



A hybrid CNN and LSTM-based deep learning model for abnormal behavior detection

Chuan-Wang Chang¹ · Chuan-Yu Chang²  · You-Ying Lin²

Received: 31 March 2021 / Revised: 23 August 2021 / Accepted: 23 December 2021 /
Published online: 17 January 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Traditional cameras can only record videos passively. If the camera can further automatically recognize human behavior and activity, it can immediately issue an alarm to notify the monitor or guards when abnormal behavior is detected. Hence, the monitor or guard can quickly take relevant actions based on the detected behavior. In this paper, we propose a deep learning model for abnormal behavior detection, which use object detection technology YOLOv3 to detect pedestrians, and then use hybrid Deep-SORT algorithm to track pedestrians to obtain tracking trajectories from the sequence frames. Then, the convolutional neural network (CNN) is used to extract the action characteristics of each tracked trajectory, and the long short-term memory network (LSTM) is used to build anomalous behavior identification model to predict abnormal behavior, such as falling, kicking, punching, etc. The experimental results show that the proposed method has a good recognition rate in different behavior data sets, and it can also meet the needs of real-time monitoring.

Keywords Deep SORT · YOLOv3 · Object detection · Object tracking · Abnormal behavior detection

✉ Chuan-Yu Chang
chuanyu@yuntech.edu.tw

Chuan-Wang Chang
cwchang@ncut.edu.tw

You-Ying Lin
m10617018@yuntech.edu.tw

¹ Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, No.57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung 41170, Taiwan

² Department of Computer Science and Information Engineering, National Yunlin University of Science and Technology, No. 123 University Rd., Sec. 3, Douliou, Yunlin 64002, Taiwan

1 Introduction

Population aging is a common problem faced by most countries today. Many elderly people live alone. Among them, falls are the most common dangerous behavior and may require emergency medical care. An automated fall detection system can help these elderly people and provide necessary assistance in a timely manner.

According to a report from the Center for Injury Research and Prevention, middle-aged and elderly people often suffer serious injuries due to falls. Many researchers have invested in the research and development of fall detection systems. Various fall detection methods can be divided into wearable and non-wearable methods. Wearables system usually include placing an accelerometer on the object, which can detect changes in acceleration, planes of motion, or impact to detect falls [9, 10, 27]. Huang et al. [25] used wearable devices (three-axis accelerometers) and KGA algorithms and genetic algorithms to identify falls that are harmful to the elderly.

However, most systems using wearable sensors will cause inconvenience or discomfort to the user, and must operate when the user is wearing it. Non-wearable systems use cameras [4, 5], acoustic sensors [31, 40], and pressure sensors [2], which are placed in the normal environment of the subject and use various measurements to determine whether the subject has fallen. Arifoglu et al. [3] used daily activity data obtained by various environmental sensors and used RNN to identify abnormal behaviors. Hu [23] proposed an improved spatial-temporal convolution neural network to fight, rob, faint, and punch. The average accuracy of the method is about 70%.

The use of artificial vision to automatically detect falls is a special case of human activity recognition and is very useful for the elderly. The fall event itself, which is automatically extracted from the video scene, represents vital information and can be used to alert emergency situations.

Although we and many researchers have proposed many methods that can detect falls in real time from surveillance videos, we have also noticed that sudden violence or injury incidents on the street have caused more and more people to pay attention to the safety of public spaces. This means that it is becoming more and more important to enhance the technology of falling detection into a video surveillance system that can recognize a variety of dangerous behaviors. Traditional surveillance cameras only provide passive image capture, storage, and playback functions to record the occurrence of events. When an abnormal event occurs, it is difficult to provide immediate warning. In particular, traditional monitoring methods mainly rely on manual labor. When the mental stress and fatigue accumulated by looking at multiple monitoring screens for a long time will be distracted and ignore the occurrence of accidents. Therefore, if the ubiquitous cameras can be combined with our intelligent surveillance technology, it will greatly increase the value of the video surveillance system.

In order to identify people (object) from videos, object detection is a hot research topic in computer vision and image processing in recent years. These methods can be divided into two categories: region proposal based and end-to-end based [55]. The R-CNN [17, 47], Fast R-CNN [16], and Faster R-CNN [43] are all object detection methods based on the proposal region. The well-known YOLO [42], YOLOv2 [26] and YOLOv3 [41] are based on end-to-end methods. Although the method based on proposal regions has higher accuracy, the recognition speed is slower, while the end-to-end method has advantages in recognition speed and can meet the real-time requirements.

This paper uses image processing and deep learning technology to analyze pedestrian activity status through camera images and identify abnormal behaviors (falls, kicks, punches) in the pictures. Since abnormal behavior is usually irregular and happens in a short period of time, it requires continuous monitoring. Therefore, the purpose of this paper is to develop a surveillance system that can instantly detect abnormal behavior. The system recognizes and judges the activity status of pedestrians in the surveillance images in real time, and realizes active detection of abnormal events. This system can effectively detect abnormal behavior events or potential hazards to reinforce the limitations of traditional monitoring systems that require labor attention and reduce the mental load of security staff.

In order to avoid misjudgment, when a certain number of detected abnormal behaviors reaches a preset threshold, the monitoring system will trigger a warning mechanism and send a message to the monitor to ensure instant reminder and response to avoid more serious consequences.

In this paper, a hybrid CNN and LSTM-based deep learning model for abnormal behavior detection is proposed. The main contributions of this paper are concluded as follows:

1. The proposed abnormal behavior detection model can be applied to video surveillance system of homes and public spaces to alert the occurrence of abnormal events in real time.
2. We combine object detection technologies, such as YOLOv3, Kalman filter and Hungarian algorithm, and propose a hybrid DeepSORT algorithm to track pedestrians to obtain tracking trajectories.
3. Under the condition of non-high-end hardware, abnormal behavior in surveillance video can be detected within 0.3 s. The method we propose can achieve the practicality of real-time detection.

The remainder of this paper is organized as follows. Various abnormal behavior detection methods and techniques are reviewed in Section 2. Section 3 presents the details of the proposed scheme. In Section 4, we present the experiment results to demonstrate the performance of the proposed scheme. Section 5 draws some conclusions and future work.

2 Related works

With the increase of elderly people living alone and the continuous opening up of society, people are increasingly concerned about the safety of their homes and the public environment. Because of the rapid development of computer vision technology, the research of human behavior analysis has made great progress. In particular, the analysis of abnormal human behavior combined with object detection and gesture recognition technology has become one of the most important technologies in the field of safety management.

