Intelligent Traffic Pedestrian Detection by Integrating YOLOv4 and Improved LSTM

Jianfeng Chen
School of Computer and Information Technology, Anhui Vocational and Technical College, Hefei 230000, China
E-mail: 18815606298@163.com

Keywords: YOLOv4, LSTM, transportation, deep learning, CBAM

Received: May 17, 2024

As an important part of modern urban traffic management, intelligent transportation system aims to improve road safety, optimize traffic flow and reduce traffic congestion. In this study, an intelligent traffic pedestrian detection model combining YOLOv4 and improved time cycle neural network is proposed. The model is based on YOLOv4-tiny algorithm to detect human key points of pedestrians, and introduces convolutional attention module to improve the detection accuracy of the model. After that, a pedestrian intention recognition model is established by using long term memory network and Openpose technology. Based on the key point information detected by YOLOv4-tiny, the intention recognition model determines the pedestrian crossing motive. The experimental results show that the detection accuracy of pedestrian intention detection model combined with pedestrian position and human key point information is 77.8%, and the frame rate is 18FPS. When the research design model is used to detect the key points and position information of human body, its average confidence is about 90%, showing high stability and accuracy. The experimental results prove that the intelligent traffic pedestrian detection system designed in this study can meet the real-time and accuracy requirements of vehicle-mounted systems for pedestrian detection, and can accurately capture pedestrian intention.

Povzetek: Študija uvaja inteligenten sistem za zaznavanje pešcev v prometu, ki združuje YOLOv4 algoritem in izboljšani LSTM.

1 Introduction

As an important component of modern urban traffic management, intelligent transportation systems aim to improve road safety and reduce traffic congestion [1]. In this field, pedestrian detection technology is one of the core issues, which is of great significance for preventing traffic accidents and improving road use efficiency. As deep learning technologies develops, pedestrian detection methods using these technologies have become a research hotspot. You Only Look Once version 4 (YOLOv4) is an advanced real-time object detection framework widely used in various scenarios due to its high efficiency and accuracy. However, in complex traffic environments such as changing lighting conditions, occlusion, and pedestrians from different angles, YOLOv4 still faces challenges in pedestrian detection [2, 3]. In addition, traditional pedestrian detection methods often overlook the analysis of time series data, which is particularly important in dynamic traffic environments. Long Short-Term Memory (LSTM) is effective for processing time series data. By improving LSTM, it is possible to better capture the dynamic features of pedestrians in time series, thereby improving the accuracy and robustness of detection. However, effectively integrating LSTM with YOLOv4 to adapt to pedestrian detection in intelligent transportation systems is still a research area worth exploring. In view of this, the study proposes an intelligent traffic pedestrian detection model that integrates YOLOv4 and improved LSTM. This model aims to utilize the efficient detection capability of YOLOv4 and improve the time series analysis capability of LSTM to enhance pedestrian detection performance in complex traffic environments. The research will focus on the design, optimization, and experimental verification of the model, aiming to address the limitations of existing pedestrian detection methods in intelligent transportation systems. The research aims to provide a more accurate and robust solution for pedestrian detection in intelligent transportation systems, thereby improving road safety and traffic efficiency.

The research is divided into four parts. The first part introduces the intelligent traffic pedestrian detection system, LSTM and YOLOv4. The second part proposes pedestrian detection key points and intention recognition models based on YOLOv4 and LSTM. The third part is to test and analyze model performance. The fourth part summarizes and discusses the above content.

2 Related works

Pedestrian detection is an important task of autonomous driving, and many scholars conducted research on it. Wang et al. designed a three ResNet block using a central network detection model. Three ResNet block was

proposed for network semantic information extraction. Experimental results showed an excellent accuracy and detection speed [4]. Tang et al. proposed a novel fine-tuning model for unsupervised pedestrian detection that did not require the use of source and target data. It applied multi-expert learning algorithms and integrated bounding boxes to improve accuracy. The experimental results showed that this method effectively improved detection accuracy under unsupervised settings [5]. Panigrahi and Raju introduced a novel feature extraction method for pedestrian detection, outperforming existing pre-trained CNNs with higher accuracy [6]. The YOLO series has been extensively studied by many scholars. Yan et al. used YOLOv5 to address low accuracy and slow speed in traditional coal gangue identification. Results showed that detection accuracy using the YOLOv5.1 model reached 98.34% [7]. Jia et al. presented a YOLOv5-based approach for motorcycle driver helmet detection. By incorporating soft-NMS forYOLOv5 detector fusion, they achieved impressive results in the experiment, including 97.7% mAP, 92.7% F1-score, and 63 frames per second (FPS). These outcomes surpassed those of other methods [8].

The LSTM algorithm has been applied in many fields. Nemani et al. studied the deep learning in predicting bearings' remaining life, determined the bearing fault threshold based on ISO standards, and proposed a two-stage LSTM model for extracting fault feature signals of bearings. Gaussian layers were embedded in the LSTM model for parameter optimization. Results showed a good accuracy in predicting bearing life [9]. Chen et al. proposed a novel algorithm by integrating spatio-temporal attention mechanism with ConvLSTM for fake face detection improvement. The excellent performance results showed that this algorithm performed better than existing algorithms [10]. To address the challenges of transparency and interpretability in machine learning algorithms, Kaadoud et al. understood the results from simple clues and rules, and used internal state clustering algorithms in LSTM models to study the hidden states of spatial knowledge extraction. They established and validated automatic sequences extracted on the basic syntax. The experimental results showed that the sequences extracted on the original syntax had high recognition rates [11].

In conclusion, while numerous algorithms can optimize dataset and feature extraction to enhance pedestrian detection system accuracy, there remains a scarcity of systems that combine high-speed computing power with lightweight embedded device characteristics. These two have strong potential application value in vehicle detection.

The summary of the relevant work of the existing research is shown in Table 1.

