Computer Science and Information Systems 20(2):679–700 https://doi.org/10.2298/CSIS220813014Y

Using Machine Learning Approach to Construct the People Flow Tracking System for Smart Cities

Baofeng Yao*, Shijun Liu, and Lei Wang

College of Computer Engineering, Bengbu University,
Bengbu 233000, Anhui, China
ybf@bbc.edu.cn
liushijun@bbc.edu.cn
bbxywl@bbc.edu.cn

Abstract: In the crowd congestion in smart cities, the people flow statistics is necessary in public areas to reasonably control people flow. The You Only Look Once-v3 (YOLOv3) algorithm is employed for pedestrian detection, and the Smooth_L1 loss function is introduced to update the backpropagation parameters to ensure the stability of the object detection model. After the pedestrian is detected, tracking the pedestrian for a certain time is necessary to count out the specific number of pedestrians entering and leaving. Specifically, the Mean Shift algorithm is combined with the Kalman filter to track the target. When the target is lost, the Mean Shift algorithm is used for iterative tracking, and then the Kalman prediction is updated. In the experiment, 7,000 original images are collected from the library, mentioning 88 people of which 82 are recognized, and the detection accuracy reaches 93.18%. The 12,200 original images collected in the teaching building include149 people, of which 139 are recognized, with the detection accuracy reaching 93.29%. Therefore, the people flow statistics system based on machine vision and deep learning can detect and track pedestrians effectively, which is of great significance for the people flow statistics in public areas in smart cities and for the smooth development of various activities.

Keywords: smart city; machine learning; machine vision; people flow tracking system; YOLOv3

1 Introduction

At present, construction of smart cities is intensively pursued worldwide as a result of the continuous growth of the population, and the corresponding leading technologies are applied in Europe, North America, Japan, and South Korea. The construction of the smart cities begins in first-tier cities and developed in second-tier cities, and the increasingly "smart" cities bring huge impacts on people's lives [1]. While improving the quality of lives of people, the "smart" city sees increased people flow compared to the past because it brings about complex and diverse "smart" activities. Especially in stations, airports, scenic spots, and large shopping malls, the people gathering results in queuing, congestion, and even trampling in public areas, bringing many security risks to

* Corresponding author

people's social activities [2,3]. Therefore, in public places, timely diversion of the crowd is urgently to be solved. In the process, real-time statistics of the number of people in the area become very important. Through the statistics of people flow, the people flow density can be strictly controlled to maintain social order and ensure the safety of people in public areas [4].

In recent years, the regional population statistics and people flow density analysis have attracted widespread attention in many disciplines such as big data, smart city, and public services [5]. With the rapid advancement of technologies such as artificial intelligence, computer vision, and deep learning, people flow statistics are the basis for the detection of target pedestrians and the rational allocation of resources. Surveillance cameras are widely deployed and people flow statistics are extensively applied in smart cities. With the emergence of the people flow statistics based on machine vision, the pedestrian detection and multi-target tracking become the research hot spots [6,7]. Pedestrian detection is to segment the pedestrian area in the surveillance video, and then track the detected pedestrian target and identify its movement direction, providing data support for the statistics and prediction of the people flow in the area. Currently, the machine learning is applied to integrate the pedestrian features to find the basic characteristics of moving targets, and then deep learning-related algorithms are employed to classify the pedestrian features. The detection performance of the model established is subsequently tested [8-10]. However, the detection of moving pedestrians is still shortcoming due to the dynamic variability of the actual environment and the irregularity of pedestrian movement. Ullah et al. (2022) [11] proposed an intelligent and efficient human flow anomaly detection and recognition system based on artificial intelligence and big data technology. The self-pruning fine-tuned lightweight convolutional neural network classifies the ongoing events in the AI IoT environment as normal or abnormal. When abnormal human traffic is detected, the edge device alerts the relevant department and transmits the abnormal frame to the cloud analysis center. Dong et al. (2021) [12] designed an Ethernet-based flow detection and alarm system based on artificial intelligence. If an exception occurs (such as a long retention time), the system will alert the server. The workflow of parameter setting system and alarm system based on Web Server is introduced. The system can monitor and discover abnormal information in real time, interact with remote web server through Ethernet, and monitor special area effectively.

Statistics of human flow under intelligent video surveillance can be applied in commercial fields and play an important role in security and transportation. The people flow statistics is important to prevent dangerous events caused by excessive crowd density in public places. Therefore, a people flow statistics system is designed for smart cities based on machine vision and deep learning. In the study, the development status of comprehensive urban body under smart cities is analyzed first, and human target recognition and target tracking technology are then discussed to relieve the pressure of public facilities. The pedestrian detection in this study is implemented by YOLOv3 algorithm, and the Smooth_L1 loss function is innovatively introduced aiming at the model instability caused by L2 loss function in the process of network backpropagation parameter update. The tracking algorithm of Kalman filter is simulated according to its basic principle, and an improved tracking algorithm using Mean Shift is proposed. Finally, the human flow system is experimentally simulated to prove that the system is practical and can meet the general situations. The improved algorithm proposed in this study is applied to the statistics of the flow of people in public areas, which is of great

significance for the reasonable control of the flow of people and the maximization of the function of smart cities. The structure of this paper is as follows. The second section summarizes the current research background and the existing human flow statistics technology, and identifies the problems expected to be solved in this research. In Section 3, the flow statistics scheme based on machine vision and deep learning is implemented, and an improved algorithm introducing Smooth_L1 loss function is proposed based on YOLOv3 algorithm. In Section 4, the proposed improved algorithm is experimentally verified, which proves the advantages of the improved algorithm in pedestrian detection and recognition. The innovation and contribution of this work are as follows:

- In the module of human object detection, YOLOv3 detection algorithm is applied. On this basis, Smooth_L1 loss function is innovatively introduced to optimize the algorithm to avoid missing detection of objects.
- Considering that the short time tracking of the target has certain practical significance for the statistics of human flow, the Kalman filter and Mean Shift are combined to complete the iterative tracking first, and then the Kalman prediction is updated at last.
- The human flow statistics system based on machine vision and deep learning can effectively detect and track pedestrian targets, and provide help for the standardized development of commercial activities and leisure and entertainment projects in smart cities.

2 Research Background

“Smart City" emerged at the end of the last century to promote the sustainable development of cities. The new generation of information technology such as the Internet of Things and cloud computing has promoted the construction of “smart cities”, and to realize the sustainable development of the city has become one of the characteristics of the current era. The essence of a smart city is to realize information sharing, rational allocation of resources, and collaborative operations among different systems, based on which the decisions being conducive to urban development and management can be made to predict and respond to emergencies effectively. The emergence of urban complexes has played an important role in urban development and regional positioning. However, the ecological complexity has become increasingly significant as the central business districts gradually become saturated. The urban complex combines many functions such as residence, office, commerce, and entertainment, which influence and promote each other [13]. The urban functional morphology is essentially the interaction between the hierarchical units, and the urban complex should be integrated into it as an important functional node of the city, coexisting with the urban system. Thanks to the diverse functions of the urban complex, it can serve the people more effectively and increase people flow [14]. The digital technology makes crowd control possible in the rapid urbanization to mitigate the increasing congestion of human and road traffic and to jointly built sustainable, energy-saving, and livable cities (Figure 1).