Many scholars have conducted a lot of research and experiments on the analysis/detection technology of human abnormal behavior in video surveillance. These methods can be roughly divided into two types, namely model-based behavior detection and scene density & object interaction-based behavior detection [33, 56].

2.1 Model-based behavior detection

Model-based behavior analysis methods need to set abnormal behavior judgment conditions, extract the motion feature information of moving targets in the video sequence, compare these information with the judgment conditions, and establish a normal behavior model. The classification method can be divided into three categories: supervised, semi-supervised and unsupervised [33].

Supervised approaches aim to model normal and abnormal behavior through labeled data. Several literatures have been proposed to detect specific events in the video. They are usually used to detect specific abnormal behaviors predefined in the training phase, such as fall detection [45], fight detection [7, 19], and wandering detection [18].

Semi supervised approaches need only normal video data for training and can be divided into rule based and model based approaches. The rule-based semi supervised approach aims to develop a rule using normal patterns. Then, any sample that does not fit this rule is considered as an abnormal. For example, Tani et al. [46] used rules obtained through ontology-based approach to detect abnormal events in video surveillance. Nguyen et al. [36] proposed a fall detection system based on rules extracted from shape features. In the model-based semi supervised approach, abnormal patterns correspond to instances that deviate from the model representing the normal behaviors. Hidden Markov model (HMM) and Gaussian Mixture Model (GMM) are the most used models. Nannan et al. [35] proposed a method for anomaly detection using Gaussian process. First, use HOF to extract low-level features to describe pattern motion. Then, a Gaussian process model is built to generate a normal behavior distribution, and this model is used to detect anomalies in the video. In Feng et al. [15], a deep GMM is used to learn normal patterns.

Unsupervised approaches aim to learn normal and abnormal behaviors from statistical features extracted from unlabeled data. For example, Weiya et al. [50] proposed an unsupervised kernel framework for anomaly detection based on feature space and support vector data description (SVDD).

2.2 Scene density & object interaction-based behavior detection

The density of the scene indicates the number of people present in the surveillance video. In different application scenarios, the number of people monitored in the scene is different, and the behavior detection methods used are also different. The choice of technology used to characterize behavior is directly affected by scene density. Therefore, we divide the scenes into two categories: non-crowded scenes, which are characterized by the presence of one or several people in the monitored video at the same time and crowded scenes, which contain many people.

Fall detection, loitering detection [18], detection of violent incident in elevator [57] and the detection of entering restricted areas (being in the wrong placed) [48] are application scenarios that belong to non-crowded scenarios. Van Beeck et al. [48] proposed a framework to detect abnormal behaviors in LWIR surveillance images of train platforms. The framework consists of two consecutive stages: first, reliable pedestrian detection is performed, and then the detection path is classified using a Markov model.

Based on human dynamics, Zhang et al. [54] proposed a human posture representation model called the “five-point inverted pendulum model”, and used an improved two-branch

multi-stage convolutional neural network (M-CNN) for fall detection. Pan et al. [39] proposed a fighting detection method based on pose estimation and motion optical flow.

In crowded scenes, trajectory descriptor can be obtained through tracking algorithm, which can accurately describe the movement of objects in the scene. Laxhammar et al. [30] and Bera et al. [6] proposed useful trajectory modeling methods to learn normal trajectory patterns and identify abnormal behaviors with abnormal trajectories.

In recent years, many methods have been proposed to detect abnormal behavior in crowded scenes. For example, method based on hidden Markov models [29], based on social force models [34], based on chaotic invariants [53], based on kinematic features [1], or based on streakline [49].

Chaker et al. [12] introduced an unsupervised framework based on social network model to capture the crowd interaction and the scene dynamics. Hu et al. [24] proposed a weakly supervised framework for Abnormal Behavior Detection and Localization (ABDL) in the scenes. First, the objects in the scene such as pedestrians, vehicles, etc. are detected using the Faster Regional Convolutional Neural Network (Faster R-CNN); then, the object behavior is described by a Histogram of Large Scale Optical Flow (HLSOF) descriptor; finally, the Multiple Instance Support Vector Machine (MISVM) is trained and then used to identify the testing behaviors as normal or abnormal.

3 Materials and methods

In order to detect abnormal behaviors in the scene, in this paper, we propose a hybrid behavior detection model based on deep learning technology. First, we acquire a streaming image of the surveillance scene and convert the streaming image into continuous picture frames. We use object detection technology to detect pedestrians in the picture. Since abnormal behavior is usually irregular and instantaneous, it is necessary to track each detected target. After tracking, we will obtain the trajectory of each target and then use CNN to extract motion features. Then input the features of these trajectories into the two-layer LSTM model to obtain the final prediction results of abnormal behavior. Figure 1 shows the overall system flow chart. The important technologies used in this system will be introduced below, including object detection, object tracking, and behavior recognition models.

3.1 Object detection

The main purpose of object detection is to find human objects in images. Because there is a lot of unwanted background information in the image. Therefore, to detect each object in a complex background, it is necessary to filter out unnecessary background noise and preserve the complete outline of the object, then extract the necessary object pose, and further provide a basis for behavior analysis. The object detection method used in this paper is YOLOv3 [41]. YOLO (You Only Look Once) is a one-stage object detection algorithm based on regression method, which achieves a complete single training to share features under the premise of ensuring a certain accuracy. The main reason why we choose YOLOv3 as the object detection method is that in an open environment with a complex background, YOLOv3 can detect various angles and multiple targets in real time while maintaining good detection results.

YOLO Darknet-53 consists of 53 convolutional layers and each convolutional layer is followed by a layer of batch normalization (BN) and leaky rectified linear units (ReLU). As