Table 1: Summary of relevant work

Document serial number	Method	Key features	Dataset used	Accuracy	FPS	Remarks
[4]	Central network detection model	Pedestrian key point information	Caltech and ETH	89.74%	18FPS	-
[5]	Pedestrian detection based on unsupervised adaptive framework	Pedestrian position point	BIWI walking pedestrian's dataset	92.47%	22FPS	-
[6]	SVM and multi-layer feature fusion	Shape, color, and texture features	DukeMTMC-reID	93.15%	17FPS	-
[7]	Improved YOLOv5	Spectral feature	HDA	98.34%	-	-
[8]	Improved YOLOv5	Driver helmet feature	TWHD	97.7%	63FPS	-
[9]	Two-stage LSTM model	Fault signature signal	FEMTO bearing data set	88.47%	22FPS	-
[10]	Based on temporal attention and ConvLSTM	Spatio-temporal characteristics	LFW	71.58%	-	-
[11]	Clustering algorithm and LSTM	Spatio-temporal series feature	ETH	75.93%	11FPS	-

According to Table 1, although the above studies achieved remarkable results in pedestrian detection, bearing life prediction and prosthetic face detection, there were still some challenges and areas to be improved. Especially in real-time and embedded system applications, most methods, while highly accurate, often sacrifice computing speed or are difficult to deploy on lightweight devices. Therefore, the future research should pay more attention to how to improve the real-time and embeddability of the system while ensuring the accuracy.

3 Intelligent traffic pedestrian detection algorithm combining YOLOv4 and Improved LSTM

In intelligent driving systems, pedestrian detection plays a crucial role, and its basic function is pedestrians presence identification and their specific positions determination, laying the foundation for further pedestrian intention analysis. Aiming at the challenges of pedestrian detection in autonomous driving environments, such as occlusion, lighting changes, scale differences, and noise, a fusion method of pedestrian detection and human key point detection is proposed to improve detection accuracy and satisfy vehicle systems. On the basis of pedestrian detection, research also proposed a traffic pedestrian crossing intention recognition based on LSTM, providing a more solid guarantee for the safety of intelligent driving.

3.1 Design of pedestrian and key point detection based on YOLOv4

In order to deeply analyze the movement trend and crossing intention of pedestrians, the research focus has been extended to human key point detection, aiming to improve pedestrian intention recognition accuracy by analyzing the changes in human posture. The study adopted an improved lightweight YOLOv4-tiny algorithm suitable for vehicle use as the baseline network for the model. The YOLOv4-tiny algorithm is an efficient lightweight object detection model that is simplified based on the YOLOv4 algorithm, aiming to reduce computational resource requirements while maintaining detection performance as much as possible. This algorithm inherits the design concept of the YOLO series model, which is "You Only Look Once", and achieves target detection and classification through forward propagation of a single convolutional neural network. YOLOv4-tiny optimizes the speed and size of the model by simplifying the network structure and parameters, making it more suitable for deployment in resource constrained environments, such as mobile devices or in vehicle systems [12, 13]. YOLOv4-tiny is structured which is shown in Figure 1.

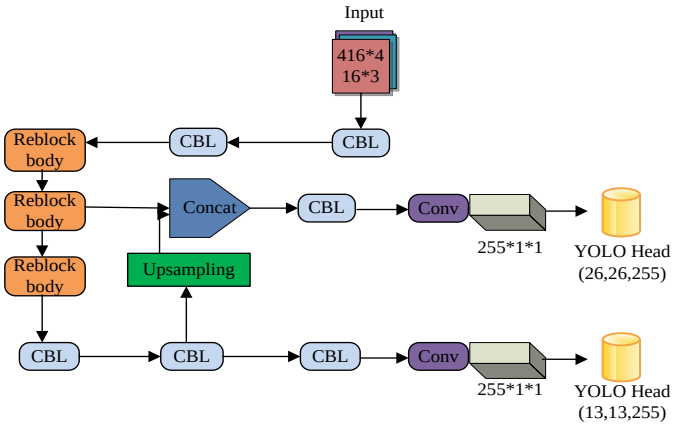


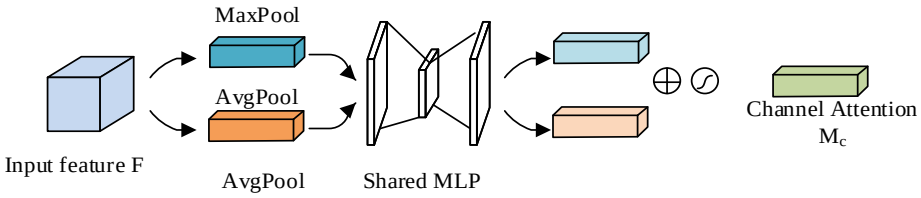
Figure 1: YOLOv4-tiny structure

As shown in Figure 1, the structure of YOLOv4-tiny algorithm is mainly composed of input layer, backbone network, feature pyramid network, detection header and output layer. Firstly, the input layer is responsible for processing the input image and transforming suitably for network processing. In this section, images are uniformly adjusted to a fixed size and normalized to prepare for subsequent feature extraction. Secondly, the backbone network is the core to extract image features. YOLOv4-tiny uses a reduced version of CSPDarknet53 as the feature extractor in this section, learning the abstract representation of images through deep convolution and residual connections. CSPDarknet53 uses LeakyReLU as the activation function, as calculated in equation (1).

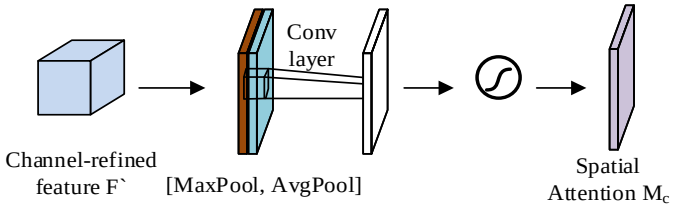
$$y_i = \begin{cases} x_i & x_i \geq 0 \\ \dfrac{x_i}{a_i} & x_i < 0 \end{cases} \tag{1}$$

In order to adapt to lightweight design, the network's depth and width are reduced to lessen computational

burden. Subsequently, Feature Pyramid Networks (FPN) and Path Aggregation Networks (PANet) are used to construct advanced semantic features and enhance the fusion of features at different scales. This section effectively integrates feature maps of different resolutions through upsampling and downsampling strategies, enhancing small targets detection. The detection head is the decision-making part of the algorithm, which uses a series of convolutional layers to predict bounding boxes, target categories, and confidence. YOLOv4-tiny simplifies the design of the detection head, reduces convolutional layers, and uses an anchor box mechanism for prediction accuracy improvement. Finally, the output layer converts the predicted results of the detection head into the final detection output. This information is then used in Non-maximum Suppression (NMS) to remove overlapping detections and preserve the best prediction results. However, the YOLOv4-tiny algorithm is limited by its embedded design, simplified feature extraction operations, and low detection accuracy. To improve its detection accuracy, research is being conducted to integrate the Convolutional Block Attention Module (CBAM) to improve the feature weight allocation ability of YOLOv4-tiny on feature maps, in order to solve small target missed detection due to local occlusion and natural background confusion. Figure 2 shows the CBAM structure.



