



# Video anomaly detection based on scene classification

Hongjun Li<sup>1,2</sup> · Xulin Shen<sup>1</sup> · Xiaohu Sun<sup>1</sup> · Yunlong Wang<sup>1</sup> · Chaobo Li<sup>1</sup> · Junjie Chen<sup>1,2</sup>

Received: 10 August 2022 / Revised: 26 November 2022 / Accepted: 6 April 2023 /

Published online: 29 April 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

## Abstract

As a significant research hotspot in the field of computer vision, video anomaly detection plays an essential role in ensuring public safety. Anomaly detection remains a challenging task given the complex situation in public areas and the large random distribution of crowds. The density of people in the same scene varies greatly due to the instability of the pedestrian volume. Specifically, the characteristics of crowd distribution mainly include low density, small aggregation and dispersion, or large aggregation and severe occlusion. Considering the large difference between high-density and low-density crowd characteristics, we propose an anomaly detection algorithm based on scene classification in order to obtain better anomaly detection result. Firstly, we propose a novel scene classification method, which uses pre-trained YoloV4 model to detect the number of people in the video frames and generate heatmaps, and extracts pixel features through the Double-Canny algorithm to represent the occlusion degree of the crowd. Furthermore, K-Means clustering is used to adaptively divide the scene into sparse and dense. Secondly, the Generative Adversarial Network (GAN) based on prediction and reconstruction is introduced to detect anomalies respectively, and the final accuracy is achieved by combining the detection accuracy of both networks. Finally, experiments on three benchmark datasets demonstrate the competitive performance of our method with the state-of-the-art methods.

**Keywords** Video anomaly detection · Scene classification · Generative adversarial network

## 1 Introduction

In recent years, a large number of researchers in the field of computer vision and artificial intelligence have focused on the automatic understanding and analysis of scenes in videos. The analysis of crowds in computer vision has grown in popularity. In order to reduce the loss of life and property caused by unexpected abnormal events, the corresponding alarm can be sent in time by automatically identifying the abnormal events in the video stream. The intelligent monitoring equipment will completely replace the traditional monitoring video as a standard facility. At the same time, video abnormal

---

✉ Hongjun Li  
lihongjun@ntu.edu.cn

Extended author information available on the last page of the article

behavior detection and its related applications have been intensively studied by researchers in industry and academia.

The current algorithm rarely combines crowd density to detect abnormal behavior due to the limitations of uneven crowd distribution and other factors in the scene. In settings with significant variances in crowd distribution, the majority of algorithms merely use the same framework to detect abnormal behaviors. Alafif et al. [2] proposed to extract dynamic features through optical flow and combine a transfer learning strategy to detect anomalous behavior in large-scale crowd scenes. Song et al. [27] proposed autoencoder networks based on adversarial attention, which integrate attention models into the decoder to detect specific individual anomalous behavior in video. These methods perform well in scenes with relatively uniform crowd distribution. However, when the video involves more complex scenes, such as the occurrence of assembled crowds, different sparsity of the same scene, etc., this not only greatly increases the complexity of system computing, but may lead to the detection failure of abnormal behavior [28]. As a result, a targeted design is required for the detection of abnormal behavior of crowd with uneven distribution in the actual scene. However, due to the unpredictable movement of pedestrians and the inability to effectively judge the density of crowd distribution in advance, it is difficult to find adaptive detection methods. In order to represent the crowd distribution, Xiong et al. [29] combined crowd counting results with crowd entropy, and highlighted the movement of the whole crowd by calculating optical flow, which solved the anomaly detection based on the global movement of the crowd. However, crowd density is variable, and the characteristics of different crowd density are quite diverse. There are some shortcomings in analyzing crowd density only from the global perspective. The detecting scenario can be generally classified into sparse scene and dense scene depending on the number of crowd and whether there is occlusion. Specifically, in dense scenes, there are a large number of people or people gather to form occlusion, while in sparse scenes, it is the opposite [17]. In the methods of recent years, the idea of reconstruction or future frame prediction is predominant for detecting anomalies. The reconstruction-based method is not good enough to distinguish between normal and abnormal samples. Even abnormal samples can be successfully reconstructed, which is particularly apparent in sparse scenes with a simple crowd distribution. The prediction models has to gain previous information and tries to predict what will happen next, which are sensitive to noises and any changes of former frames. In dense scene, the number of people is large and the distribution is complex. When the crowd changes, the abnormal scores will drop rapidly, making more normal frames judged as abnormal ones.

Combined with the above analysis, different methods are required for sparse and dense scenes. Therefore, this paper proposes an anomaly detection method based on scene classification to improve the performance of video anomaly detection in practical applications. We summarize our contributions as follows: i) Adaptive scene classification method. First, the number of people in the video frame is detected and the heatmap is generated by using the pre-trained YoloV4 model. Then, the foreground extraction and edge detection are carried out by using the background difference method and Double-Canny Edge operator respectively, and the pixel features such as foreground pixels and the ratio of edge pixels to foreground pixels are extracted. Finally, the scene is classified using K-Means clustering for the above mentioned extracted features. ii) Considering that different crowd densities have different distribution characteristics, we use a prediction-based method in sparse scenes, while a reconstruction-based method for dense scenes. The final detection accuracy is calculated by the weighted and summed of two methods.

The rest of the paper is organized as follows. In Section 2, we discuss the related work of crowd density estimation and video anomaly detection. We introduce the overall structure in Section 3. Then, the process of optimizations and verifications for the anomaly detection approach is implemented through a series of experiments in Section 4. We conclude the paper with a summary, limitations and future study in Section 5.