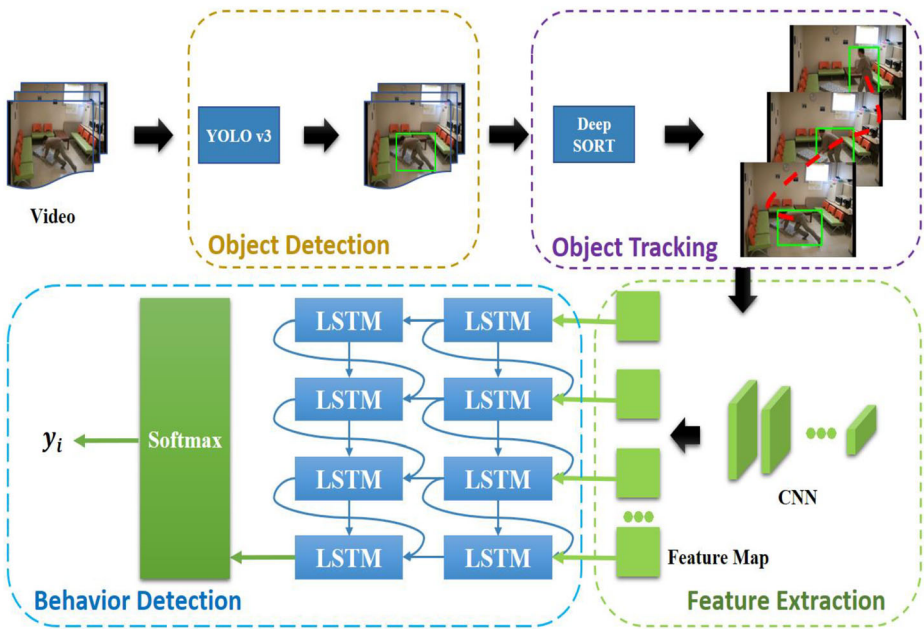


Fig. 1 System flow chart

shown in Fig. 2, YOLOv3 uses the Eqs. (1) ~ (4) for position prediction, where (t_x, t_y, t_w, t_h) denotes the predicted output of the model, (c_x, c_y) denotes the coordinate offset of the grid cell, (ρ_w, ρ_h) denotes the side length of the preset anchor box, σ is activation function, and (b_x, b_y, b_w, b_h) is the center coordinate and width as well as height of the final predicted bounding box.

$$b_x = \sigma(t_x) + c_x \tag{1}$$

$$b_y = \sigma(t_y) + c_y \tag{2}$$

$$b_w = \rho_w e^{t_w} \tag{3}$$

$$b_h = \rho_h e^{t_h} \tag{4}$$

In some complex scenarios, an object may belong to multiple categories. YOLO v3 changes the classification prediction function from *softmax* to multiple independent logical classifiers to solve the multi-label classification problem. YOLOv3 refers to the upsampling and fusion practice of the FPN [32] network, and combines three scale feature maps to independently detect the fusion feature maps on multiple scales, thereby improving the detection of small objects. The feature extraction model of YOLOv3 introduces the architecture of the residual network [21] to remove the fully connected layer in a fully convolutional manner, so that the network becomes deeper, and therefore more semantic information can be obtained, and it can also solve Gradient problem and get better detection accuracy.

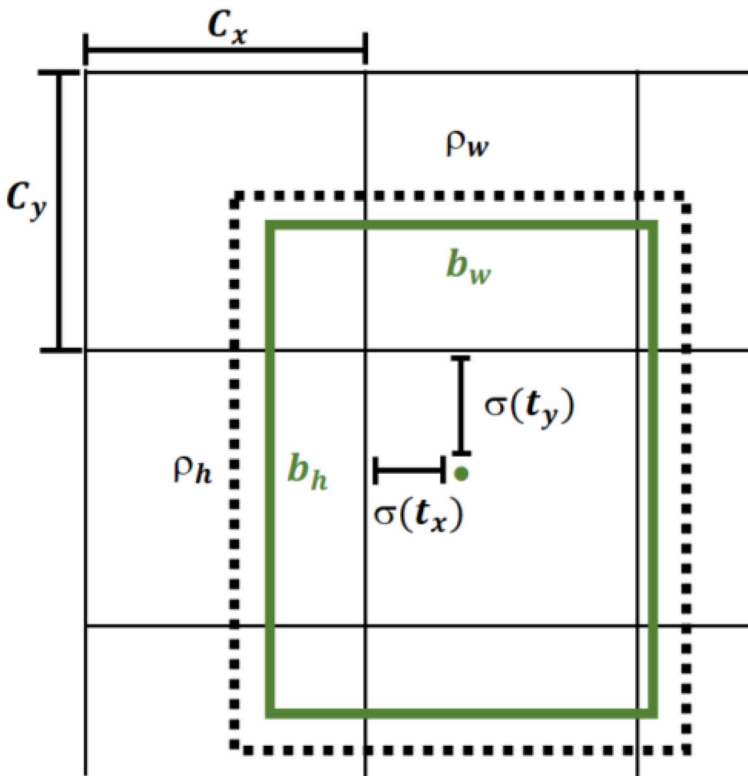


Fig. 2 Bounding boxes with dimension priors and location prediction [9]

In this work, for each bounding box detected by YOLO, the position information of all detected targets stored in an eight-dimensional matrix is extracted and used to represent the current state of the target. The matrix is given by

$$P = (u, v, \gamma, h, \dot{x}, \dot{y}, \dot{\gamma}, \dot{h}) \tag{5}$$

where (u, v) is the position of the bounding box, γ is the aspect ratio of the bounding box, h is the height of the bounding box, and $(\dot{x}, \dot{y}, \dot{\gamma}, \dot{h})$ is the corresponding velocity information in the coordinate, which is $(0,0,0,0)$ initially.

3.2 Object tracking

Abnormal events often happen unexpectedly and suddenly. Therefore, when monitoring the screen, it is important to understand the changes in the behavior of the people in the video. The main task of object tracking is to detect, extract, identify and track moving objects in continuous image sequences. And further obtain the position relationship and movement trajectory of the moving object to realize the behavior analysis of the moving object. The SORT (Simple Online and Realtime Tracking) algorithm [8] proposed by Bewley et al. is a real-time multi-target tracking algorithm, which has high accuracy during real-time detection.

However, when the target is occluded for a long time or cannot be detected, the tracking trajectory of the target will be interrupted and a new trajectory will be formed, which leads to

the problem of frequent ID switching. Deep-SORT [52] adds depth features and combines the target's motion information with appearance information as a new measure, which increases the robustness of the algorithm and greatly reduces ID switching.

In the Deep-SORT algorithm, an 8-dimensional matrix $(u, v, \gamma, h, \dot{x}, \dot{y}, \dot{\gamma}, \dot{h})$ generated by YOLOv3 is first used to record the state of the trajectory at a certain moment, which is expressed as the center position of the bounding box (u, v) , the ratio of width to height (γ, h) , and the speed corresponding to the image coordinates (u, v, γ, h) . Then we use the Kalman filter's [28, 51] equidistant motion and linear observation model to predict and update the trajectory-

Kalman filter is a regression algorithm for optimizing discrete data, as long as the estimated value of the state at the previous moment and the observation value of the current state are known, and a series of measured values over time are used. Although this measurement is not accurate and contains statistical errors, an estimate of the current state can be calculated. Therefore, it is very suitable for target tracking. The Kalman filter mainly includes two processes, *prediction* and *correction*, as shown in Eqs. (6)~(7) and (8)~(10), respectively.

