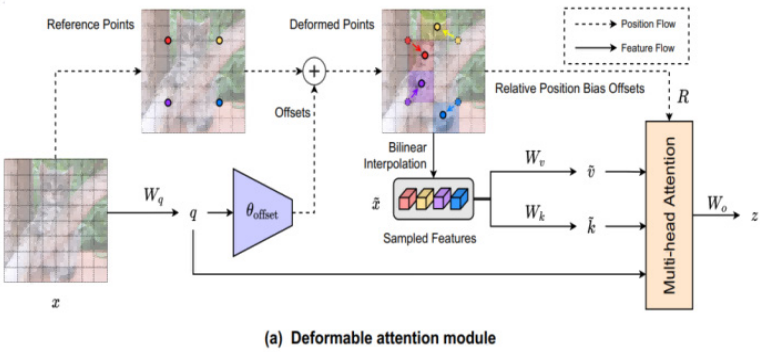


Figure 6. Deformable attention module.

2.4. Gold-Yolo

In the YOLO series network architectures, the neck typically employs a FPN for multi-level feature fusion. However, its efficiency is relatively low during cross-level propagation. To reduce information loss during cross-level fusion in neck network, this paper adopts the structure of Gold-YOLO, open-sourced by Huawei Noah's Ark Lab in 2023, which improves the neck of Yolo11m-pose [17]. As shown in **Figure 7**, this structure introduces an aggregation-distribution mechanism to enhance the method of information fusion, and designs two branches: Low-GD and High-GD. Low-GD replaces the upsampling fusion stage of PANet, mainly aggregating shallow-level information to improve large object detection capability, while High-GD replaces the downsampling fusion stage of PANet, integrating deep semantic information to enhance small object detection capability [18].

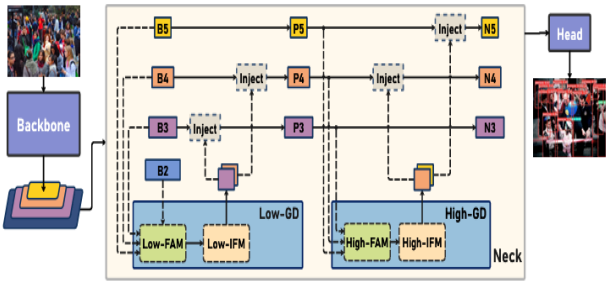


Figure 2: The architecture of the proposed Gold-YOLO.

Figure 7. The architecture of Gold-YOLO.

The GD implementation of the collect-and-distribute process includes three modules: the Feature Alignment Module (FAM), the Feature Information Fusion Module (IFM), and the Feature Information Distribution Module (Inject). First, FAM collects feature maps from different scales of the backbone and aligns them through upsampling or downsampling; then, IFM fuses the aligned features to generate global features, which are then split into two parts for targeted distribution to other scales; finally, the Inject distribution module, using a self-attention-like approach with simple attention operations, enhances the detection capability of the branches by distributing the split global features to different levels.

As shown in **Figure 8(a)**, Low-FAM needs to consider both making the feature maps as large as possible to retain more low-level information and preventing feature maps from becoming too large, which would increase computational burden. Therefore, a compromise is made using B4 as the reference: larger feature maps B2 and B3 are downsampled through average pooling, while the smaller feature map B5 uses bilinear interpolation to upsample, achieving uniform feature map sizes. Then, concatenation is performed to obtain

the merged feature Falign.