(a) Channel Attention Module



(b) Spatial Attention Module

Figure 2: CBAM structure

In Figure 2, CBAM combines Spatial Attention (SA) and Channel Attention (CA) mechanisms to improve the performance of convolutional neural networks. CBAM can adaptively assign different attention weights to different network parts, thereby helping the network better capture and utilize important information of input features. Through this approach, CBAM helps improve CNN in image classification and object detection. CBAM's front module is CA, and this part is calculated in equation (2).

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \quad (2)$$

As shown in equation (2), F represents the input feature map; MLP represents the shared fully connected layer; $AvgPool$ and $MaxPool$ represent average and maximum pooling, respectively. The CBAM front-end processes the input feature layer and performs global $AvgPool$ and $MaxPool$. These processing results are merged after being processed by MLP, and then the Sigmoid function calculates channel weights, which are then multiplied with the input features.

$$M_S(F) = \sigma(f^{7\times7}([AvgPool(F); MaxPool(F)])) \quad (3)$$

In equation (3), $f^{7\times7}$ represents the convolution kernel size. The CA superimposes the maximum and average values of the features passed through the front on the feature channel, and then performs convolution operations. The sigmoid function is used to calculate the weights and multiply them with the input layer features. The study chose the Openpose algorithm to extract human key points from images for pedestrian pose estimation. Openpose was proposed by researchers from Carnegie Mellon University in 2017. This algorithm detects human keypoints in images through convolutional neural networks and uses Part Associated Fields (PAFs) to identify spatial relationships of body parts, effectively distinguishing multiple poses. The Openpose algorithm can detect joint nodes in human posture, and its key point mapping table is shown in Table 2.

Table 2: Key point mapping

Serial number	Name of key points	Serial number	Name of key points
0	Noses	9	Left Knee
1	Neck	10	Left Ankle
2	Left Shoulder	11	Right Buttock
4	Left Elbow	12	Right Knee
5	Left Wrist	13	Right Ankle
6	Right Shoulder	14	Left Eye
7	Right Elbow	15	Right Eye
8	Right Wrist	16	Left Ear

Openpose uses a bottom-up approach to estimate real-time pose of multiple people in the image, and effectively identifies and associates key points of various parts of the human body through Part Affinity Fields (PAFs) technology. This method avoids failures caused by missed detection boxes and improves detection speed and robustness. The algorithm uses a greedy strategy to globally optimize key points, ensuring the accuracy of the results even with a slight increase in computational complexity, without being significantly affected by the increase in people in the image. Openpose network is structured in Figure 3.

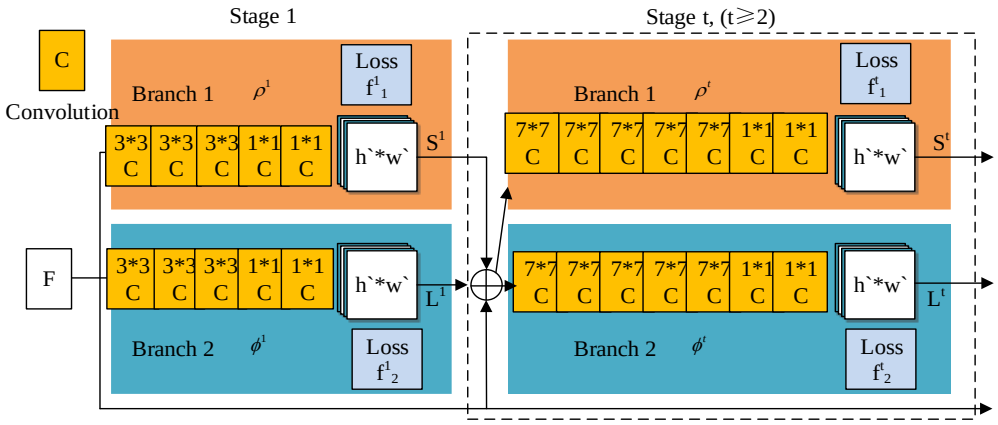


Figure 3: Openpose network structure

As shown in Figure 3, Openpose network is mainly composed of convolutional neural network and PAFs to achieve feature extraction and connection direction prediction respectively. These two parts alternate in a multi-stage convolutional network, gradually refining the prediction results, ensuring accurate positioning of key points and correct association of various body parts. Firstly, the input image is passed through Openpose to generate a feature map F. Then, the feature map is input into two branch convolutional networks to generate a detection confidence map $S^1 = \rho^1(F)$, as well as a partial affinity field $L^1 = \phi^1(F)$. The calculation expression for the detection confidence map is shown in equation (4).

$$S^t = \rho^t(F, S^{t-1}, L^{t-1}), \ \forall t \geq 2 \tag{4}$$

In equation (4), t represents the iteration stage or number of iterations, and some affinity field calculations can be found in equation (5).

$$L^t = \phi^t(F, S^{t-1}, L^{t-1}), \ \forall t \geq 2 \tag{5}$$

To gradually optimize the network prediction accuracy for body key points and their corresponding PAFs, at the end of each stage, the network refines the body part recognition of the first branch and PAFs map of the second branch by applying two specialized loss functions. These two loss functions are optimized for the outputs of both branches. The first branch loss function is shown in equation (6).

$$f_s^t = \sum_{j=2}^{J} \sum_{P} W(p) \bullet \left\| S_j^t(p) - S_j^*(p) \right\|_2^2 \tag{6}$$