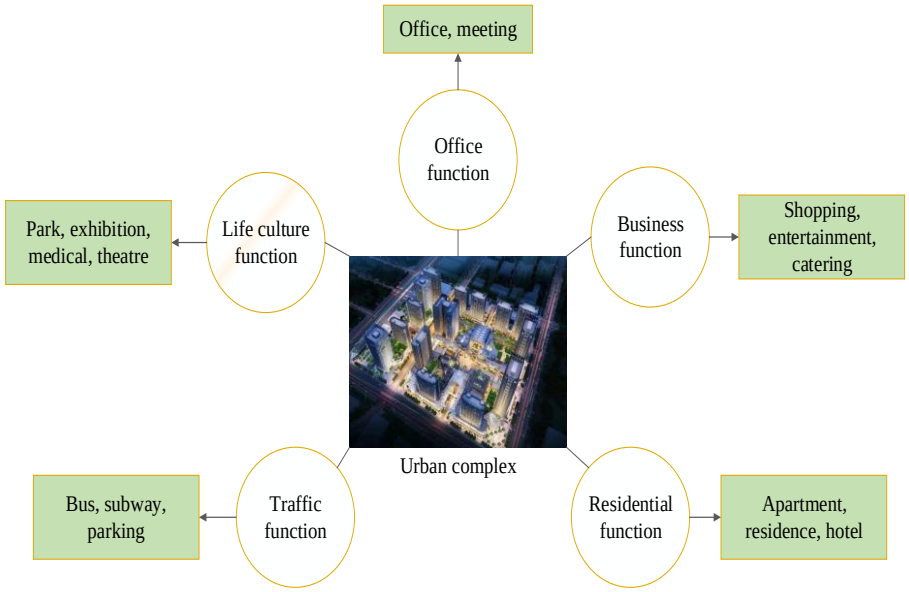


Fig. 1. Relationships between urban complexes in smart cities and their functions

While the living standards of people are steadily improving, the spiritual needs become higher and higher. The commerce and tourism industries in smart cities have achieved unprecedented development under the advancement of modern information technology and Internet technology. People have more entertainment projects, with more attention paid to the diversified development of life. Although the development of smart cities enriches the leisure and entertainment life of the masses, the tourists gathering brings a huge test to the crowd control of public areas such as scenic spots and shopping malls. The rapid social development and excessive people flow have caused many social problems and security risks. Consequently, real-time statistics of the number of people in the area are very important. Scientific control in the number of visitors can improve the management and service levels, prevent crowding, detention, and trampling, guaranteeing the personal safety and property safety. The people flow statistics is an important research method, and its related technologies have become common in some developed countries such as Europe and the United States. The people flow statistics enables the public facilities to operate within the maximum carrying capacity, and the people flow density is always controlled within a safe range.

Chen et al. (2022) [15] proposed the distributed probability adjusted confidence (DPAC) function, which can optimize the reliability of model prediction according to the actual situation. Based on the target confidence of the YOLOv4 network, the DPAC function was added to reduce or increase the confidence according to the target distribution and then output the final confidence. The proposed method showed obvious advantages in image-based traffic statistics. Zhang et al. (2020) [16] proposed a bus passenger flow statistics algorithm based on Single Shot MultiBox Detector (SSD) and Kalman filter to obtain passenger flow statistics from surveillance cameras on buses. In this method, the SSD model was modified into two types of models. Firstly, the two types of SSD models were trained by using the bus dataset, and then the passenger

position in each frame was detected by the model and tracked by the Kalman filter. Finally, according to the passenger trajectory, the traffic statistics of passengers getting on and off the bus were generated. Xie et al. (2020) [17] believed that from the perspective of spatial prediction measure, urban flow prediction can be divided into three categories, namely, city level, regional level, and road level. From the perspective of time prediction, urban flow prediction can be divided into three categories, such as short, medium, and long-term flow prediction. Many new methods have been proposed in recent years. The main methods can be divided into five categories: statistical methods, traditional machine learning methods, deep learning-based methods, reinforcement learning methods, and transfer learning methods.

It is concluded that most of the existing pedestrian detection technologies use machine learning technology to integrate pedestrian features and detect moving pedestrian objects in video. These methods generally find the basic features of moving objects through many samples, use neural networks and deep learning algorithms to train and classify various pedestrian features to form reliable data sets and models, and then bring these models and data sets into common detection scenarios. However, the information of moving objects is lost or mutated due to the dynamic changes of the background environment and the irregularity of pedestrian movement, which seriously affects the detection and tracking of moving pedestrians. Based on this, this work is developed by focusing on this difficulty.

3 The People Flow Statistics Based on Machine Vision and Deep Learning

3.1 Human Body Recognition

Innovation and digitalization have developed into new growth poles in recent years gradually. Countries around the world are actively looking for solutions for a balance among economy, technology, and society development. Generally, the intelligent video surveillance has become an inevitable trend in the development of the smart cities. It can extract and screen abnormal behaviors in video and issue early warnings in time, completely changing the passiveness that traditional surveillance can only "monitor" but not "control" [18,19]. From a technical point of view, there are mainly two categories of intelligent analysis technologies: one is to extract and detect the video targets with the image segmentation and foreground extraction technologies, and the other is to build a model of specific objects in the image using the pattern recognition technology. After a large amount of sample training, the detection of specific targets in the video is completed. The ultimate task of people flow statistics based on intelligent video surveillance is to count the number of people. Dynamic human dynamic characteristics refer to the distinctive characteristics that people show under dynamic conditions, such as the swing amplitude of shoulder, arm, elbow, leg, and foot of each person is different when walking. The difficulty lies in the capture of dynamic human target through video. Hence, target detection and human tracking should be completed.

Background modeling is the most common in target detection. It is a method to segment all moving targets in a scene under a fixed camera. The key is how to build a

background model from a video sequence. The classic background modeling methods include single Gaussian, inter-frame difference, mixed Gaussian, VIBE, and CodeBook. In practice, a suitable method is chosen by jointly considering the application scenario, the type of detection target, and the performance of the hardware platform [20]. The modeling principle is as follows. A background model is established based on the relevant parameters of the background, the difference operation of the current frame of the video is performed then, and the binarization result of the difference image is obtained. The foreground and background images can be distinguished by the gray value change degree, and the background image is the one with less gray value change. The specific process of the background modeling method is shown in Figure 2. In actual applications, the moving targets are detected through background modeling and then sent to the deep learning network for classification and recognition, which can ensure the real-time performance and the accuracy of classification [21].

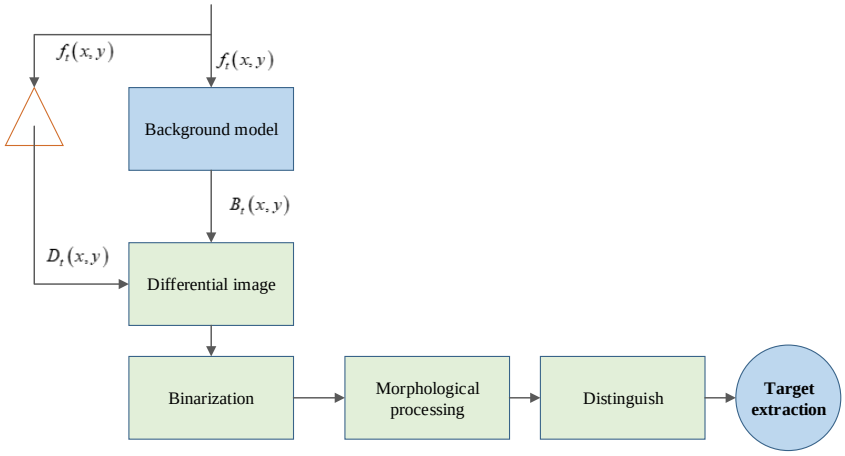


Fig. 2. The specific process of the background modeling method

Assuming that the current frame image is $f_t(x,y)$ and the background image is $B_t(x,y)$. (x,y) refers to the coordinate of image, and n represents the frame of the image. The image is input and compared with the background model to obtain the change information $D_t(x,y)$ of each pixel in the image to better detect the moving target. The accuracy of $B_t(x,y)$ directly affects the detection effects of moving targets. The premise of the background modeling method is to obtain the background image quickly and accurately. Further, $B_t(x,y)$ will change due to factors such as illumination, and the background model is updated in real time to adapt to the changing background.