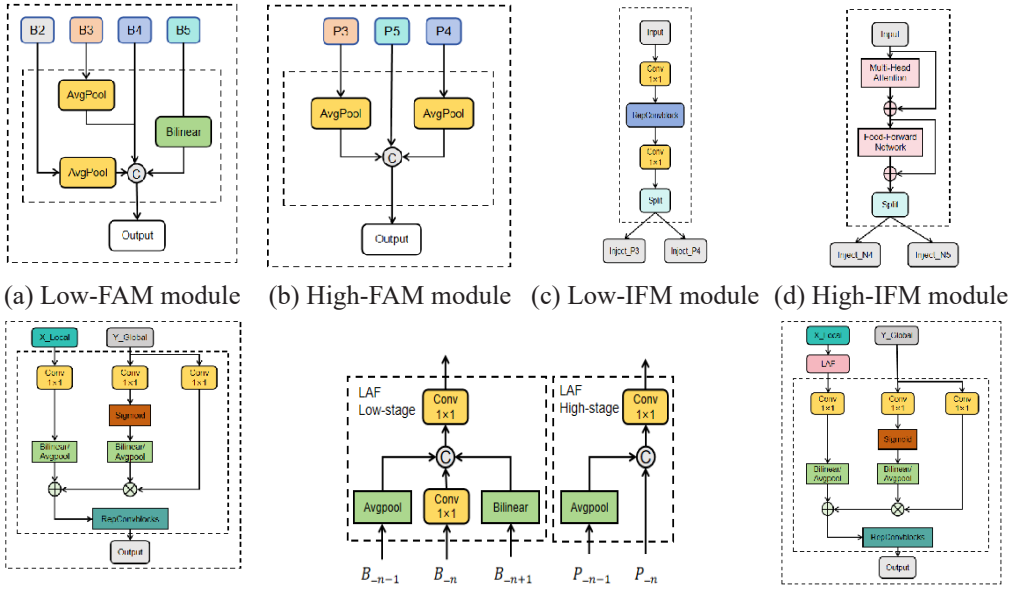
As shown in **Figure 8(b)**, High-FAM is composed of AvgPool average pooling, which reduces the size of the input features to a unified dimension through average pooling. First, P3 and P4 are downscaled to the size of P5, and then the feature maps are merged through concatenation. This module extracts larger-sized features and reduces the computational complexity of subsequent operations.

As shown in **Figure 8(c)**, the Low-IFM design includes Conv, the RepBlock module based on RepVGG theory, and the Split operation. Specifically, the Low-FAM module-aligned and concatenated features F_align are input into the RepBlock module to obtain the fused features F_fuse, while Conv adjusts the number of channels to suit models of different sizes. F_fuse is split along the channels into Finj_P3 with 256 channels and Finj_P4 with 512 channels, which are then fused with different levels of features in the next step.

As shown in **Figure 8(d)**, High-IFM mainly consists of the Multi-Head Attention module, Feed-Forward Network, and Split. The features Falign obtained from High-FAM are first processed through four-headed Multi-Head Attention and Feed-Forward Network to obtain Ffuse; then Ffuse is adjusted in channels via a 1x1 convolution to facilitate the subsequent Split operation; through Split, the feature map is sliced into Finj_N4 with 512 channels and Finj_N5 with 1024 channels.

As shown in **Figure 8(e)**, the Inject module adopts a form similar to self-attention; the input consists of the current scale features to be distributed, X_Local, such as B3 and B4, as well as the global features previously generated by the IFM, Y_Global, such as Finj_P3 and Finj_P4. During the fusion process, if the sizes are inconsistent, they can be adjusted using average pooling or bilinear interpolation. Finally, information is further extracted and fused through processing with the RepBlock.

As shown in **Figure 8(f)**, the lightweight adjacent layer fusion (LAF) module is applied to the input X_Local of Inject module. As shown in **Figure 8(g)**, in order to achieve a balance between speed and accuracy, two types of LAFs are designed: low-stage LAF and high-stage LAF, which are used for low-level injection and high-level injection, respectively. Low-level injection merges features from two adjacent layers, while high-level injection merges features from one adjacent layer.



(a) Low-FAM module (b) High-FAM module (c) Low-IFM module (d) High-IFM module

(e) Inject module (f) LAF module (g) Inject-LAF module

Figure 8. Gold-YOLO module structure.

3. Method for detecting and identifying passenger escalator falls

The escalator fall detection system is crucial for reducing response time in emergency situations. Prompt detection and rapid response to falls, as well as performing emergency braking operations on the escalator, are particularly important. This section provides an overview of how our proposed detection algorithm can be effectively applied in real-world scenarios.