In equation (6), j and J represent key points; $S_j^*(p)$ represents their true confidence map; $S_j^t(p)$ is their predicted confidence map; $W(p)$ represents the binary mask at the image p, and the second branch loss function is shown in equation (7).

$$f_L^t = \sum_{c=2}^{C} \sum_{P} W(p) \bullet \left\| L_j^t(p) - L_j^*(p) \right\|_2^2 \quad (7)$$

In equation (7), $L_j^*(p)$ represents the actual PAFs; $L_j^t(p)$ represents the predicted PAFs; c and C represent the types of connection site. The overall loss function is shown in equation (8).

$$f = \sum_{t=1}^{T} \left(f_S^t + f_L^t \right) \quad (8)$$

In equation (8), pixel's value on the confidence map represents the probability value that the point is a key point. The true value modeling of the confidence map is shown in equation (9).

$$S_{j,k}^* = \exp(-\frac{\left\| p - x_{j,k} \right\|_2^2}{\sigma^2}) \quad (9)$$

In equation (9), $S_{j,k}^*$ represents the confidence level of j corresponding to pedestrian k; $x_{j,k}$ represents the true coordinates of the key point, and the maximum value is used in the calculation of $S_j^*(p)$. The calculation expression is shown in equation (10).

$$S_j^*(p) = \max_k S_{j,k}^*(p) \quad (10)$$

3.2 Intelligent traffic pedestrian crossing intention recognition based on LSTM

The study adopts an improved LSTM network combined with Openpose technology, based on pedestrian position and key point information, to train a model to capture the intrinsic patterns of pedestrian intentions, aiming to achieve more accurate and reliable recognition of pedestrian crossing intentions. LSTM is an advanced recursive neural network that can learn and predict long-term dependencies in time series data. In 1997, it was first proposed by Hochreiter and Schmidhuber that LSTM effectively avoids the gradient vanishing problem of traditional RNNs by cleverly gating the information flow in long sequence data. The core of LSTM is unit state, which, in conjunction with input, forget, and output gates, can maintain and transmit critical state information between time steps. These gates control the storage, updating, and output of information, endowing LSTM networks with the ability of long-term and short-term memory, enabling them to perform well in complex sequence learning tasks such as language modeling and time series analysis. The LSTM structural units are shown in Figure 4.

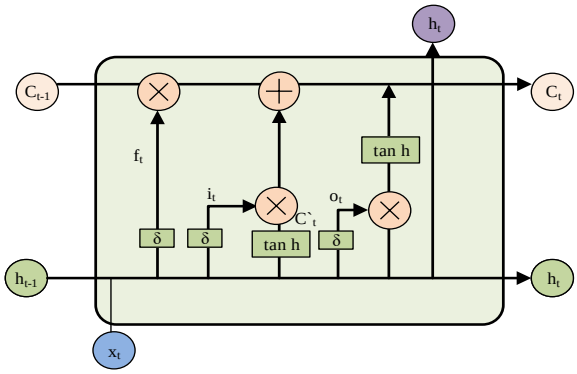


Figure 4: LSTM structural units

As shown in Figure 4, the LSTM structural unit consists of a forgetting gate, an input gate, an output gate, and a memory unit. Forgetting gate f_t determines the forgotten data using the current input x_t and the previous state output h_{t-1}. The input gate i_t also determines the information entering the neural unit through the valve based on x_t and h_{t-1}. Memory unit C_t is simultaneously affected by the previous memory unit C_{t-1}, input gate, and forgetting gate. The calculation expression for the input gate is shown in equation (11).

$$i_t = \sigma(W_i \times [h_{t-1}, x_t] + b_i) \quad (11)$$

As shown in equation (11), W_i, W_f, W_o, W_t, U_t, U_o, U_f U_i are weight matrices, and b_i, b_f, b_c and b_o are deviation vectors. The expression for updating cell status is shown in equation (12).

$$C_t = f_t * C_{t-1} + i_t * C`_t \quad (12)$$

In equation (12), variable $C`_t$ is the value output by the activation function. The calculation expression for the output gate is shown in equation (13).

$$o_t = \sigma(W_o | h_{t-1}, x_t | + b_o) \quad (13)$$

In equation (13), σ is the sigmoid activation function, and h_t is calculated in equation (14).

$$h_t = o_t * \tanh(C_t) \quad (14)$$

Equation (14) calculates the new value of the output gate and the final output results tanh and σ, both of which are activation functions, by memorizing the new state of the cells. This can linearize the linear output results in the neural network and enable it to simulate nonlinear functions. In order to address the issue of reduced influence of early input information caused by the extension of time series in LSTM models for pedestrian crossing intention recognition tasks, a

multi-head attention mechanism is introduced in the study. This mechanism enhances the capture and utilization of early important signals by parallelizing information processing and endowing the model with the ability to consider the entire time series information more evenly at each time step, ensuring that the influence of key information in model prediction is maintained. Through this method, the model can more accurately identify the pedestrian's intention to cross the street, improving the overall recognition performance. The structure of the LSTM model optimized by combining multi-head attention mechanism is shown in Figure 5.

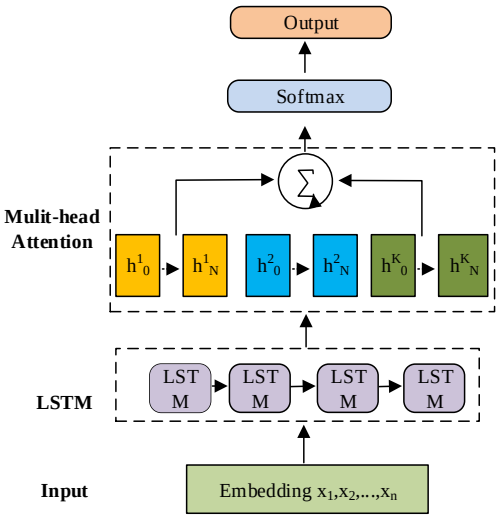


Figure 5: Optimized LSTM model structure based on multi-head attention mechanism

As shown in Figure 5, pedestrian position and keypoint data are first transformed into vector sequences through the embedding layer, and then processed through the LSTM layer to obtain high-level features. These features are transformed into matrices, input into a multi-head attention layer, and processed through a Softmax classifier to output the final result.