The average background model is a background modeling algorithm that is more sensitive to environmental lighting changes and background multi-modality. The basic idea is as follows. If the coordinate location of the image is set as *(x,y)*, the average value of each pixel is calculated as its background modeling. When the current frame is detected, it only needs to subtract the average pixel *u(x,y)* of the same position in the background model from the pixel of the current frame *I(x,y)* to get the difference *d(x,y)*,

which will be compared with a threshold *TH*. It is deemed as foreground when the value is greater than the threshold, otherwise it is the background. Input image is a binary image.

The calculation process of the average background model is as follows [22].

$$d(x,y) = I(x,y) - u(x,y) \tag{1}$$

$$output(x,y) = \begin{cases} 1, & |d(x,y)| > TH \\ 0, & otherwise \end{cases} \tag{2}$$

The threshold TH can be determined by an adaptive algorithm, and the average value u_{diff} and standard deviation $diff_{std}$ of the inter-frame difference of each pixel are calculated. $I_t(x,y)$ represents the pixel value in the image (x,y) at time t, inter represents the interval between two frames, and $F_t(x,y)$ is as follows.

$$F_t(x,y) = \left| I_t(x,y) - I_{t-\text{inter}}(x,y) \right| \tag{3}$$

$$u_{diff}(x,y) = \frac{1}{M} \sum_{t=\text{inter}}^{M} F_t(x,y) \tag{4}$$

$$diff_{std}(x,y) = \sqrt{\frac{1}{M} \sum_{t-\text{inter}+1}^{M} \sum \left(F_t(x,y) + u_{diff}(x,y) \right)^2} \tag{5}$$

In the above equations, M refers to the number of iterations. Usually, M should be large enough to ensure the accuracy of u_{diff} and $diff_{std}$, and TH can be determined according to u_{diff} and $diff_{std}$.

$$\text{TH} = u_{diff}(x,y) + \beta \times diff_{std}(x,y) \tag{6}$$

where, the value of β is generally 2.

After the background modeling, the human body is detected from the video sequence and obtain the position information using the human body recognition. The human body recognition generally includes extracting the feature information of the target object and constructing the corresponding classifier to complete the target classification. The key of human body recognition method is to extract good features for clustering and design efficient classifier for behavior recognition. As a common feature in behavior recognition, the global feature is usually defined by background subtraction and tracking location, and then the interesting region is divided into a whole for description. Human body feature descriptors can be classified into three categories: low-level features, learning-based features, and mixed features. The global feature extraction regards the object as a whole. The target tracking algorithm locates the human body in the video, and then encodes the positioned target to form its global feature. However, the global feature extraction relies too much on the underlying visual processing, and the global feature is very sensitive to noise and occlusion in the image [23]. The local feature extraction is to extract the spatiotemporal interest points first in the video, and extract corresponding image blocks around these points. Finally, all image blocks are combined to jointly describe a specific action. The premise of local feature extraction is to have a sufficient number of stable interest points related to the action category. As a

result, the preprocessing process is more cumbersome [24]. The global feature extraction and local feature extraction are shown in Figure 3 and Figure 4, respectively. Global feature refers to the global feature information of each image, and the local feature is that in a certain area in the image. Finally, multiple local features are merged as the final feature.

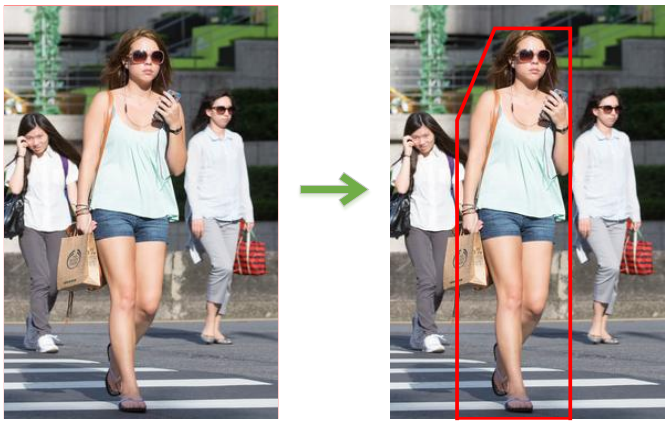


Fig. 3. Global feature extraction

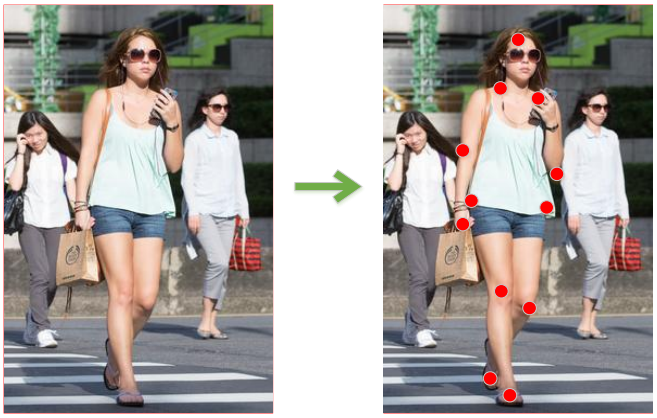


Fig. 4. Local feature extraction

To extract target features is the prerequisite for human body recognition, and the classifier is the key to classifying the image using the extracted target features [25]. After the HOG feature is extracted as the feature basis of human body, how to make the next decision is what we need to study. This section will make a detailed analysis of the training method of pedestrian classifier. After comparing different classifiers, we decided to use the support vector machine (SVM) with the most reliable sample partitioning effect as the classifier [26-28]. Many samples should be trained before classification. If the pedestrian classifier in Opencv library is directly used, it can be used. Due to the great difference between the training samples and the hypothetical scenarios in the system in this work, the recognition accuracy is not high. Therefore, it

is necessary to train the classifier with samples more in line with the actual scene to improve the accuracy to the required level. The major objective of this work is to count the pedestrian flow, so it is necessary to select a database containing various pedestrian poses for training.