Time Update (Predict):

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (6)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (7)$$

Measurement Update (Correct):

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (8)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (9)$$

$$P_k = (I - K_k H)P_k^- \quad (10)$$

where x_k is the system state at time k , \hat{x}_k is an a posteriori estimate of the state at step k , \hat{x}_k^- is an a priori estimate of the state at step k , \hat{x}_{k-1} is the predicted result from the previous state, z_k is the actual measurement of x at time k , P_k^- is the a priori estimate error covariance, P_k is the a posteriori estimate error covariance, K_k is the Kalman gain, H is the noiseless connection between the state vector and the measurement vector, $H\hat{x}_k^-$ is the measurement prediction. Q is the process noise covariance, R is the measurement noise covariance.

The material meaning of Eqs. (6) and (7) is that when the target moves, the target frame position and speed of the current frame and other parameters are predicted through the target frame and speed of the previous frame.

In this work, the state of the current target generated by YOLOv3 is fed into the Kalman filter, and the target is predicted and updated. The prediction result is stored in the tracked list. The result predicted by the Kalman filter is set to the tentative state, which to be further observed; or a confirmed state, which has been matched successfully; or a deleted state, which cannot be matched and should be deleted. The position information of the target set to the confirmed state is stored in the track matrix.

Then, we combine the motion information and appearance information, and use the Hungarian algorithm [11] to solve the matching problem between the predicted state and the current detection state.

For position measurement, we calculate the Mahalanobis distance [52] between the position predicted by the Kalman filter and the position of the detection frame detected by YOLOv3. The formula is as shown in Eq. (11)

$$d^{(1)}(i, j) = (d_j - y_i)^T S_i^{-1} (d_j - y_i) \quad (11)$$

where d_j represents the position of the j -th detection frame, y_i is the predicted position of the target by the i -th tracker, and S_i^{-1} represents the covariance matrix between the detection position and the tracking position.

As in Eq. (12), we calculate the minimum cosine distance between the i -th predicted track and the j -th trajectory, and use a pre-trained CNN network to extract appearance features.

$$d^{(2)}(i, j) = \min \left\{ 1 - r_j^T r_k^{(i)} \mid r_k^{(i)} \in R_i \right\} \quad (12)$$

The Mahalanobis distance is very useful for short-term prediction and matching based on the altruistic position of the moving target. However, for a target that is occluded for a long time, the cosine distance takes into account the appearance information, which can effectively restore the target's identity. In order to solve the problem of establishing associations, we combine two metrics to obtain the final decision message:

$$c_{i,j} = \lambda d^{(1)}(i, j) + (1 - \lambda) d^{(2)}(i, j) \quad (13)$$

For each newly predicted frame position, position division is performed and different thresholds are set for different position areas. Then, compare with the positions of all predicted boxes that have appeared, and calculate the distance between the newly appearing box and the previously appearing boxes. When the Euclidean distance to a box that appeared before is shorter than the set threshold, it will be regarded as the next position.

Because when a target is occluded for a long time, the uncertainty of Kalman filter prediction will increase greatly. We further use matching cascade strategy [52] to improve matching accuracy. Especially, the blocked target can be retrieved again, thereby reducing the number of ID Switches. We divide the objects into different levels according to the time the object is occluded. The less the occlusion time, the higher the matching level. First, the tracking block set and the prediction block set are obtained, and A_{max} is set to the maximum number of the block allowed to be missed in the track. The cost matrix is obtained by the weighted sum of the two measurement indicators of position and appearance, and then through the matching cascade condition, thresholds are set to calculate the position and appearance information respectively.

We summarize the object detection and object tracking process used in this article as the hybrid DeepSORT algorithm in Fig. 3 and the detailed flow chart in Fig. 4.

3.3 Behavior recognition model

The purpose of behavior recognition is to locate the target's position on the received continuous images and determine the current behavior. There are dependencies between action

Algorithm: Hybrid DeepSORT

Input: The image frames captured by the camera.

Output: The processed image frames that the predicted target can be matched with the detected target.

Step1. Use *YOLOv3* to detect the target in the video frame, and save the position and ID of the detection bounding box.

Step2. Use the *Kalman filter* to predict the position of the targets, and save the position and the ID of the predicted bounding boxes.

Step3. Use the *Mahalanobis distance* and the *Hungarian algorithm* to handle the matching problem between the predicted position and the current detection position.

Step4. Use the *Matching cascade* strategy to improve the matching accuracy.

Fig. 3 The hybrid DeepSORT algorithm

sequences. Therefore, we must consider not only the characteristics of the skin texture in the spatial domain, but also the correlation between frames in the time domain.

3.3.1 Behavior classification

When pedestrian motion is to be captured from a continuous image sequence, there may be many repetitive or slightly different motion frames. Since most abnormal behaviors occur suddenly and have large changes in body movements, continuous frames with very small differences can be regarded as redundant frames. Therefore, it is necessary to give these consecutive motion frames a posture name as a representative action of abnormal behavior, and remove the redundant frames. In this paper, we extract a sequence of segments from 8 strides, and select 5 frames from the sequence as the action sequence representing abnormal behavior, as shown in Fig. 5.