4 Performance testing of intelligent traffic pedestrian detection algorithm integrating YOLOv4 and improved LSTM

A mixed dataset containing 2000 pedestrian images was constructed, combining VOC2007 and JAAD pedestrian datasets. In this dataset, 80% images were trained, 10% for testing, and 10% for validation to assess model's generalization ability and accuracy. The experiment used PyTorch as a deep learning framework. During the training process, all images were uniformly adjusted to a size of 416×416 pixels, and input batch size to 16. The training strategy included using label smoothing techniques to reduce overfitting. The training process was divided into two stages: the first 50 epochs used preheating training method, with a 0.001 learning rate for parameters optimization. Subsequently, using cosine annealing strategy, the learning rate was gradually reduced to 0.0001 to refine the learning process. The experiment was done on a high-performance computing cluster equipped with NVIDIA Tesla V100 Gpus with 32GB of RAM and eight CPU cores per node. To make full use of computing resources, the experiment adopted a distributed training strategy, and divided the dataset into multiple batches through PyTorch's data parallel module and processed them in parallel on multiple Gpus. In terms of the segmentation of training and test datasets, the research first randomly shuffled the original data set, and then divided the data into training sets, test sets and verification sets according to the ratio of 80%, 10% and 10%. After training, the loss function curve was obtained to locate the loss curve. The classification and confidence loss curves are shown in Figure 6.

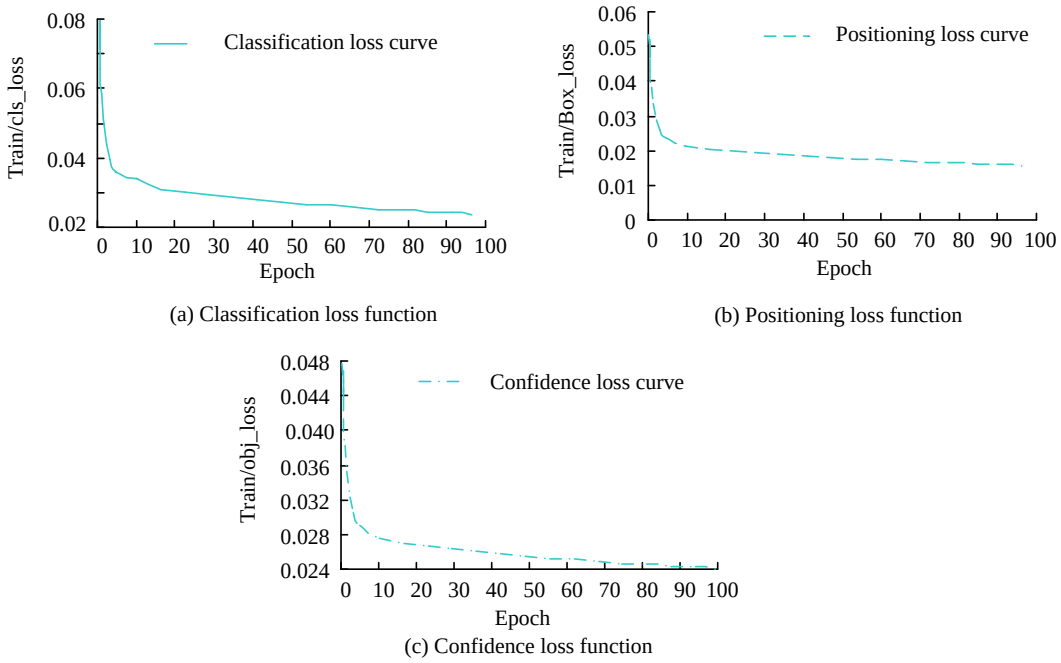


Figure 6: Loss curve

Figure 6 illustrates the training process of the model, which exhibited a typical pattern with a consistent downward trend. The loss experienced rapid decline in the initial 20 epochs and then stabilized around the 100th epoch. The final localization loss converged at approximately 0.017, classification loss remained steady at around 0.00117, and confidence loss stabilized at around 0.028. Comparative experiments were conducted using four models: Fast-RCNN, YOLOv4, YOLOv3 tiny, and YOLOv4-tiny. All models were compared under the same hardware conditions and the same dataset was used. The experimental results were evaluated using average precision (AP) and FPS as performance metrics. AP was used to measure the accuracy of model detection, while FPS reflected the speed at which the model processes images. Through these indicators, the performance of different models in pedestrian detection tasks was comprehensively evaluated. The comparative experimental results are shown in Figure 7.