3.2 Human Target Detection Algorithm Based on Yolov3 Network

Generally, the deep learning-based target tracking includes target tracking based on deep features and target tracking based on the deep neural network (DNN). The DNN-based target detection shows a wider application range and better effects. Convolutional neural network (CNN) is a multi-layer neural network. Once the distribution of training data and test data is different, the generalization ability of the network is also greatly reduced [29,30]. The YOLOv3 network (Figure 5) is a CNN for detection tasks. The v3 algorithm is formed on the basis of v1 and v2. The speed and accuracy can be controlled by changing the size of the model structure. YOLO only uses convolutional layers. It contains 75 convolutional layers, with skipping connection and up-sampling layers. Besides, it doesn't use any form of pooling, but uses a convolutional layer with a stride of 2 to down-sample the feature map, which is conducive to preventing loss of low-level features due to pooling. Layers 0 to 74 of YOLOv3 are the main network structures for feature extraction, and their convolutional layers are obtained by integrating various mainstream network structures with better convolutional layers. The 75th to 106th layers are feature interaction layers to predict boxes of different scales [31]. The priori detection system of YOLOv3 reuses the classifier or locator to perform the detection task, and applies the model to multiple positions and scales of the image, and those areas with higher scores can be regarded as the detection results.

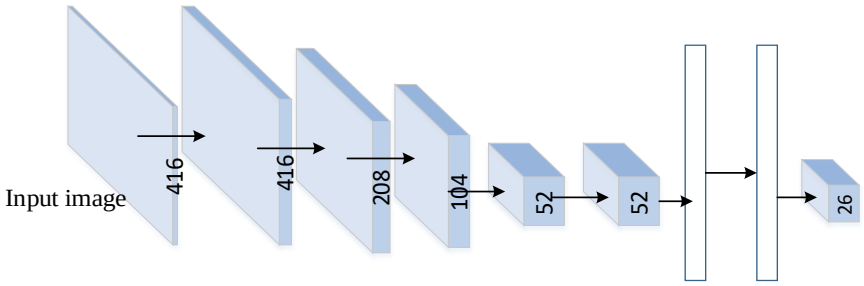


Fig. 5. YOLOv3 network structure

YOLOv3 is a detection network based on regression, and it judges whether there is an interest target in the image to extract target features through the network. YOLOv3 uses the previous 52 layers of darknet-53, with a large number of residual skip-floor connections applied. Meanwhile, the conv's stride is adopted for downsampling to reduce the gradient negative effects caused by pooling [32]. In this network structure, a convolution with a step size of 2 is used for downsampling. The image is divided into $S \times S$ by YOLOv3, each grid predicts bounding boxes at three different scales, and each bounding box predicts 4 coordinate values (t_x, t_y, t_w, t_h) and c classification predictions,

where (t_x, t_y) is the coordinate of the predicted bounding box relative to the grid unit, and (t_w, t_h) is the logarithmic ratio of the predicted bounding box size to the preset a priori box size. The bounding box is shown in Figure 6. The distance from the target grid to the upper left corner of the image is expressed as (c_x, c_y), and the position information of the bounding box is then obtained as follows [33].

$$b_x = \sigma(t_x) + c_x \tag{7}$$

$$b_y = \sigma(t_y) + c_y \tag{8}$$

$$b_w = p_w e^{t_w} \tag{9}$$

$$b_h = p_h e^{t_h} \tag{10}$$

$$confidence = \sigma(t_o) = p_r(object) \times IOU(b, object) \tag{11}$$

where, t_o is the target prediction, and $confidence$ is the confidence level to judge whether the bounding box is true, $p_r(object)$ judges whether there is a target in the grid, and $IOU(b, object)$ is the intersection ratio of the predicted bounding box and the true bounding box, defined as below (Figure 7).

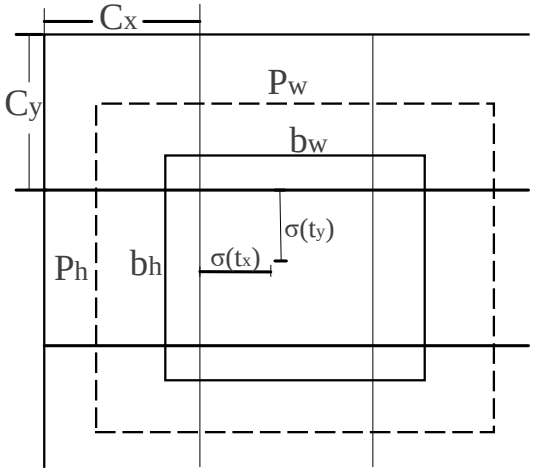
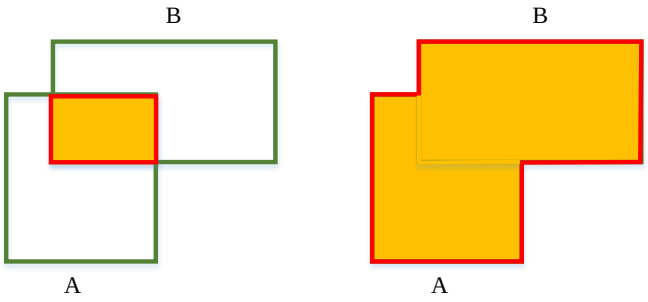


Fig. 6. Boundary box

The Smooth_L1 loss function is used in the Faster R-CNN algorithm. Compared with the L2 loss function, Smooth_L1 is not sensitive to abnormal values. The derivative of the average absolute value error of L1 loss function is constant, so when the loss value is small, the gradient obtained is relatively large, which may cause model oscillation and is not conducive to convergence. The L2 loss function, due to the square operation, will amplify the error when the difference between the predicted value and the true value is greater than 1. Especially when the input value of the function is far

from the center value, the gradient is large when the gradient descent method is used to solve it, which may cause gradient explosion. At the same time, when there are multiple outliers, these points may occupy the main part of Loss, and many effective samples need to be sacrificed to compensate for it. Therefore, in Bounding box regression of object detection, Smooth L1 Loss is considered early for the following reasons: (1) compared with L1 Loss, it can converge faster; and (2) compared with L2 Loss, it is not sensitive to outliers and outliers, and the gradient change is relatively smaller, so it is not easy to run and fly during training.



$$IOU = \frac{A \cap B}{A \cup B}$$

Fig. 7. Schematic diagram of IOU definition

In this work, the Smooth_L1 loss function is introduced to calculate the loss of the width and height of the bounding box to overcome the model instability caused by large parameter changes with abnormal values. The Smooth_L1 loss function essentially combines the L1 and the L2 loss functions. L2 is dominate when the difference between the predicted value and the target value is small; while L1 is active when the predicted value differs greatly from the target value. The curve graphs of the three loss functions are shown in Figure 8.

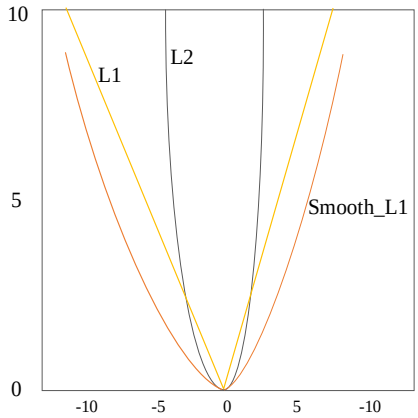


Fig. 8. Three loss functions of L1, L2, and Smooth_L1

The Smooth_L1 loss function can be expressed as follows:

$$Smooth_{L1}\left(b_i - f\left(a_i\right)\right) = \begin{cases} 0.5\left(b_i - f\left(a_i\right)\right)^2, \left(\left|b_i - f\left(a_i\right)\right| < 1\right) \\ \left|b_i - f\left(a_i\right)\right| - 0.5, \left(\left|b_i - f\left(a_i\right)\right| \geq 1\right) \end{cases} \quad (12)$$

The detection algorithm based on deep learning gives higher requirements on the data set, and the trained network weight model shows a higher accuracy rate. The characterization model is selected based on the characteristics of the whole human body, and detection results depend on the completeness of the human target in the video. However, since human targets are easily occluded, good detection results are usually not achieved. In addition, there are characterization models for head information, facial features, etc., which can reduce the occlusion of human targets, whereas missed detection will occur due to the limited camera capture. In this work, the representative pedestrian detection sample set ImageNet is used to perform high-resolution pre-training on YOLOv3 to obtain the classification network. The VOC data set is then used for training to obtain the final training model. All images in the ImageNet image data set have at least one frame. The detection for 200 targets has produced 470,000 images, and each image has 1.1 targets on average. The VOC data set mainly includes four categories, namely people, common animals, transportation vehicles, and indoor furniture. It mainly serves three types of tasks: image classification, object detection and recognition, and image segmentation. The performance of the pedestrian detection algorithm is evaluated using the P-R curve.