The escalator fall detection model proposed in this paper utilizes human keypoint detection to analyze the spatial variation characteristics of human posture and comprehensively determines whether a fall event has occurred through multiple indicators. First, the surveillance video is read and frames are extracted to obtain each frame image. Then, the improved SCGD-Yolo11m-pose is applied to each frame image for detection. The target object detection box is first identified, and skeletal keypoints within the detection box are extracted to obtain relevant feature information of the detection box and keypoints. These detection data are then analyzed to determine significant posture changes that may indicate a fall. This determination method uses multiple features, such as the width-to-height ratio of the detection box, body tilt angle, relative position of the hip centers, and the relative position of the center of gravity [19–21]. These features were selected after multiple repeated experiments to ensure reliable fall detection. **Figure 9** shows the flowchart of the fall detection system.

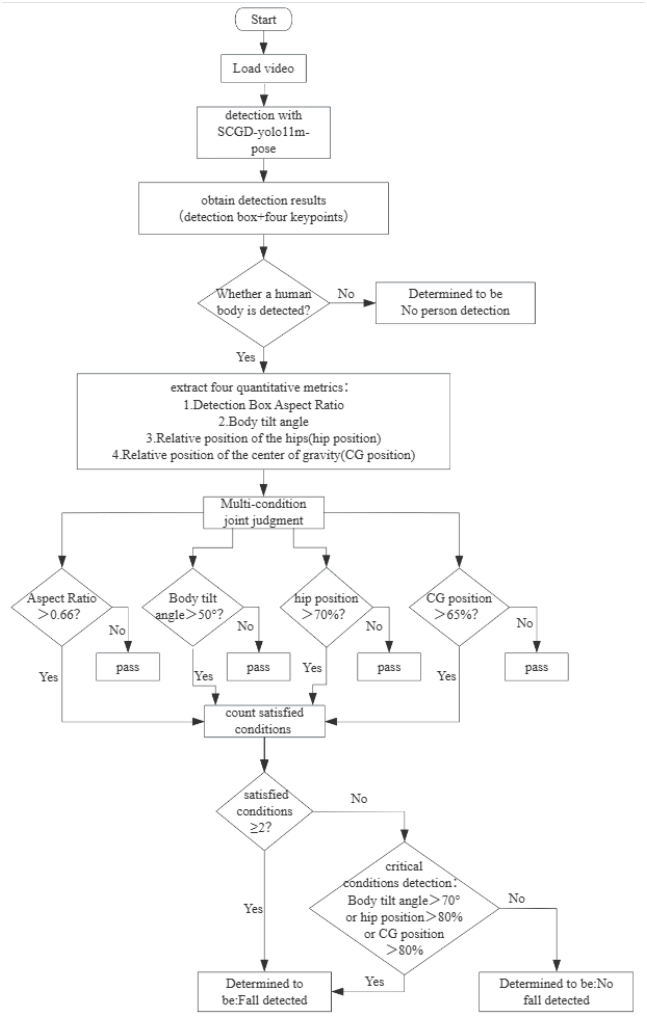


Figure 9. Flowchart of fall detection in surveillance video.

In keypoint detection, we use four custom keypoints, namely the left shoulder, right shoulder, left hip, and right hip, labeled as numbers 0, 1, 2, and 3, respectively. c1 represents the shoulder center, c2 represents the hip center, and c3 represents the center of mass, as shown in **Figure 10**.

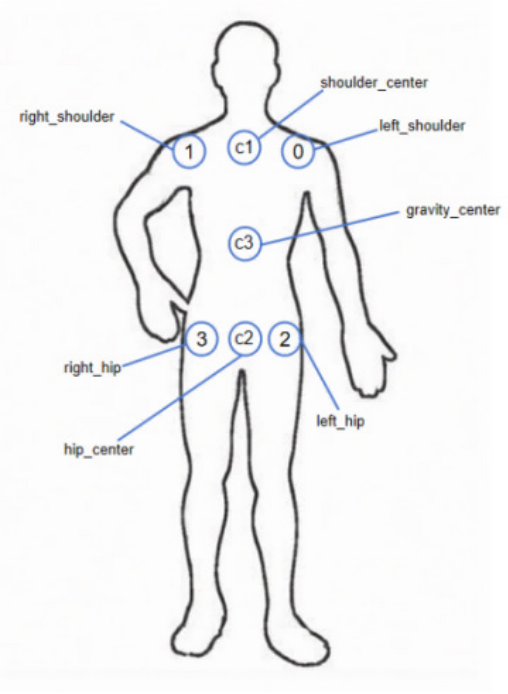


Figure 10. Example diagram of custom human keypoints.