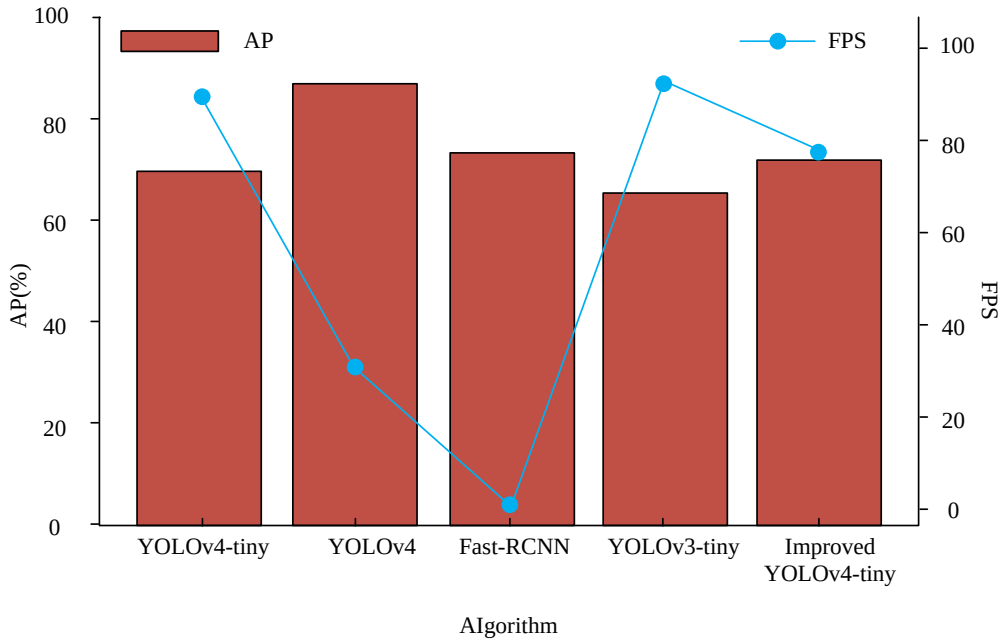


Figure 7: Comparison of experimental results

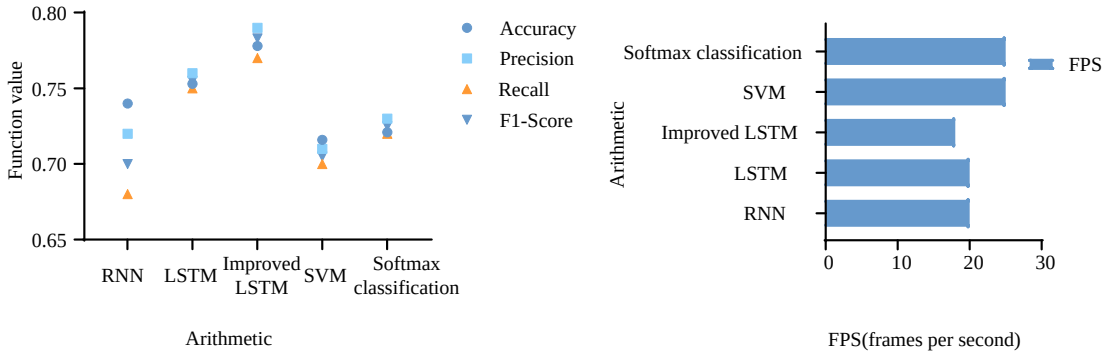
The study compared YOLOv4, YOLOv4-tiny, and improved models on the test dataset. Although YOLOv4 had the highest AP value of 87.09%, its FPS was significantly lower than other models and was not suitable for in vehicle systems. In contrast, the enhanced YOLOv4-tiny model demonstrated a 2.3% increase in AP, despite a slight decrease in FPS. However, it successfully met the real-time detection requirements and demonstrated its compatibility with embedded systems in vehicles. In Figure 8, a comparison was made between YOLOv4-tiny and the enhanced algorithm in terms of actual detection performance.

(a) YOLOv4-tiny algorithm detection results

(b) Detection result graph of the improved YOLOv4-tiny algorithm

Figure 8: Comparison of experimental results

As shown in Figure 8, in the detection result graph of improved YOLOv4-tiny algorithm, it successfully detected nearby pedestrians and provided high confidence, but there were still challenges in detecting distant pedestrians. The improved model exhibited better detection ability under occlusion conditions, detecting pedestrians partially occluded in the distance in the image. However, there were still situations where the improved algorithm could not detect pedestrians who were too far away or not fully entered the screen. This indicated that although the improved algorithm outperformed the original version in specific situations, further improvement in detection robustness was still needed in certain extreme cases. The study integrated the JAAD and PIE datasets as the test set, with a total of 1257 videos labeled as pedestrian crossing or not crossing. According to the video content, 80% was trained, 10% was validated, and 10% was tested. The video format was adjusted to 30 FPS, with a size of 500×375. To improve efficiency, 20 frames of pedestrian behavior sequences were cropped and extracted every 60 frames from the behavior video. Pedestrian information was extracted using YOLOv4-tiny and Openpose, and then input into an improved LSTM network for training. The comparison results of model recognition ability are shown in Figure 9.

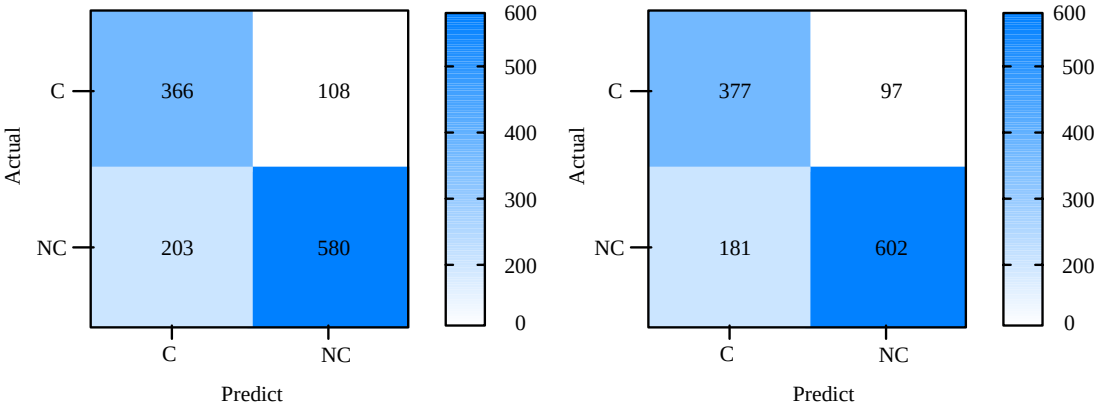


(a) Comparison of accuracy and recall results

(b) FPS comparison results

Figure 9: Comparison results of model recognition ability

The performance comparison of the proposed model with other classification methods in the study is shown in Figure 9. The results demonstrate that the enhanced LSTM model, coupled with a multi-head attention mechanism, achieved an accuracy of 77.8% and maintained a processing speed of 18 FPS. Compared to the basic LSTM model, the improved LSTM improved accuracy by about 2.5 percentage points. Although FPS slightly decreased, the impact of this change on real-time processing capability was minimal. Compared with RNN, SVM, and Softmax classifiers, the improved LSTM significantly improved accuracy while maintaining similar processing speed, verifying its effectiveness in real-time pedestrian crossing intention recognition. The study conducted a classification test on pedestrian crossing intention for all videos in the dataset and recorded experimental data to visually demonstrate the classification effect. The results are shown in Figure 10.