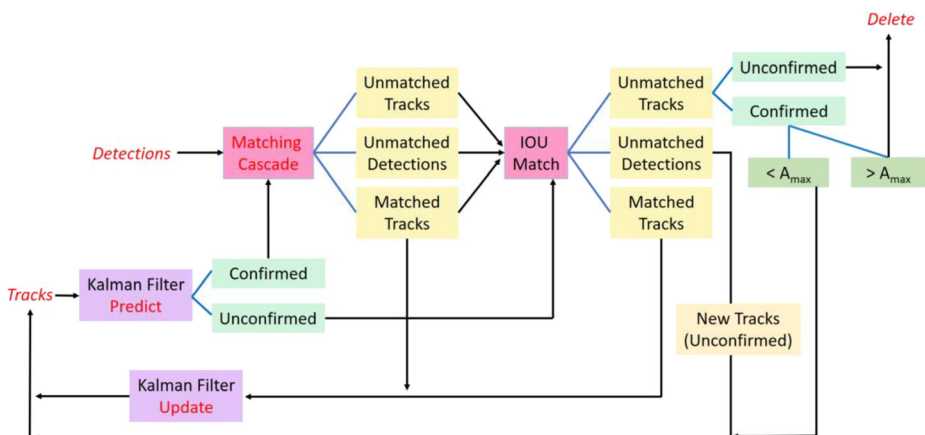


Fig. 4 The process flow of the hybrid DeepSORT algorithm

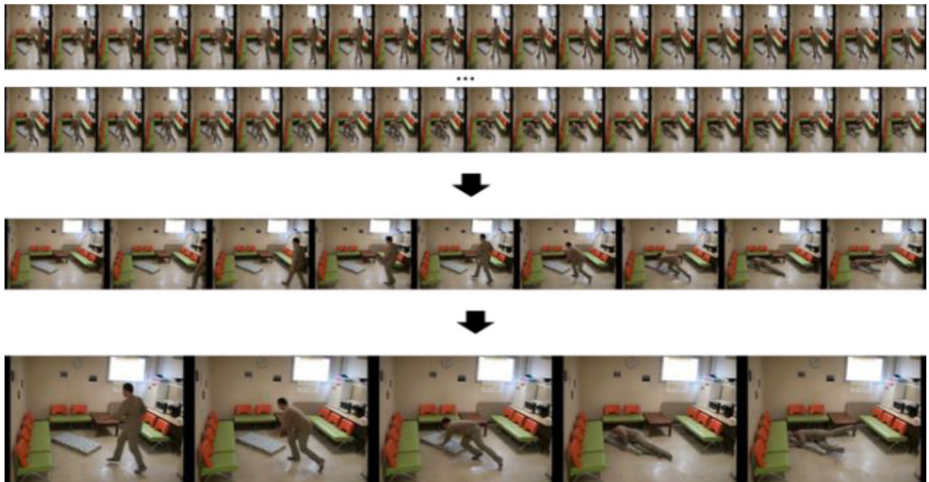


Fig. 5 Frame selection for abnormal behavior

Because CNN has a strong ability to learn image features, we input the action sequence into the CNN network with shared weights, and analyze the spatial features and local motion features. Therefore, in this paper, we use CNN to construct a feature detector to find obvious action features in abnormal behavior. Because the position and posture of each pedestrian in the image are different, the size of the pedestrian image captured by the detection frame is different, so we normalized these pedestrian images and adjusted them to 80×80 size, as shown in Fig. 6.

Due to the sequential nature (time series) of the abnormal behavior, we hope that the network can retain a certain memory when it continuously reads the input frames. When inputting a new frame, the network model will take the newly observed information into consideration for adjusting the memory, and then adjust the memory. That is, the internal

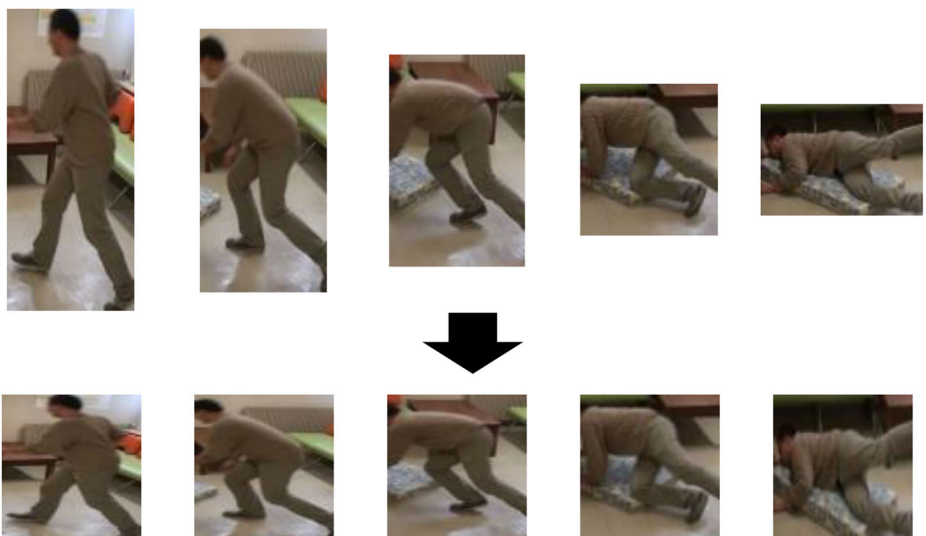


Fig. 6 Behavioral sequence normalization

memory can retain the key information. The Long Short-Term Memory model (LSTM) [22, 38] has long-term memory capabilities and is very suitable for processing and predicting important events with very long intervals and delays in a time series. Its main feature is to add input gate, output gate, forget gate and memory cell in the hidden layer to do more complex processing of the message.

The core of LSTM is the cell state C_t , which runs through the entire network operation cycle. It holds some key information learned over time. As shown in Fig. 7, the three gates in the LSTM model all use *Sigmoid* function as the activation function. Its purpose is to allow the output value to be in the range of $0 \sim 1$ to indicate the degree of message filtering. When the value is equal to 0, it means that no messages are allowed to pass, while the value equal to 1 means that all messages are allowed to pass. It is first necessary to determine which messages should be discarded (forgotten) from the cell state, which is controlled by the sigmoid layer of the forget gate. Such a mechanism can avoid the occurrence of gradient explosions. Its input includes the output from the CNN network x_t and the previous hidden state h_{t-1} . Then its output is multiplied by the previous memory state C_{t-1} to determine the forgotten information and form part of the new memory cell. Then, determine which new messages need to be retained through the input gate, and then multiply it by the possible new candidate values \hat{C}_t to form the second part of the new memory cell. Then, the memory states of the previous two parts are added together to form a new cell state. Finally, the new cell state is multiplied by the output gate to determine the degree of output. The reason for using the *tanh* function in the output gate is to set the output value between -1 and 1 , and -1 means that long-term memory will be removed.

Suppose $x_t \in R^d \times 1$ is the input vector, $W_* \in R^p \times p$ is the weighted matrix from the hidden layer to the specific gate, $U_* \in R^p \times d$ is the weighted matrix of the input vector to the specific gate, $b_* \in R^p \times 1$ is the bias of the specific gate, the forward propagation process of forget gate, input gate and output gate can be expressed as Eqs. (14)–(19).