In each frame, the pose keypoints are analyzed and four quantitative metrics are extracted. Each metric is calculated one by one. If these metric features exceed preset thresholds, such as a detection box width-to-height ratio above 0.66 indicating a potential fall, the system accumulates counts after evaluating each quantitative metric. If the fall decision mechanism is satisfied, the frame will be marked as a fall. The thresholds are determined through experiments and tests on the escalator monitoring video dataset to optimize the system's performance in various fall scenarios.

These quantitative features are crucial for distinguishing falls from daily activities. The following are the definitions and preset threshold analyses of these quantitative features as follows:

(1) Detection Box Aspect Ratio: This feature calculates the ratio of the width to the height of the current target detection box. When a person is standing, the width is usually smaller than the height (ratio between 0.2–0.4), whereas when falling, the width is greater than the height (ratio > 0.66);

(2) Body tilt angle: The angle between the line connecting the center of the shoulders to the center of the hips and the vertical direction, reflecting the degree of forward or backward lean of the body. If this angle exceeds 50°, it may be due to bending over or losing balance and leaning forward, indicating a falling posture;

(3) Relative position of the hips: The ratio of the height of hip center to the height of detection frame indicates the relative height of hip in human body. The normal standing relative position is between 0.5 and 0.6. If the relative position height is greater than 0.7, it may indicate that the person has fallen to the ground;

(4) Relative position of the center of gravity: The center of gravity can be expressed by the center positions of the shoulder center and the hip center, that is, the approximate position of the human centroid, and the relative position of the center of gravity is the ratio of the height of the center of gravity to the height of the detection frame. If the relative height of the center of gravity is greater than 0.65, it may indicate a fall has occurred.

The fall decision mechanism are as outlined:

(1) If two or more conditions are met, it is determined as a 'fall';

(2) If any one of the following conditions is met: body tilt angle > 70°, relative hip position > 0.8, or relative center of gravity position > 0.8, it is determined as a 'fall'.

4. Experiment

4.1. Experimental environment and parameter configuration

The experimental environment and parameter configuration of this paper are shown in T**able 1**.

Table 1. Experimental environment and parameter configuration

Environment and parameter	Name	Details
Hardware environment	CPU	Intel(R) Core(TM) i5-8265U 1.60GHz
	GPU	NVIDIA GeForce RTX 3090
Software environment	Operating system	Windows 10
	Programming language	Python 3.10.18
	computing platform	CUDA 12.1
	Deep learning framework	Pytorch 2.1.2
Hyperparameter	Batch size	16
	workers	16
	lr	0.01
	Epochs	400
	optimizer	SGD

4.2. Dataset introduction

The subway escalator dataset for this experiment consists of two parts: one part is composed of images collected from the internet, and the other part is simulated shooting using a Huawei Mate40 phone based on the positions of surveillance cameras in subway stations. The resolution of the images collected from the internet is 720*1280, and the resolution of the data captured by the phone is 2160*3840. After selecting and integrating all the images, they were divided into training, validation, and test sets in an 8:1:1 ratio.

The divided dataset is annotated using the LabelMe tool. First, the detection boxes of people in the images are labeled and named as "person". To reduce the computational load of the model, four key points, namely the left shoulder, right shoulder, left hip, and right hip, are selected for annotation in the human skeletal keypoint annotation, represented by 0, 1, 2, and 3 respectively.

After labeling, generate a COCO-format JSON annotation file, then use a script to convert the JSON file into a YOLO-format txt file. Use 'class' to represent the category of the bounding box, 'x, y, width, height' to represent the normalized center coordinates and size of the bounding box, and each keypoint is represented by (kpx, kpy, v) to indicate the normalized keypoint coordinates and visibility (**Figure 11**).