(a) LSTM-based confusion matrix for crossing intention recognition

(b) Confusion matrix for LSTM crossing intention recognition incorporating multi-head attention mechanism

Figure 10: Classification effect

Figure 10 shows the classification effect before and after introducing multi-head attention mechanism. These matrices revealed the model performance in distinguishing whether pedestrians crossed the street or not, showing that the improved LSTM enhanced classification ability in both cases, especially in identifying non-crossing intentions of pedestrians with greater accuracy. The study considered both pedestrian position and key point information as inputs to the intent recognition model and evaluated the contribution of these two types of information to the model performance in Table 3.

Table 3: Model performance evaluation

Pedestrian location	Key points of the human body	Accuracy	FPS
Yes	NO	0.532	76
NO	Yes	0.726	20
Yes	Yes	0.778	18

As shown in Table 3, when only pedestrian position information was used, the model accuracy was 53.2% and the processing speed reached 76 FPS. When relying solely on human keypoint information, the accuracy significantly improved to 72.6%, but the processing speed decreased to 20 FPS. When combining pedestrian position and human key point information, the model achieved the highest accuracy of 78.8%, although the processing speed slightly decreased to 18 FPS at this time. This indicated that combining these two types of information was crucial for improving the accuracy of intent recognition, even at the expense of some processing speed.

To fully verify the superiority of the designed model, the significance test was carried out. The YOLOv4-tiny algorithm before and after the improvement was compared with the LSTM before and after the improvement, as well as the pedestrian intention recognition model. The specific results are shown in Table 4.

Table 4: Results of significance test

Model/Algorithm	AP (%)		t	P
	Before improvement	After improvement		
YOLOv4-tiny	84.79	87.09	2.89	<0.01
LSTM	75.31	77.84	2.13	<0.05
Pedestrian intention recognition model	70.02	78.25	3.67	<0.01

In Table 4, after significance test, it was observed that both YOLOv4-tiny algorithm and LSTM model showed statistically significant performance improvement after improvement. The AP value of YOLOv4-tiny was increased from 84.79% to 87.09%, the t-value was 2.89, and the P-value was less than 0.01, indicating that the improved model achieved significant performance improvement in pedestrian detection tasks. After the improvement of the LSTM model, the accuracy rate was increased from 75.3% to 77.8%, the t-value was 2.13, and the P-value was less than 0.05, which also showed statistical significance. When both the pedestrian position and the key point information of human body were used as input, the performance of the intention recognition model was improved more significantly. The accuracy rate increased from 70.02% to 78.25%, the t-value was 3.67, and the P-value was less than 0.001, which fully proved that the combination of these two kinds of information had a very important impact on improving the accuracy of pedestrian crossing intention recognition.

To evaluate the performance of the research design model (Model 1) more comprehensively, recall rate, precision and F1-score were introduced as evaluation indexes. The comparison methods included the traffic pedestrian detection model in literature [14] (model 2), the traffic pedestrian detection model in literature [15] (model 3), and the traffic pedestrian detection model in literature [16] (model 4). The comparison results are shown in Table 5.

Table 5: Comparison of model performance evaluation indexes

Model	Recall (%)	Precision (%)	F1-score (%)
Model 1	83.54	82.23	82.80
Model 2	78.12	80.29	79.21
Model 3	76.52	79.78	78.10
Model 4	79.24	77.87	78.46

From Table 5, compared with the models in the existing literature, the model designed in this paper showed higher performance in terms of recall rate, precision and F1-score. Among them, the recall rate reached 83.5%, which meant that the model could accurately capture most of the real intention of pedestrians when recognizing the intention of pedestrians crossing the street. The accuracy of the model was 82.2%, which indicated that the model had high precision in recognizing the intention of crossing the street. The F1-score of 82.8% was the harmonic average of recall rate and precision, which further proved the excellent performance of the model in balancing recall rate and precision.

5 Conclusion

As an important component of modern urban traffic management, intelligent transportation systems aim to improve road safety and reduce traffic congestion. This research proposed an intelligent traffic pedestrian detection model that integrated YOLOv4 and improved LSTM. Based on the performance test results, the model's final positioning loss converged at approximately 0.017, while the classification and confidence losses stabilized at around 0.00117 and 0.028, respectively. Although the enhanced YOLOv4-tiny model displayed a slight decrease in FPS, it still fulfilled the real-time detection requirements and proved suitable for embedded systems in vehicles, with an increase of 2.3% in AP. The improved model exhibited better detection ability under occlusion conditions, detecting pedestrians partially occluded in the distance in the image. The improved LSTM model combined with multi-head attention mechanism achieved an accuracy of 77.8% and maintained a processing speed of 18 FPS. Compared to the basic LSTM model, the improved LSTM improved accuracy by about 2.5 percentage points. Although FPS slightly decreased, the impact of this change on real-time processing capability was minimal. The improved LSTM enhanced its classification ability in both scenarios, especially in identifying pedestrian non-crossing intentions with greater accuracy. When only using pedestrian position information, the accuracy was 53.2%, and the processing speed reached 76 FPS. When relying solely on human keypoint information, the accuracy significantly improved to 72.6%, but the processing speed decreased to 20 FPS. When combining pedestrian position and human key point information simultaneously, the model achieved the highest accuracy of 78.8%. Although the processing speed slightly decreased to 18 FPS, it indicated that the key points of human actions played a decisive role in identifying pedestrian crossing intentions, while position information effectively enhanced the accuracy of the model and was a powerful supplement to key point information. The experimental results proved that the intelligent traffic pedestrian detection system designed in the study met the real-time and accuracy requirements of vehicle mounted systems for pedestrian detection and optimized the accurate capture of pedestrian intentions. However, there are still some shortcomings in the research. Although there have been improvements in pedestrian crossing intention recognition compared to existing models, challenges still exist in practical applications. The focus of the research is on optimizing the detection network and extracting pedestrian features in depth. Future work can expand the dataset to include more pedestrian postures and special population categories for model's generalization ability. Its input also needs to be rich, considering the integration of features such as vehicle speed and pedestrian relative distance to cope with the situation of high-speed vehicles. In addition, optimization of model structure and parameter selection, such as pruning the Openpose network model, may further improve the balance between running speed and detection accuracy.