3.3 Multi-Head Target Tracking Based on Kalman Filter

After the human targets are extracted and detected based on deep learning algorithms, the pedestrians are tracked in the video for a certain period to count the specific number of pedestrians entering and exiting in detail and to improve the accuracy. Usually, multiple pedestrians appear in the same scene at the same time. Hence, a multi-target tracking algorithm is employed to track multiple pedestrian targets at the same time [34]. The Kalman filter algorithm is a mainstream algorithm with an excellent processing effect on Gaussian process noise during target tracking. The Kalman filter includes a state model and an observation model. The state model is used to describe the relationship between continuous-time states, which can be expressed as equation (13), and the observation model can be expressed as equation (14).

$$x_k = F_k x_{k-1} + B_k u_k + W_k \quad (13)$$

$$z_k = H_k x_k + v_k \quad (14)$$

where, x_k and x_{k-1} represent the state vector at time k and k-1, respectively. F_k represents the state transition matrix, which is used to describe the relationship between the system state at time k and k-1. B_k is the control input matrix, which represents the connection between the control input parameters and the state vector. W_k is the process noise vector, which can be regarded as a normal distribution that obeys the zero mean value. z_k represents the observation vector, H_k represents the transition matrix, and v_k is the observation noise.

The Kalman filter algorithm has two basic hypotheses. I. A sufficiently accurate model is a linear (or time-varying) dynamic system excited by white noise; and II. Each measurement signal contains additional white noise components [35]. When the above hypotheses are met, the Kalman filter algorithm can be applied. The Kalman filter mainly includes two stages: prediction and update. The prediction stage is to predict the next time information based on the current state information pair, and the update stage is to update the target position based on the predicted information. The update equation for the Kalman filter time can be expressed as follows [36].

$$\hat{x}_{\bar{k}} = A\hat{x}_{k-1} + Bu_{k-1} \tag{15}$$

$$P_{\bar{k}} = AP_{k-1}A^T + Q \tag{16}$$

$$K_k = \frac{P_{\bar{k}}H^T}{HP_{\bar{k}}H^T + R} \tag{17}$$

$$\hat{x}_k = \hat{x}_k + K_k\left(z_k - H\hat{x}_{\bar{k}}\right) \tag{18}$$

$$P_k = \left(I - K_k H\right)P_{\bar{k}} \tag{19}$$

where, $\hat{x}_{k-1}$ and $\hat{x}_k$ represent the posterior state estimates at t-1 and t, respectively, and are the filtering results; $\hat{x}_{\bar{k}}$ represents the prior state estimates at t; P_{k-1} and P_k represent the posterior estimate covariance at t-1 and t, respectively; $P_{\bar{k}}$ is the a priori estimated covariance at time t. H is the conversion matrix from state variable to observation, which connects the state and observation; z_k is the observation value, and is the input of filtering; K_k is the filter gain matrix; A is the state transition matrix; Q is the process excitation noise covariance; R is the measurement noise covariance; B is the matrix that converts the input to state.

The Kalman filter tracking algorithm only needs to input the current state information as the initial data to predict the target motion state at the next moment, and the previous moment data can be discarded after each prediction update [37-39]. Therefore, the Kalman filter has a good tracking effects on continuous moving targets. However, when there are multiple targets in the video and there is interference between each other, it will cause chaos in multi-target tracking or target loss. The Mean Shift algorithm relies on the probability density value of the current human head target feature sample in the region, so it can be used for clustering, image segmentation, and target tracking. Therefore, the Mean Shift algorithm is incorporated into the Kalman filter to tract the target first. When the target is lost, the current tracking information of the Kalman filter is brought into the Mean Shift algorithm for iterative tracking [40-42]. Finally, the Kalman prediction update is performed. Since the offset of the targets in two adjacent frames is very small and the target box B1 of the current frame F1 is known, a heuristic approach is to still select the known B1 area in the next frame F2 to infer the offset required by the target box according to the similarity between the B1 area in F2 and the B1 area in F1, and then get the target box B2 of F2. This is what the MeanShift algorithm needs to solve. Combined with MATLAB source code, the

tracking process of MeanShift is shown in Table 1. The flow of the optimized pedestrian multi-target tracking algorithm is shown in Figure 9.

Table 1. The tracking process of the Mean Shift algorithm

```
To calculate the histogram of the candidate area
hist2 = zeros(1,4096);
To calculate the histogram of the candidate area
for i = 1:a
   for j = 1:b
      % rgb color space quantization 16 is *16*16 bins
      q_r = fix(double(current_temp(i,j,1))/16); % fix integral function is taken as it approaches 0
      q_g = fix(double(current_temp(i,j,2))/16);
      q_b = fix(double(current_temp(i,j,3))/16);
      q_temp(i, j) = q_r*256 + q_g*16 + q_b;    % to set the proportion of red, green, and blue
components to each pixel
      % to calculate the weight of each pixel in the histogram
      hist2(q_temp(i, j)+1) = hist2(q_temp(i, j)+1) + m_wei(i,j);
   end
end
hist2 = hist2 * C;  The above steps complete the histogram of the target kernel function
```

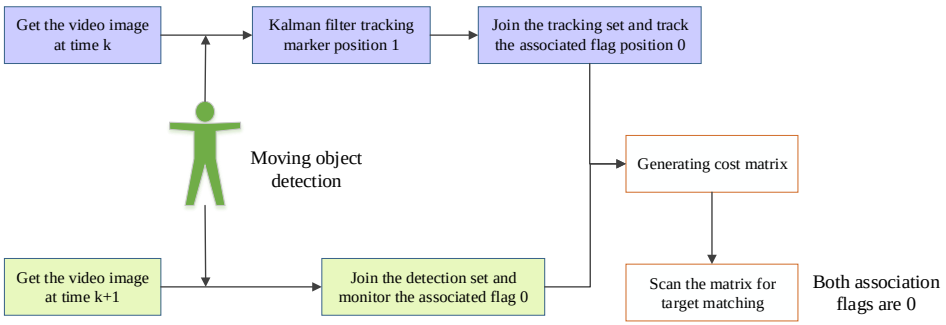


Fig. 9. Optimized pedestrian multi-target tracking algorithm

3.4 Experimental Design and Model Performance Evaluation

The machine learning is combined with the Kalman filter algorithm in this work to complete the detection and tracking of human head targets for people flow statistics. The pedestrian motion video sequence in the surveillance video is acquired first, followed by the segmentation of the area where the pedestrian is located. The YOLOv3 network is used to detect the extracted target to identify all the head targets. Finally, the optimized pedestrian multi-target tracking algorithm is used to achieve the people flow statistics. The software development environment of the people flow statistics system is Visual Studio 2013, together with the OpenCV image processing library. Databases for image and video processing for the OpenCV are the most comprehensive, which can satisfy the video extraction, image morphology processing, and target detection. The face recognition in the video includes two steps. Step 1: the image size is adjusted to

92×112, that is, the same size as the trained image. Step 2: the predict function is used to detect the face in the image, and the function concerns two elements: the label to identify the individual and the confidence. A smaller confidence indicates a higher degree of match. 0 means a perfect match, but the score mechanism of the confidence is different for of different algorithms.