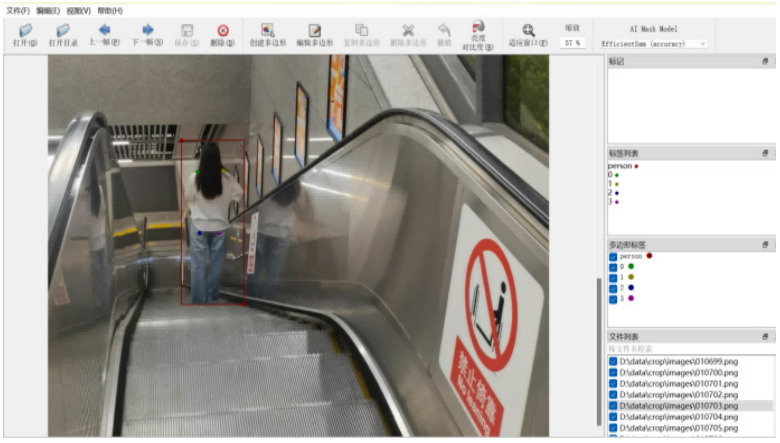


Figure 11. Annotation of customized keypoints.

Subsequently, the training set images were augmented using horizontal flipping, rotation, translation, and brightness variations, as shown in **Figure 12**, ultimately resulting in 4,778 images in the training set, 622 in the validation set, and 716 in the test set.

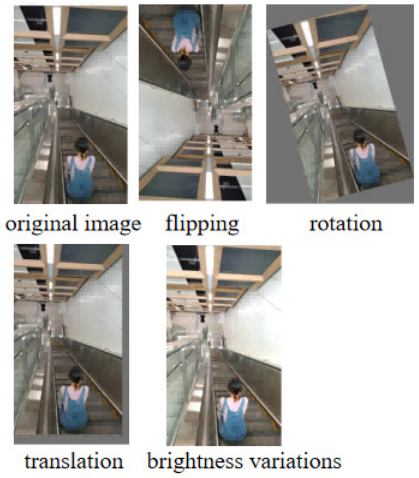


Figure 12. Enhanced image visualization.

4.3. Experimental evaluation metrics

4.3.1. OKS (Object keypoint similarity)

In human keypoint evaluation tasks, the quality of keypoints obtained by a network is not calculated merely through simple Euclidean distance; rather, a certain scale is incorporated to compute the similarity between two points. OKS is used to measure the degree of match between the detected keypoint positions and the ground truth positions. The formula is as follows:

$$OKS = \frac{\sum_i exp(-d_i^2 / 2s^2\sigma_i^2)\delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \tag{5}$$

Here, d_i represents the Euclidean distance between the currently detected keypoint and the keypoint in the ground truth (GT); s denotes the scale factor of the pedestrian in the GT, and $s^2$is the area of the pedestrian detection bounding box, that is, the product of the width and height of the bbox; σ_i reflects the standard error of the i-th predicted keypoint relative to the GT position; v_i is the keypoint visibility flag, where $v_i > 0$ indicates the keypoint is visible, otherwise it is not visible. δ (*) indicates that if the condition * is met, then δ (*) = 1, otherwise δ (*) = 0, meaning that only the points annotated in the GT are considered. For each keypoint, the range of OKS is between 0 and 1. The OKS value is equivalent to the IOU in object detection.

4.3.2. Precision and recall

By calculating the OKS metric, TP, FP, and FN are distinguished according to thresholds, Precision and Recall are calculated, and then Average Precision (AP) is computed. Precision refers to the proportion of instances predicted as positive by the model that are truly positive. Recall refers to the proportion of all true positive instances that are correctly predicted as positive by the model. The calculation formulas are as follows:

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

Among them, TP represents the number of positive samples correctly predicted, FP represents the number of negative samples incorrectly predicted, and FN represents the number of positive samples incorrectly predicted.

4.3.3. AP (Average precision)

AP is the average precision of keypoints at a threshold x. By defining different thresholds, different AP values can be obtained. For example, AP50 represents the AP at OKS = 0.5, and AP50:95 is the mean of average precision across 10 thresholds from OKS 0.5 to 0.95.

4.4. Comparison of experimental results

Under the condition that the experimental environment, parameter settings and datasets are kept consistent, this algorithm is compared with the commonly used Yolov8-pose, Yolo11-pose, as well as the latest Yolo series models Yolov12-pose and Yolo26-pose, using metrics such as AP50, AP50:95, Parameters and FLOPs for evaluation, as shown in **Table 2**.