6 Discussion

In the field of intelligent transportation, pedestrian detection and intention recognition have always been the focus and difficulty of research. The intelligent traffic pedestrian detection model proposed in this study, which combines YOLOv4 and improved LSTM, has achieved remarkable results in terms of real-time and accuracy. In the course of the experimental analysis, it was found that the accuracy of the model designed in the study was significantly improved compared with the model using only traditional machine learning algorithms such as SVM and Softmax classifier. This is mainly due to the ability of deep learning algorithms to learn complex data features, especially the advantages of LSTM in processing time series data. In addition, the performance of LSTM in recognizing pedestrian's intention to cross the street was further improved by introducing multi-head attention mechanism. Compared with some models using more complex network structures, such as those based on 3D convolutional networks, the model designed in the study had certain advantages in terms of processing speed and hardware requirements. This is because YOLOv4 and LSTM networks are relatively simple and easy to implement in on-board embedded systems. In the aspect of pedestrian crossing intention recognition, the accuracy of the model can be significantly improved by combining pedestrian location and human key point information. This finding provides a new direction for future research on how to more effectively integrate multi-source information to improve the performance of the model. The optimization and extension of the model can be carried out from the following aspects: First, by increasing the number and diversity of data sets, the generalization ability of the model is improved. The second is to further improve the running speed and detection accuracy of the model by improving the network structure and parameter selection. The third is to consider introducing more characteristic information, such as speed, relative distance of pedestrians, etc., to cope with the situation of high-speed vehicles.

References

[1] N. Islam, and C. Phillips, “Intelligent Traffic Engineering (TE) system for rural broadband,” Computer Networks, vol. 208, pp. 1088-1100, 2022. https://doi.org/10.1016/j.comnet.2022.108888

[2] R. Hu, Y. Xu, H. L. Chen, and F. M. Zou, “A novel method for the detection of road intersections and traffic rules using big floating car data,” IET Intelligent Transport Systems, vol. 16, no. 8, pp. 983-997, 2022. https://doi.org/10.1049/itr2.12116

[3] D. Q. Liu, J. Zhang, J. C. Jin, Y. S. Dai, and L. G. Li,

"A new approach of obstacle fusion detection for unmanned surface vehicle using Dempster-Shafer evidence theory," Applied Ocean Research, vol. 23, no. 3, pp. 1191-1202, 2022. https://doi.org/10.1016/j.apor.2021.103016

[4] M. Y. Wang, H. Ma, S. C. Liu, and Z. D. Yang, e "A novel small-scale pedestrian detection method base on residual block group of CenterNet," Computer Standards and Interfaces, vol. 84, pp. 3-10, 2023. https://doi.org/10.1016/j.csi.2022.103702

[5] Z. R. Tang, R. H. Hu, Y. H. Chen, Z. H. Sun, and M. Li, "Multi-expert learning for fusion of pedestrian detection bounding box," Knowledge-Based Systems, vol. 241, pp. 254-365, 2022. https://doi.org/10.1016/j.knosys.2022.108254

[6] S. Panigrahi, and U. S. N. Raju, "Pedestrian detection based on hand-crafted features and multi-layer feature fused-ResNet model," International Journal of Artificial Intelligence Tools: Architectures, Languages, Algorithms, vol. 30, no. 5, pp. 21-44, 2021. https://doi.org/10.1142/S0218213021500287

[7] P. C. Yan, Q. S. Sun, N. N. Yin, L. L. Hua, S. H. Shang, and C. Y. Zhang, "Detection of coal and gangue based on improved YOLOv5.1 which embedded scSE module*," Measurement, vol. 22, no. 3, pp. 530-542, 2022. https://doi.org/10.1016/j.measurement.2021.110530

[8] W. Jia, S. Q. Xu, Z. Liang, Y, Zhao, H. Min, S. Li, and Y. Yu, "Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector," IET Image Processing, vol. 15, no. 14, pp. 3623-3637, 2021. https://doi.org/10.1049/ipr2.12295

[9] V. P. Nemani, H. Lu, A. Thelen, C. Hu, and A. T. Zimmerman, "Ensembles of probabilistic LSTM predictors and correctors for bearing prognostics using industrial standards," Neurocomputing, vol. 491, pp. 575-596, 2022. https://doi.org/10.1016/j.neucom.2021.12.035

[10] B. J. Chen, T. M. Li, and W. P. Ding, "Detecting deepfake videos based on spatiotemporal attention and convolutional LSTM," Information Sciences: An International Journal, vol. 601, pp. 58-70, 2022. https://doi.org/10.1016/j.ins.2022.04.014

[11] I. C. Kaadoud, N. P. Rougier, and F. Alexandre, "Knowledge extraction from the learning of sequences in a long short-term memory (LSTM) architecture," Knowledge-Based Systems, vol. 235, pp. 107657, 2022. https://doi.org/10.1016/j.knosys.2021.107657

[12] A. A. Yurdusev, K. Adem, and M. Hekim, "Detection and classification of microcalcifications in mammograms images using difference filter and Yolov4 deep learning model," Biomedical Signal Processing and Control, vol. 80, pp. 104360, 2023. https://doi.org/10.1016/j.bspc.2022.104360

[13] S. Dlamini, C. Kao, S. Su, C. Feng, and J. Kuo, "Development of a real-time machine vision system for functional textile fabric defect detection using a deep YOLOv4 model," Textile Research Journal, vol. 92, no. 5/6, pp. 675-690, 2022. https://doi.org/10.1177/00405175211034241

[14] F. Li, Z. Jiang, S. Zhou, Y. Deng, and Y. Bi, "Spilled load detection based on lightweight YOLOv4 trained with easily accessible synthetic dataset," Computers and Electrical Engineering, vol. 100, no. 10, pp. 44-47, 2022. https://doi.org/10.1016/j.compeleceng.2022.107944

[15] H. Mangotra, V. Dabas, B. Khetharpal, A. Verma, S. Singhal, and A. K. Mohapatra, "University auto reply FAQ chatbot using NLP and neural networks," Artificial Intelligence and Applications, vol. 34, no. 5, pp. 278-286, 2022. https://doi.org/10.47852/bonviewAIA3202631

[16] S. Mukherjee, and S. Das, "Application of transformer-based language models to detect hate speech in social media," Journal of Computational and Cognitive Engineering, vol. 23, no. 10, pp. 88-97, 2022. https://doi.org/10.47852/bonviewJCCE2022010102