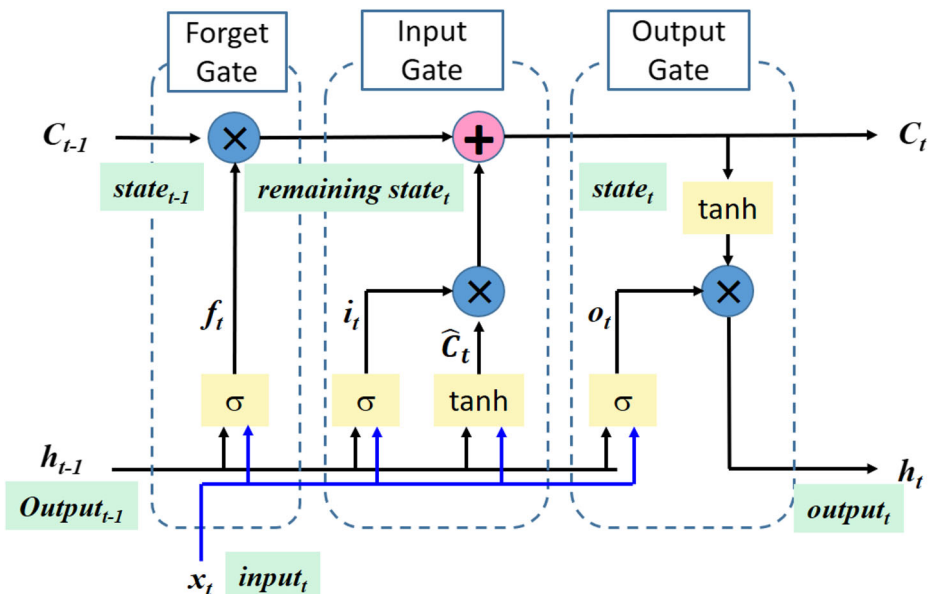


Fig. 7 The LSTM model

$$f_t = \sigma(W_f h_{t-1} + U_f X_t + b_f) \quad (14)$$

$$i_t = \sigma(W_i h_{t-1} + U_i X_t + b_i) \quad (15)$$

$$C_t = \tanh(W_c h_{t-1} + U_c X_t + b_c) \quad (16)$$

$$C_t = f_t \times C_{t-1} + i_t \times \hat{C}_t \quad (17)$$

$$O_t = \sigma(W_o h_{t-1} + U_o X_t + b_o) \quad (18)$$

$$h_t = O_t \times \tanh(C_t) \quad (19)$$

3.3.2 Model architecture

The proposed architecture for abnormal behavior detection is based on the VGG-16 network, which contains 10 convolution layers to extract features. There are 5 pooling layers interspersed between the convolutional layers, as shown in Table 1. The first layer of convolutional layers uses 7×7 convolution kernel to preserve the action space features, and most of the remaining convolution layer are replaced by 3×3 , which can obtain more subtle features to enhance the network recognition ability and reduces the number of parameters. As the network deepens, the amount of computation will increase. In order to avoid this problem, we reduce the number of input channels before using a larger convolution kernel, such as 7×7 . The number of channels in the first layer is 32. After each layer of pooling, the number of channels will double, up to 512.

After completing the feature extraction of the convolution operation, a feature vector of length 5 will be obtained. We convert it into a one-dimensional vector and input it into the LSTM network [11, 28]. Our proposed classification model uses two layers of LSTM. The first layer LSTM will send the results to the next layer and the next self. The purpose of this is that we want to input the output information of each time to the next layer of LSTM as training data, and at the same to the next layer of LSTM as training data, and at the same time pass the information to the next self as input data. Finally, it passes through the fully connected layer and obtains the probability of each behavior category through *softmax*, and the highest score is used as the recognition result.

4 Experimental and discussion

4.1 Training dataset

In this work, we use Fall Detection Dataset [13] as the training data set. This dataset was released by ImVia Laboratory of University of Bourgogne. The videos of this dataset were recorded using a single camera in a home and office environment. The scenes in the video include different light sources, object occlusions, and backgrounds with complex textures. The

Table 1 The proposed architecture for abnormal behavior detection

	Type	Size/stride
	Input	
1	Convolutional	7×7/2
2	Convolutional	3×3/1
3	Maxpool	2×2/2
4	Convolutional	3×3/1
5	Convolutional	3×3/1
6	Maxpool	2×2/2
7	Convolutional	3×3/1
8	Convolutional	3×3/1
9	Maxpool	2×2/2
10	Convolutional	3×3/1
11	Convolutional	3×3/1
12	Maxpool	2×2/2
13	Convolutional	3×3/1
14	Convolutional	3×3/1
15	Maxpool	2×2/2
16	Flatten	
17	LSTM	
18	LSTM	
	Detection	

actors make various falling postures in different walking directions. The dataset contains 191 videos, including 143 videos with falling and 48 videos without falling. The frame rate is 25 frames/s and the resolution is 320×240 pixels.

4.2 Metrics criteria

The evaluation metrics, such as *sensitivity*, *specificity*, *accuracy* and *precision*, are used for assessing the performance of the proposed hybrid model. These metrics are computed by the following equations:

$$\textit{Sensitivity} = \frac{TP}{TP + FN} \quad (20)$$

$$\textit{Specificity} = \frac{TN}{TN + FP} \quad (21)$$

$$\textit{Accuracy} = \frac{TP + TN}{P + N} \quad (22)$$

$$\textit{Precision} = \frac{TP}{TP + FP} \quad (23)$$

where TP stands for true positive, It represents the number of videos with actual falling images that were correctly classified as falls and FP stands for false positive, it indicates the number of non-fall images that are falsely detected as falls. True negative (TN) is the number of the non-

Table 2 Comparison of methods on Fall Detection Dataset [22]

Methods	Sensitivity (%)	Specificity (%)	Accuracy (%)	Precision (%)
Nunezez-Marcos et al. [37]	99.0	97.0	97.0	–
Fan et al. [14]	98.4	100	–	–
Harrou et al. [20]	100	95.0	96.7	94.0
Ours	98.6	97.2	97.4	97.3

fall images that were detected as non-fall. False negative (FN) is the number of video with actual falling images that were falsely detected as non-fall. P and N is the total number of actual fall image and actual negative image. These metrics can help us to classify a small number of samples that are of interest, and solve the number of fall samples is far less than the number of samples without falls.

4.3 Experimental results

The experimental hardware and software environment is set to: Intel(R) Core i7-8700K, DDR4 32GB, NVIDIA GeForce 1080Ti, Ubuntu 16.04, 64-bit operating system.

This experiment used 100 fall films and 40 normal active films as the training set, and the remaining 43 falling films and 8 normal activity films as the test set. The optimizer used in the experiment is stochastic gradient descent (SGD), and set the initial learning rate to 0.01, batch size to 35, and reduce 1/10 after every 10 epochs, and terminate the training process after 50 epochs.

The proposed hybrid model has been compared with various classifier including

Nunezez-Marcos et al. [37], Fan et al. [14], and Harrou et al. [20] Table 2 presents a comparison of our method with other researchers' methods. Although the accuracy of our proposed method in the Fall Detection Data set is less than Fan et al. [14] and less sensitive than Harrou et al. [20], it still has a sensitivity of 98.6%, a specificity of 97.2% and an accuracy of 97.4%. The result indicates that our method is competitive with the existing state-of-the-art methods.