Table 2. Comparison results of keypoints detection performances on self-built dataset

Model	AP50/%	AP50-90 (%)	Parameters/M	FLOPS (G)
Yolov8m-pose	67.8	62.7	26.40	80.8
Yolo11m-pose	67.6	62.3	20.88	71.3
Yolo12m-pose	66.5	60.5	20.96	70.8
Yolo26m-pose	63.1	58.2	21.20	71.5
SCGD-Yolo11m-pose	69.9	64.8	22.94	87.6

According to the data, taking into account the balance between AP and parameters, this paper proposes improvements to the SCGD-Yolo11m-pose based on the Yolo11m-pose model, effectively enhancing detection performance. Compared with the original Yolo11m-pose, AP50 increased by 2.3%, and AP50:95 increased by 2.5%. Meanwhile, the parameters and computational complexity of the improved model increased slightly, by 2.06M and 16.3G, respectively.

4.5. Ablation experiment results

In order to demonstrate that the introduction and improvement of various modules in the SCGD-Yolo11m-pose network are effective, this section conducts an ablation experiment on a self-built experimental dataset. This experiment still uses AP50, AP50:95,Parameters and FLOPs as evaluation metrics, with the experimental results shown in **Table 3**, where it presents detection results under seven different network structures. These seven structures are: the feature extraction module of Yolo11m-pose with the C2DA structure, the downsampling structure with the SRFD structure, the Neck part with the Gold-Yolo structure, SRFD + C2DA, SRFD + Gold-Yolo, C2DA + Gold-Yolo, and the network structure proposed in this paper, SCGD-Yolo11m-pose.

Table 3. Ablation experiment results of improved network

Model	SRFD	C2DA	Gold-Yolo	AP50/%	AP50:95/%	Parameters/M	FLOPs/G
Yolo11m-pose	√			67.9	62.6	20.95	71.4
		√		67.6	63.6	19.13	75.5
			√	69.5	64.7	23.99	82.6
	√	√		67.8	62.1	19.84	76.3
	√		√	69.8	64.6	22.88	87.5
		√	√	68.0	62.7	24.05	82.7
	√	√	√	69.9	64.8	22.94	87.6

The experimental results indicate that improvements in each module lead to enhancements compared to the original Yolo11m-pose network. Specifically, in AP50, SRFD, Gold-Yolo, SRFD + C2DA, SRFD + Gold-Yolo, C2DA + Gold-Yolo, and SRFD + C2DA + Gold-Yolo architectures increased by 0.3%, 1.9%, 0.2%, 2.2%, 0.5%, and 2.3%, respectively; in AP50:95, SRFD, C2DA, Gold-Yolo, SRFD + Gold-Yolo, C2DA + Gold-Yolo, and SRFD + C2DA + Gold-Yolo architectures increased by 0.3%, 1.3%, 2.4%, 2.3%, 0.4%, and 2.5%, respectively. Using SRFD as the backbone network can effectively reduce the model parameter, achieving a 4.6% reduction.

4.6. Visualization result analysis

From the experimental results in **Figure 13**, it can be seen that the improved SCGD-Yolo11m-pose algorithm in this paper can accurately detect the falling accident in the displayed images, and the positioning and framing of the fallen passengers are more accurate. The other four algorithms have missed detection and false detection. **Figure 13(e)–(f)** shows the effect on the public fall detection dataset. The improved algorithm realizes the detection of small targets, and can detect four key position coordinates of human body more accurately, so as to make more accurate discrimination and reduce the false detection rate.