The hardware platform of the traffic statistics system can be divided into five parts: front-end acquisition module, signal transmission module, information storage module, computer processing module, and output display module. The front-end acquisition module is the premise and foundation of the whole system, and the scene in the front-end detection area is converted into image signals by CCD cameras. The signal transmission transmits the collected image signal to the processing equipment by Ethernet. The information storage module mainly refers to the server, which is used to store the image data collected by the front end for later search; the computer processing module mainly uses workstations or high-performance computers, which is the core part of the whole system and is used to analyze and process image signals. The output module can display front-end monitoring images and real-time traffic data.

To detect the accuracy, the original video is produced in the campus library and the teaching building, and the video resolution is 640×480. The tracked pedestrians are marked with the yellow line frame to better identify pedestrian targets and facilitate the pedestrian order distinction, based on which the detection effects are then evaluated.

4 Results and discussion

4.1 YOLOv3 +Smooth_L1 model

In this work, the weight model of the YOLOv3 algorithm is applied to detect the pedestrians in real life scenes. The model is trained using the VOC data set, which uses the characteristics of the whole human body as the representation information. Adam is used as an optimizer to minimize the loss function, $\beta_1 = 0.9$ $\beta_2 = 0.999$. Training can be carried out when batchsize is set to 2, so the experiments are carried out when batchsize=2. An epoch refers to a complete training of all training data. The initial learning rate is set to 0.0001. After 10 epochs, it is reduced to one-tenth of the current value. When 30 epochs are finished, the model has converged. The specific hyperparameter setting of the model is listed in Table 2 below.

Table 2. The specific hyperparameter setting of the model

Hyperparameters	Value
β_1	0.9
β_2	0.999
Batchsize	2
Initial learning rate	0.0001

Entrance of the campus teaching building is selected as the video scene for detection, as shown in Figure 10. It was evident that the detection effects of the YOLOv3 model are good, but pedestrians with serious occlusion are missing. When the people flow

density is intensive, the occlusion of human targets will be more serious, bringing great difficulty for people flow statistics.

Fig. 10. Pedestrian detection results based on the YOLOv3 model

To further reduce the missed detection of the YOLOv3 model, the Smooth_L1 loss function is introduced to calculate the width and height of the bounding box. The model established is then trained and tested, and compared with the weight model obtained using the L2 loss function. The P-R curve obtained is shown in Figure 11. It is evident that the P-R curve of the Smooth_L1 loss function is closer to the upper right, indicating that the model trained is superior in accuracy and recall rates and is more efficient.

The experimental results demonstrate that the mAP value of the improved YOLOv3 algorithm by introducing Smooth_L1 loss function is higher than the optimal mAP value of YOLOv3 algorithm. Mean average precision (mAP) refers to the average mean recognition accuracy of each class of objects when the model recognizes multiple classes of objects. The higher the mAP value, the higher the recognition accuracy of the model. The average precision (AP) is the area under the P-R curve of the pedestrian detection algorithm. The higher the AP value, the better the detection performance of the algorithm. The data calculation results of the training log are shown in Table 3. It can be observed that, compared with YOLOv3, the recall rate of pedestrian target detection of YOLOv3 algorithm improved by introducing Smooth_L1 loss function is increased from 82% to 84%. In the study of Zhang et al. (2021), a Caps-YOLO model based on YOLOv3 real-time object detection model was proposed for pedestrian problem detection. In this model, dense connections are used to replace the shortcut connections in the original network, and dense block components are constructed to improve the utilization of feature maps. This is consistent with the original intention of choosing the YOLO model in this work, and also confirms the reliability of the proposed model in pedestrian detection.

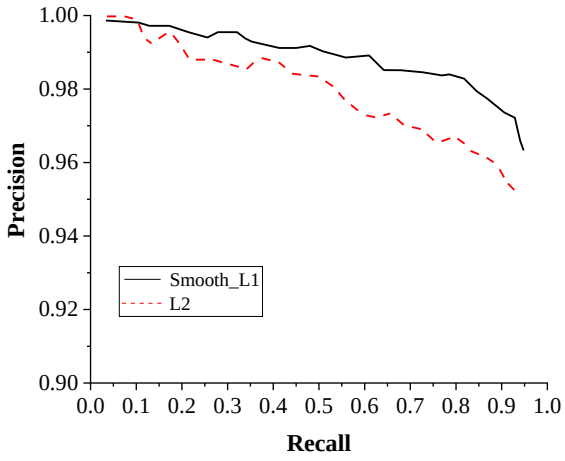


Fig. 11. PR curves of different loss function models

Table 3. Comparison of target detection results between YOLOv3 and YOLOv3+ Smooth_L1

Algorithm	AP/100%	AP50/%	AP60/%	P/100%	R/%
YOLOv3	23.2	72.9	46.7	53.5	82
YOLOv3+Smooth_L1	27.2	76.3	52.7	64.5	84

Furthermore, the proposed algorithm is compared with the human FLOW statistics method based on contour rules on the flow dataset. The FLOW dataset consists of 1148 frames and includes 9 persons. It simulates the scene of rapid vertical flow of people, such as underground passage and building entrance and exit, and includes part of lateral movement. In this work, the sequence is used to evaluate the statistical accuracy of the proposed algorithm in the case of rapid movement and direction change. The comparison results are shown in Table 4. It can be observed that the F1 score of the algorithm in this work is more than 95% and even as high as 100% in the vertical counting of rapid personnel flow, which is obviously better than the human flow detection method based on contour rules.

Table 4. Performance comparison of YOLOv3+Smooth_L1 and contour rule-based human flow statistics method

Algorithm	Direction	Precision (%)	Recall (%)	F1 (%)	Tracking time (s)
Contour rule-based human flow statistics method	Longitudinal (upward)	100	90	95	0.009
YOLOv3+Smooth_L1	Vertical (down)	94	88	88	0.009
	Horizontal (to the left)	100	100	100	0.014
	Landscape (to the right)	95	100	95	0.014

4.2 Pedestrian Tracking Results Based on Kalman Filter

In this work, the Mean shift algorithm is incorporated into the Kalman filter to track the moving pedestrian targets, which can correlate images of different frames of the same target to avoid no repeated count. At the same time, the movement direction of the target is identified to count the specific number of targets accurately. Figure 12 suggests that the system can track the two targets effectively, avoiding the target distortion and target loss.