We also use the UT interaction dataset [44] to evaluate the application of the model in complex human activities categories. Experimental results in Table 3 show that our proposed model can also achieve the results of the method proposed by Ryoo et al. [44]. This also illustrates the feasibility of our method applied to complex human activities recognition. From the experimental results, the model we proposed has a low accuracy in identifying kicks and punches. This phenomenon may be because the size of the data set is too small, there are only 20 videos in one category and the resolution of the movie is too low, or it may be because the

Table 3 Comparison of methods on the UT-interaction dataset [44]

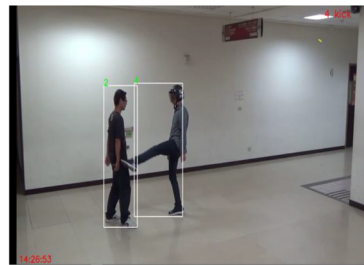
Method	Shake hand (%)	Hugging (%)	Kicking (%)	Pointing (%)	Punching (%)	Pushing (%)
Ryoo et al. [44]	Sensitivity 89	86	95	52	89	78
	Precision 92	88	97	77	65	84
Ours	Sensitivity 92	94	65	88	67	82
	Precision 93	86	82	80	78	94

Table 4 The verification results of the 10 films

Video #	Accuracy (%)	Recognition time(sec)
A	100.0%	0.283
B	66.6%	0.263
C	100.0%	0.262
D	100.0%	0.286
E	100.0%	0.283
F	66.6%	0.279
G	66.6%	0.198
H	100.0%	0.183
I	100.0%	0.284
J	33.3%	0.281
	83.31%	0.260



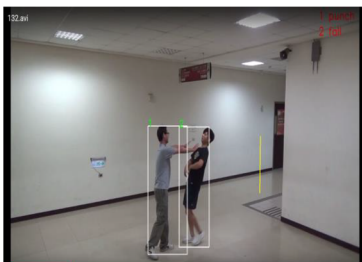
(a)Correct – object1 is falling



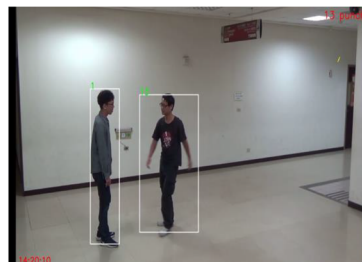
(b)Correct – object4 is kicking



(c)Correct – object2 is punching



(c) Misjudgment – object1 is punching and object2 is falling



(d) Misjudgment – object13 is punching

Fig. 8 The behavior recognition results in different scenarios

body movements of kicking and punching are not obvious or blocked. This problem can be improved by adding high-resolution image data sets.

In addition, In order to verify whether our proposed method can be applied to the actual field, this work also recorded 10 videos containing three abnormal behavior scenarios (falling, kicking, and punching). The length of each video is about 1 min. Table 4 lists the verification results of these 10 films. The average accuracy of this system is 83.31%, especially the average time from detecting abnormal behavior to making a judgment is only 0.26 s, which proves that the method we proposed can achieve real-time monitoring.

Figure 8 shows the behavior recognition results in different scenarios, including correct and incorrect detection results. We further inspected the video that caused the misjudgment and found that when the body is too tilted or the arm swings greatly, it is easy to cause misjudgment. It is inferred that the reason may be due to the inability to extract a complete object detection frame under certain conditions (such as, light, similar background, and overlapping targets, etc.) in the object detection stage. In the future, the models in the object detection and object tracking stages will be adjusted and optimized to strengthen the robustness of identification under various environmental noises.

5 Conclusion

This paper proposes an abnormal behavior detection method based on CNN and LSTM, which can detect three types of abnormal behavior: falling, kicking or punching. Experimental results show that the proposed method can effectively detect abnormal behaviors in surveillance images.

The detection accuracy rate of using the Fall Detection Dataset to verify is 97.4%, and the accuracy rate of using the UT interaction data set to verify the identification of various behaviors exceeds 80%. Moreover, the accuracy of the method used in real scene recognition reaches 83.31%, and it can achieve real-time detection efficiency. Although this paper focuses on abnormal behavior detection, the proposed method can also be applied to other behavior recognition tasks as long as the training data set is replaced.

Acknowledgments This work was supported by the Ministry of Science and Technology, Taiwan, under the grants MOST 109-2637-E-167-004 -.

References

1. Aliet S, Shah M (2010) Human action recognition in videos using kinematic features and multiple instance learning. *IEEE Trans Pattern Anal Mach Intell* 32(2):288–303
2. Alwan M et al (2006) A smart and passive floor-vibration based fall detector for elderly. In: 2nd International Conference on Information and Communication Technologies
3. Arifoglu D, Bouchach A (2017) Activity recognition and abnormal behaviour detection with recurrent neural networks. In: *Proc. of MobiSPC*. pp 86–93
4. Belshaw M, Taati B, Giesbercht D, Mihailidis A (2011) Intelligent vision-based fall detection system: preliminary results from a real world deployment. *RESNA/ICTA 2011: Advancing Rehabilitation Technologies for an Aging Society*