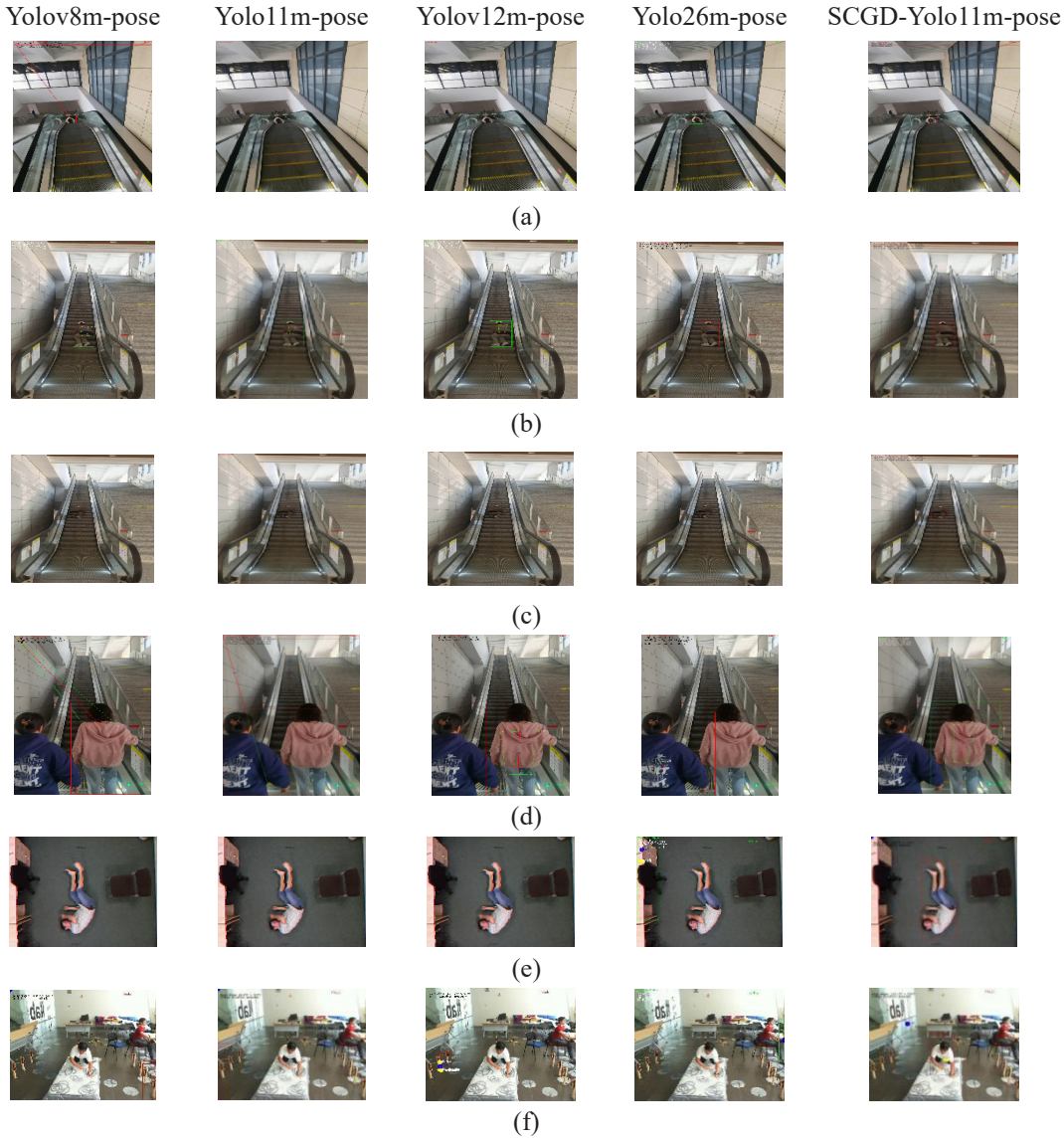


Figure 13. Visualization of passenger escalator fall detection results.

5. Conclusion

This article focuses on optimizing the detection of passengers falling on escalators inside subway stations. By improving the Yolo11m-pose high-precision detection algorithm, an enhanced model based on pose estimation, SCGD-Yolo11m-pose, is proposed. The main contributions of the model are as follows:

(1) SRFD (Shallow Robust Feature Downsampling) and DRFD (Deep Robust Feature Downsampling) modules are introduced during the backbone network downsampling stage. Through multi-branch feature fusion and the Cut Slice operation, key defect information is preserved while downsampling, significantly enhancing the ability to capture features across scale variations;

(2) The self-attention mechanism in the original C2PSA module was replaced with a deformable attention mechanism, constructing the C2DA module. This allows the model to dynamically adjust its attention focus when processing images, enhancing attention to target regions with minimal additional parameters, thereby enabling more accurate target recognition and reducing interference from irrelevant

backgrounds;

(3) Inspired by Huawei Noah's Ark Lab's development of Gold-Yolo, the GD information aggregation-distribution mechanism was innovatively integrated with the Neck part of Yolo11, utilizing its efficient global information fusion mechanism to enhance the model's detection accuracy in complex backgrounds.