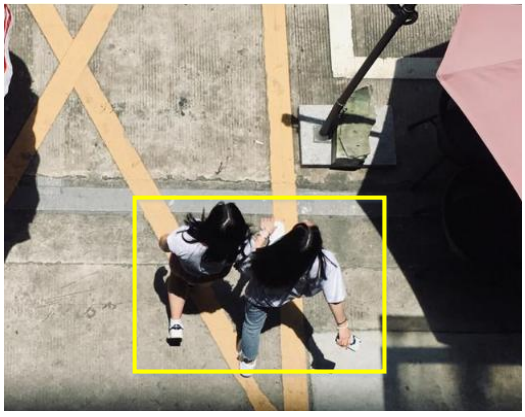


Fig. 12. Pedestrian tracking effects based on Kalman filter

In this work, people on the campus library and the teaching building are counted, and the results are shown in Figure 13. 7,000 original images are collected in the library, involving a total of 88 people. Of them, 82 people are detected, and the detection accuracy reaches 93.18%. There are 12,200 original images collected in the teaching building, involving a total of 149 people. Of them, 139 people are detected, and the detection accuracy is 93.29%. Obviously, tracking the moving target by undertaking the result of the Kalman filter as the input value of the Mean Shift algorithm can reduce the target loss caused by the occlusion effectively, ensuring high target tracking efficiency.

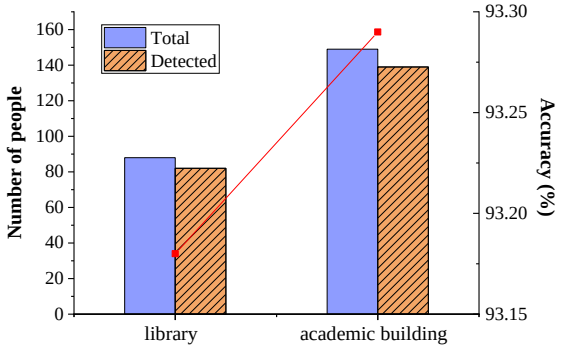


Fig.13. Pedestrian detection results in different scenarios

5 Conclusion

A smart city is characterized by the connection of the scattered and discontinuous parts of the city and multiple functions. With the continuous development of the city, the existing functions may not be able to adapt to the current public life and needs. The construction of smart city brings about crowd congestion and ineffective control. To alleviate the pressure brought by the large people flow, the technologies used for people flow statistics are analyzed, among which the human target recognition and target tracking are most important. In the human recognition module, the human body is first detected from the video sequence based on machine vision, and the position information is then obtained, and the extracted global features and local features are merged as the final feature of the target. The short-term tracking of the target has a certain practical significance for people flow statistics, so the Mean Shift algorithm is incorporated into the Kalman filter to perform iterative tracking, and then the Kalman prediction is performed.

The innovation of this work lies in the application of YOLOv3 detection algorithm in the human object detection module, and on this basis, Smooth_L1 loss function is introduced to optimize the algorithm to avoid missing detection of the object. The experimental results reveal that the YOLOv3 model shows a good target detection effect, but pedestrians with serious occlusion are missing. After the Smooth_L1 loss function is introduced, the P-R curve is closer to the upper right side, which can avoid the model instability caused by large parameter changes with abnormal values. By incorporating the Mean shift algorithm into the Kalman filter to track moving pedestrian targets, high target tracking efficiency is achieved in both the campus library and the teaching building. This work contributes that the human flow statistics system based on machine vision and deep learning can effectively detect and track pedestrian targets and provide help for the standardized development of commercial activities and leisure and entertainment projects in smart cities. There are still some shortcomings in this work. The computation amount of the system algorithm is huge, and the processing ability of the computer is high. In addition, the system shows poor response ability to fast pedestrians, which is easy to miss, resulting in reduced accuracy. Therefore, the algorithm will continue to be optimized in the subsequent research to reduce the computational complexity of the algorithm.

Acknowledgment. This work was supported by the Natural Science Research Project of Anhui Province, China (KJ2021A1121).

References

1. Meijer, A., Bolívar, M. P. R.: Governing the smart city: a review of the literature on smart urban governance. International review of administrative sciences, Vol. 82, No. 2, 392-408. (2016)
2. Meijer, A., Thaens, M.: Urban technological innovation: Developing and testing a sociotechnical framework for studying smart city projects. Urban Affairs Review, Vol. 54, No. 2, 363-387. (2018)

3. Romão, J., Kourtit, K., Neuts, B., Nijkamp, P.: The smart city as a common place for tourists and residents: A structural analysis of the determinants of urban attractiveness. Cities, Vol. 78, 67-75. (2018)
4. Guevara, S., Singh, Y., Shores, A., Mercado, J., Postigo, M., Garcia, J., Newell, B.: Development of a Pilot Smart Irrigation System for Peruvian Highlands. Journal of Contemporary Water Research & Education, No. 171, 49-62. (2020)
5. El-Sayed, H., Chaqfa, M., Zeadally, S., Puthal, D.: A traffic-aware approach for enabling unmanned aerial vehicles (UAVs) in smart city scenarios. IEEE Access, Vol. 7, 86297-86305. (2019)
6. Rehman, T. U., Mahmud, M. S., Chang, Y. K., Jin, J., Shin, J.: Current and future applications of statistical machine learning algorithms for agricultural machine vision systems. Computers and electronics in agriculture, Vol. 156, 585-605. (2019)
7. Williams, H. A., Jones, M. H., Nejati, M., Seabright, M. J., Bell, J., Penhall, N. D., et al.: Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. biosystems engineering, Vol. 181, 140-156. (2019)
8. Ouyang, W., Zhou, H., Li, H., Li, Q., Yan, J., Wang, X.: Jointly learning deep features, deformable parts, occlusion and classification for pedestrian detection. IEEE transactions on pattern analysis and machine intelligence, Vol. 40, No. 8, 1874-1887. (2017)
9. Arnold, E., Al-Jarrah, O. Y., Dianati, M., Fallah, S., Oxtoby, D., Mouzakitis, A.: A survey on 3d object detection methods for autonomous driving applications. IEEE Transactions on Intelligent Transportation Systems, Vol. 20, No. 10, 3782-3795. (2019)
10. Mateus, A., Ribeiro, D., Miraldo, P., Nascimento, J. C.: Efficient and robust pedestrian detection using deep learning for human-aware navigation. Robotics and Autonomous Systems, Vol. 113, 23-37. (2019)
11. Ullah, W., Ullah, A., Hussain, T., Muhammad, K., Heidari, A. A., Del Ser, J., et al.: Artificial Intelligence of Things-assisted two-stream neural network for anomaly detection in surveillance Big Video Data. Future Generation Computer Systems, Vol. 129, 286-297. (2022)
12. Dong, K., He, Z., Shen, D., Huang, Z., Chen, X.: Design of Pedestrian Flow Detection System for Playground Entrance/Exit. International Core Journal of Engineering, Vol. 7, No. 9, 332-336. (2021)
13. Capra, C. F.: The Smart City and its citizens: Governance and citizen participation in Amsterdam Smart City. International Journal of E-Planning Research (IJEPR), Vol. 5, No. 1, 20-38. (2016)
14. Muvuna, J., Boutaleb, T., Baker, K. J., & Mickovski, S. B.: A methodology to model integrated smart city system from the information perspective. Smart Cities, Vol. 2, No. 4, 496-511. (2019)
15. Chen, W., Wu, G., Jung, H.: An Optimization Method for Personnel Statistics Based on YOLOv4+ DPAC. Applied Sciences, Vol. 12, No. 17, 8627. (2022)
16. Zhang, Y., Tu, W., Chen, K., Wu, C. H., Li, L., Ip, W. H., Chan, C. Y.: Bus passenger flow statistics algorithm based on deep learning. Multimedia Tools and Applications, Vol. 79, No. 39, 28785-28806. (2020)
17. Xie, P., Li, T., Liu, J., Du, S., Yang, X., Zhang, J.: Urban flow prediction from spatiotemporal data using machine learning: A survey. Information Fusion, Vol. 59, 1-12. (2020)
18. Wu, H., Gao, C., Cui, Y., Wang, R.: Multipoint infrared laser-based detection and tracking for people counting. Neural Computing and Applications, Vol. 29, No. 5, 1405-1416. (2018)
19. Kang, D., Ma, Z., & Chan, A. B.: Beyond counting: Comparisons of density maps for crowd analysis tasks-counting, detection, and tracking. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 29, No. 5, 1408-1422. (2018)
20. Gao, C., Wang, L., Xiao, Y., Zhao, Q., Meng, D.: Infrared small-dim target detection based on Markov random field guided noise modeling. Pattern Recognition, 2018, 76, Vol. 76, 463-475. (2018)