5. Belshaw M, Taati B, Snoek J, Mihailidis A (2011) Towards a single sensor passive solution for automated fall detection. *Conf Proc IEEE Eng Med Biol Soc*. pp 1773–1776
6. Bera A, Kim S, Manocha D (2016) Realtime anomaly detection using trajectory-wise crowd behavior learning. *Proceedings of International Conference on Computer Vision and Pattern Recognition, CVPRW'13*. pp 50–57
7. Bermejo NE, Deniz SO, Bueno GG, Rahul S (2011) Violence detection in video using computer vision techniques. In: *Proc. of the 14th international conference on computer analysis of images and patterns, CAIP2011*, pp 332–339
8. Bewley A, Ge Z, Ott L, Ramos F, Uppcroft B (2016) Simple online and realtime tracking. In: *Proc. of IEEE International Conference on Image Processing*
9. Bourke AK, O'Brien JV, Lyons GM (2007) Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. *Gait Posture* 26(2):194–199
10. Bourke AK, van de Ven PW, Chaya AE, O'Laighin GM, Nelson J (2008) Testing of a long-term fall detection system incorporated into a custom vest for the elderly. *Proc. Of IEEE Conf. on Eng Med Biol Soc*. pp 2844–2847
11. Burkard R, Dell'Amico M, Martello S (2012) Assignment problems. Revised reprint. SIAM - Society of Industrial and Applied Mathematics, Philadelphia
12. Chaker R, Aghbari ZA, Junejo IN (2017) Social network model for crowd anomaly detection and localization. *Pattern Recogn* 61:266–281
13. Charfi I, Miteran J, Dubois J, Atri M, Tourki R (2013) Optimised spatio-temporal descriptors for real-time fall detection: comparison of SVM and Adaboost based classification. *J Electron Imaging* 22(4):17
14. Fan Y, Levine MD, Wen G, Qiu S (2017) A deep neural network for real-time detection of falling humans in naturally occurring scenes. *Neurocomputing* 260:43–58
15. Feng Y, Yuan Y, Lu X (2017) Learning deep event models for crowd anomaly detection. *Neurocomputing* 219:548–556
16. Girshick R (2015) Fast R-CNN. In: *The IEEE International Conference on Computer Vision, Santiago, Chile*. pp 1440–1448
17. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA*, pp 580–587.
18. Gomez HF, Martinez TR, Arias TS, Fernandez CA, Sylvie R, Gonzalez EA et al (2015) Identification of loitering human behaviour in video surveillance environments. *Artif Comput Biol Med* 9107:516–525
19. Guang S, Fu G, Li P, Geng H (2014) Violent behavior detection based on SVM in the elevator. *Int J Secur Appl* 8:31–40
20. Harrou F, Zerrouki N, Sun Y, Houacine A (2017) Vision-based fall detection system for improving safety of elderly people. *IEEE Instrum Meas Mag* 20:19–55
21. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *Proc of IEEE Conference on Computer Vision and Pattern Recognition*. pp 770–778
22. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780
23. Hu Y (2020) Design and implementation of abnormal behavior detection based on deep intelligent analysis algorithms in massive video surveillance. *J Grid Comput* 18:227–237
24. Hu X, Dai J et al (2019) A weakly supervised framework for abnormal behavior detection and localization in crowded scenes. *Neurocomputing* 383:270–281
25. Huang Z, Niu Q, Xiao S (2020) Human behavior recognition based on motion data analysis. *Int J Pattern Recognit Artif Intell* 34(9):2056005
26. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*
27. Kangas M et al (2009) Sensitivity and specificity of fall detection in people aged 40 years and over. *Gait Posture* 29(4):571–574
28. Kim Y, Bang H (2018) Introduction to Kalman filter and its applications. Open access peer-reviewed chapter. <https://doi.org/10.5772/intechopen.80600>
29. Kim J, Grauman K (2009) Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates. *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*. pp 2921–2928
30. Laxhammar R, Falkman G (2014) Online learning and sequential anomaly detection in trajectories. *IEEE Trans Pattern Anal Mach Intell* 36(6):1158–1173
31. Li Y, Zeng Z, Popescu M, Ho KC (2010) Acoustic fall detection using a circular microphone array. *Conf Proc IEEE Eng Med Biol Soc* pp 2242–2245
32. Lin T-Y, Dollar P et al (2017) Feature pyramid networks for object detection. *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*

33. Mabrouk AB, Zagrouba E (2018) Abnormal behavior recognition for intelligent video surveillance systems: a review. *Expert Syst Appl* 91:480–491
34. Mehran R, Oyama A, Shah M (2009) Abnormal crowd behavior detection using social force model. *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*. pp 935–942
35. Nannan L, Xinyu W et al (2015) Anomaly detection in video surveillance via gaussian process. *International Journal of Pattern Recognition and Artificial Intelligence*. 29(6):1555011
36. Nguyen VD, Le MT et al (2014) An efficient camera-based surveillance for fall detection of elderly people. In: *IEEE 9th conference on Industrial electronics and applications (ICIEA)*. pp 994–997
37. Nunez-Marcos A, Azkune G, Arganda-Carreras I (2017) Vision-based fall detection with convolutional neural networks. *Wirel Commun Mob Comput* 2017:1–16
38. Olah C (2015) Understanding LSTM networks. [Online]. Available: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>. Accessed 26 June 2019
39. Pan H et al (2018) Fighting detection based on pedestrian pose estimation. In: *11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. pp 1–5
40. Popescu M, Li Y, Skubic M, Rantz M (2008) An acoustic fall detector system that uses sound height information to reduce the false alarm rate. *Conf Proc IEEE Eng Med Biol Soc*. pp 4628–4631
41. Redmon J, Farhadi A (2018) Yolov3: an incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.
42. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: unified, real-time object detection. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA*. pp 779–788
43. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell* 39:1137–1149
44. Ryoo MS, Aggarwal JK (2009) Spatio-temporal relationship match: video structure comparison for recognition of complex human activities. In: *IEEE 12th International Conference on Computer Vision, Kyoto, Japan*
45. Stone EE, Skubic M (2015) Fall detection in homes of older adults using the Microsoft Kinect. *IEEE J Biomed Health Inform* 19:290–301
46. Tani MK, Lablack A, Ghomari A, Bilasco IM (2015) Events detection using a video-surveillance ontology and a rule-based approach. *Computer vision - ECCV 2014 workshops*. pp 299–308
47. Uijlings J, van de Sande K, Gevers T, Smeulders A (2013) Selective search for object. *Int J Comput Vis* 104(2):154–171
48. Van Beeck K, Van Engeland K, Vennekens J, Goedemé T (2017) Abnormal behavior detection in LWIR surveillance of railway platforms. *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. pp 1–6
49. Wang XF, He XH et al (2016) A classification method based on streak flow for abnormal crowd behaviors. *Optik* 127(4):2386–2392
50. Weiya R, Guohui L, Boliang S, Kuihua H (2015) Unsupervised kernel learning for abnormal events detection. *Vis Comput* 31:245–255
51. Welch G, Bishop G (2006) An introduction to the Kalman filter. http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf
52. Wojke N, Bewley A, Paulus D (2017) Deep SORT: simple online and realtime tracking with a deep association metric. *arXiv preprint arXiv:1703.07402*
53. Wu S, Moore BE, Shah M (2010) Chaotic invariants of Lagrangian particle trajectories for anomaly detection in crowded scenes. *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*. pp 2054–2060
54. Zhang J, Wu C, Wang Y (2020) Human fall detection based on body posture spatio-temporal evolution. *Sensors* 20(3):946
55. Zhao Z-Q, Zheng P, Xu S-t, Wu X (2019) Object detection with deep learning: a review. *IEEE Trans Neural Netw Lear Syst* 99:1–21
56. Zhao R, Wang Y, Jia P, Li C, Ma Y, Zhang Z (2021) Abnormal human behavior recognition based on image processing technology. In: *IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. pp 1924–1928
57. Zhu Y, Wang Z (2016) Real-time abnormal behavior detection in elevator. *Intelligent Visual Surveillance: 4th Chinese Conference*. pp 154–161