Experimental results on the self-built subway escalator fall dataset demonstrate that the improved model is effective and practical in ensuring the safety monitoring of subway passengers. It not only reduces the workload of staff but also significantly enhances the efficiency of safety monitoring. In the future, the improved lightweight algorithm will be further optimized to facilitate the deployment of the model on front-end embedded devices, enabling local monitoring data collection and escalator fall detection, covering more emergency situations and fall detection in more complex environments.

Disclosure statement

The author declares no conflict of interest.

References

[1] Alam E, Sufian A, Dutta P, et al., 2022, Vision-Based Human Fall Detection Systems Using Deep Learning: A Review. Computers in Biology and Medicine, 2022(146): 105626.

[2] Zhao W, 2024, Research On 2D Human Pose Estimation, thesis, Yanshan University.

[3] Li J, Wang D, Zhang S, 2024, Deep-Learning-Based 2D Human Pose Estimation: Present and Future. Chinese Journal of Computers, 47(1): 231–250.

[4] Zhao D, Song T, Gao J, et al., 2024, YOLO-Fall: A Novel Convolutional Neural Network Model for Fall Detection in Open Spaces. IEEE Access, 2024(12): 26137–26149.

[5] Yu C, 2022, Pedestrian Safety Detection System on Escalator Based on Human Pose Estimation, thesis, Southeast University.

[6] Yang L, 2023, Research on Escalator Pedestrian Fall Detection Based on Deep Learning, thesis, Xi'an University of Science and Technology.

[7] Liang B, 2023, Real-Time Detection Algorithm of Escalator Figure Falling based on Video, thesis, Shanxi University.

[8] Wang Y, Wan H, Huang K, et al., 2024, Real-Time Detection of Abnormal Behavior of Escalator Passengers based on YOLOv5s. Laser & Optoelectronics Progress, 61(8): 211–218.

[9] Yang X, Li J, Chen L, et al., 2024, An Abnormal Behavior Recognition Method of Escalator Passengers based on Human Skeletons. Journal of Safety and Environment, 24(2): 636–643.

[10] Hou Y, Hu Y, Zhao R, et al., 2025, Escalator Passenger Safety Detection YOLO_BFROI Algorithm Based on Region of Interest. Computer Engineering and Applications, 61(6): 84–95.

[11] Khanam R, Hussain M, 2024, YOLOv11: An Overview of the Key Architectural Enhancements.

[12] Liu S, Qi L, Qin H, et al., 2018, Path Aggregation Network for Instance Segmentation. IEEE, 2018.

[13] Lu W, Chen S, Tang J, et al., 2023, A Robust Feature Downsampling Module for Remote-Sensing Visual Tasks. IEEE Transactions on Geoscience and Remote Sensing, 2023(61): 1–12.

[14] Hendrycks D, Gimpel K, 2016, Gaussian Error Linear Units (GELUs), arXiv:1606.08415

[15] Xia Z, Pan X, Song S, et al., 2022, Vision Transformer with Deformable Attention, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4794–4803.

[16] Yang P, 2025, Research on Road Traffic Object Detection Algorithms Based on Deep Learning, thesis, Guizhou University.

[17] Wang C, He W, Nie Y, et al., 2023, Gold-YOLO: Efficient Object Detector via Gather-and-Distribute Mechanism. Advances in Neural Information Processing Systems, 2023(36): 51094–51112.

[18] Xiao Y, Wang C, Qu Y, et al., 2025, An Algorithm of Detecting GSS-YOLO Object Geared to Surface Defects of Strip Materials. Journal of Air Force Engineering University, 26(5): 31–41.

[19] Chen W, Jiang Z, Guo H, et al., 2020, Fall Detection Based on Key Points of Human-Skeleton Using OpenPose. Symmetry, 12(5):744.

[20] Mali V, Jaiswal S, 2025, Pose-Based Fall Detection System: Efficient Monitoring on Standard CPUs, arXiv.

[21] Sun H, 2025, Research on Surface Defect Detection of Differential Housings Based on Deep Learning, thesis, Hubei University of Automotive Technology.