21. Bai, X., Bi, Y.: Derivative entropy-based contrast measure for infrared small-target detection. IEEE Transactions on Geoscience and Remote Sensing, 2018, 56(4), Vol. 56, No. 4, 2452-2466. (2018)
22. Lu, Z. M., Zhu, F. C., Gao, X. Y., Chen, B. C., Gao, Z. G.: In-situ particle segmentation approach based on average background modeling and graph-cut for the monitoring of l-glutamic acid crystallization. Chemometrics and Intelligent Laboratory Systems, Vol. 178, 11-23. (2018)
23. Vishnu, V. C. M., Rajalakshmi, M., Nedunchezhian, R.: Intelligent traffic video surveillance and accident detection system with dynamic traffic signal control. Cluster Computing, Vol. 21, No. 1, 135-147. (2018)
24. Zhang, H. B., Zhang, Y. X., Zhong, B., Lei, Q., Yang, L., Du, J. X., Chen, D. S.: A comprehensive survey of vision-based human action recognition methods. Sensors, Vol. 19, No. 5, 1005. (2019)
25. Luo, F., Guo, W., Yu, Y., Chen, G.: A multi-label classification algorithm based on kernel extreme learning machine. Neurocomputing, Vol. 260, 313-320. (2017)
26. Raj, R. J. S., Shobana, S. J., Pustokhina, I. V., Pustokhin, D. A., Gupta, D., Shankar, K. J. I. A.: Optimal feature selection-based medical image classification using deep learning model in internet of medical things. IEEE Access, Vol. 8, 58006-58017. (2020)
27. Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., Benediktsson, J. A.: Deep learning for hyperspectral image classification: An overview. IEEE Transactions on Geoscience and Remote Sensing, Vol. 57, No. 9, 6690-6709. (2019)
28. Lv, N., Chen, C., Qiu, T., Sangaiah, A. K.: Deep learning and superpixel feature extraction based on contractive autoencoder for change detection in SAR images. IEEE transactions on industrial informatics, Vol. 14, No. 12, 5530-5538. (2018)
29. Xue, W., Jiang, T.: An adaptive algorithm for target recognition using Gaussian mixture models. Measurement, Vol. 124, 233-240. (2018)
30. Zhao, L., Li, S.: Object Detection Algorithm Based on Improved YOLOv3. Electronics, Vol. 9, No. 3, 537. (2020)
31. Pang, L., Liu, H., Chen, Y., Miao, J.: Real-time Concealed Object Detection from Passive Millimeter Wave Images Based on the YOLOv3 Algorithm. Sensors, Vol. 20, No. 6, 1678. (2020)
32. Xu, D., Wu, Y.: Improved YOLO-V3 with DenseNet for Multi-Scale Remote Sensing Target Detection. Sensors, Vol. 20, No. 15, 4276. (2020)
33. Zhao, L., Li, S.: Object detection algorithm based on improved YOLOv3. Electronics, Vol. 9, No. 3, 537. (2020)
34. Wu, Z., Fu, M., Xu, Y., Lu, R.: A distributed Kalman filtering algorithm with fast finite-time convergence for sensor networks. Automatica, Vol. 95, 63-72. (2018)
35. Yang, H., Wang, J., Miao, Y., Yang, Y., Zhao, Z., Wang, Z., et al.: Combining Spatio-Temporal Context and Kalman Filtering for Visual Tracking. Mathematics, Vol. 7, No. 11, 1059. (2019)
36. Jover J M, Kailath T.: A parallel architecture for Kalman filter measurement update and parameter estimation. Automatica, Vol. 22, No. 1, 43-57. (1986)
37. Zhang, Z., Fu, K., Sun, X., Ren, W.: Multiple target tracking based on multiple hypotheses tracking and modified ensemble Kalman filter in multi-sensor fusion. Sensors, Vol. 19, No. 14, 3118. (2019)
38. Huang, M., Guan, W., Fan, Z., Chen, Z., Li, J., Chen, B.: Improved target signal source tracking and extraction method based on outdoor visible light communication using a cam-shift algorithm and kalman filter. Sensors, Vol. 18, No. 12, 4173. (2018)
39. Fang, Y., Yu, L., Fei, S.: An Improved Moving Tracking Algorithm with Multiple Information Fusion Based on 3D Sensors. IEEE Access, Vol. 8, 142295-142302. (2020)
40. Xie, Z., Guan, W., Zheng, J., Zhang, X., Chen, S., Chen, B.: A high-precision, real-time, and robust indoor visible light positioning method based on mean shift algorithm and unscented Kalman filter. Sensors, Vol. 19, No. 5, 1094. (2019)

41. Huang, M., Guan, W., Fan, Z., Chen, Z., Li, J., Chen, B.: Improved target signal source tracking and extraction method based on outdoor visible light communication using a cam-shift algorithm and kalman filter. Sensors, Vol. 18, No. 12, 4173. (2018)
42. Zhang, C., Luo, K., Gu, S., Chen, L., Xia, Z., Gao, J.: Caps-YOLO: Pedestrian detection method of complex posture combined with capsules network. Journal of Flow Visualization and Image Processing, Vol. 28, No. 3. (2021)

Baofeng Yao was born in Bengbu，Anhui, P.R. China, in 1980. He received his B.S. degree in computer science from the Huaibei Normal University, Huaibei, China in 2001, M.S. degree from the Hefei University of Technology, Hefei, China in 2009. Now, he is an Associate Professor with the School of Computer Science and Information Engineering, Bengbu University. His research interest include machine learning, information security and big data analysis. E-mail: ybf@bbc.edu.cn

Shijun Liu was born in 1980 in Wuhe, Bengbu, Anhui, China. He received his B.S.degree in computer science from the Anqing Normal University,Anqing，China in 2004， M.S.degree from the Southeast University，Nanjing, China in 2016.Now， he is a Lecturer with the School of Computer Science and Information Engineering, Bengbu University. His research interest include machine learning, computer vision, privacy protection algorithms and deep learning. E-mail: liushijun@bbc.edu.cn

Lei Wang was born in Suzhou，Anhui, P.R. China, in 1978. He received his B.S. degree in computer science from the Huaibei Normal University, Huaibei, China in 2001, M.S. degree from the Hefei University of Technology, Hefei, China in 2009. Now, he is an Lecturer with the School of Computer Science and Information Engineering, Bengbu University. His research interest include machine learning, information security and big database analysis. E-mail: bbxywl@bbc.edu.cn

Received: August 13, 2022; Accepted: December 25, 2022.