## 2 Related work

### 2.1 Crowd density classification

Crowd density classification refers to divide people with different densities into different grades in a crowded scene [14]. Early research techniques directing at the crowd density classification problem mainly rely on detection framework. In the methods for classifying crowd density classes, pixels are the more underlying features in video images. Pixel-based methods establish a mapping from foreground pixels to the number of people. Davies et al. implemented crowd density estimation [11] based on pixels, and applied background subtraction and edge detection to extract foreground pixels. The algorithm is simple to implement and has a low computation, and the accuracy is high in low-density crowd scenes. However, in high-density scenes, the linear relationship between the number of people and the number of pixels is affected by crowd occlusion, and the error of pixel statistical method is large. Compared with the pixel feature, the texture feature of the image is a higher-level feature description of the image. Most of the texture-based methods utilize Gabor features, LBP, GLCM, SIFT and Wavelet analysis for the estimation of crowd density [3, 8, 16], and achieved ideal results in high density. Jia et al. [13] combined pixel and texture features to adaptively switch algorithm with different crowd density scenes. The conventional texture analysis methods are incapable to retain the temporal information of moving crowd which is quite valuable information in the analysis of dynamic properties. A rotation invariant spatio-temporal local binary pattern [15] is proposed to extract dynamic texture of the moving crowd. Furthermore, CNN also plays an important role including learning linear or nonlinear functions from image patterns to corresponding crowd number. A deep residual architecture [22] based on multitask learning is proposed for the classification of extreme dense crowds. Huynh et al. [12] introduced an encoder-decoder architecture, which is composed of inception modules to learn the multi-scale feature representations for the density level classification. Besides, better performance is obtained by utilizing fully convolutional neural network [4] or dilated convolutional neural network [5]. In an extended depth of field image with a crowded scene, the distribution of people is highly unbalanced and often severely occluded each other. In order to accurately estimate the number of people, the Depth Information Guided Crowd Counting [30] uses the depth information of the image to divide the scene into a far-view area and a near-view area, then maps the far-view area to its crowd density map, and uses Yolo to detect the number of people in the near-view area. In order to solve the problems of occlusion, non-uniform density and angle of view change in the scene, Zhu et al. [31] divided the image into patches and sends image patches of different density levels into different regression networks to get the corresponding density maps by combing the patch scale discriminant regression network and CAM method. From the above analysis, the algorithm based on pixel feature statistics is simple to implement and has a low computation, while the algorithm based on texture features is easily affected by background noise under low crowd density, thus

affecting the classification performance. While CNN-based method relies on the validity of the data, which brings some difficulties to further study the inner principle.

## 2.2 Abnormal behavior detection

Video anomaly detection refers to the automatic identification of anomalous objects, behaviors or events in a specific environment. With the development of deep networks, reconstruction based and future frame prediction are the main methods to detect anomalies. Generative Adversarial Network (GAN) [10], as the representative of unsupervised network structure in recent years, has received extensive attention from the academic community due to its strong generative ability, and its unique adversarial learning idea also shows good development potential in the field of anomaly detection. In [25], the convolutional auto-encoder is combined with GAN, and the discriminator of GAN is used to improve the reconstruction ability of the auto-encoder. Additionally, the pixel intensity loss of the auto-encoder and the discriminator loss of the GAN are thoroughly taken into account in the abnormality detection process to identify abnormality. AdaNet [27] integrates the attention mechanism and ConvLSTM into a generator composed of an autoencoder network, which enables it to dynamically select the informative part of the encoded features for decoding, thereby improving the reconstruction accuracy of the generator. Ravanbakhsh et al. [24] designed a cross-channel network architecture including a dual-channel GAN for mutual generation of frames and optical flow. The majority of these models implement anomaly detection by feeding reconstruction images into the discriminator. Liu et al. [21] first used U-Net to predict future frames, then estimated the corresponding optical flow, and employed multiple loss constraints to enhance the appearance and motion information of predicted frames. Lee et al. [18] proposed an inter-frame predictor, which input the five frames before and after a given frame into the forward and backward ConvLSTM to predict the intermediate frame. At the same time of prediction, the inter-frame predictor adds an attention mechanism to obtain an attention map, which makes it more effectively focus on the prediction of motion information. After adversarial training, the generated and original sequences and attention maps are input to an appearance-motion joint detector to evaluate anomalies. In order to better extract spatial and temporal information, Lei et al. [19] cascade two AutoEncoders and stack the output of spatial network with input to feed the temporal information network. In addition, the anomaly detection system may have over fitting, low classification accuracy and high false positive rate when facing large and diverse data. Therefore, [26] combines Ant colony optimization algorithm and deep neural network to deal with these problems. To reduce the training time of the network model, [23] uses the combined local and global feature descriptors to select basic local and local image features. By combining the prediction of weak learners and using it as the knowledge input of multilayer perceptron meta learners, the generalization ability of the network is greatly improved.

In summary, The reconstruction-based method can provide video frames a multi-scale representation with improved spatial resolution, and the learning of the reconstruction network is frequently independent of any prior knowledge and class labels [7]. However, reconstruction method is not good enough to distinguish between normal and abnormal samples. Future frame prediction methods heavily rely on previous information, thus the detection results are sensitive to noises and any changes of former frames. The results may drop rapidly since prediction is not robust to the noise in actual surveillance videos.

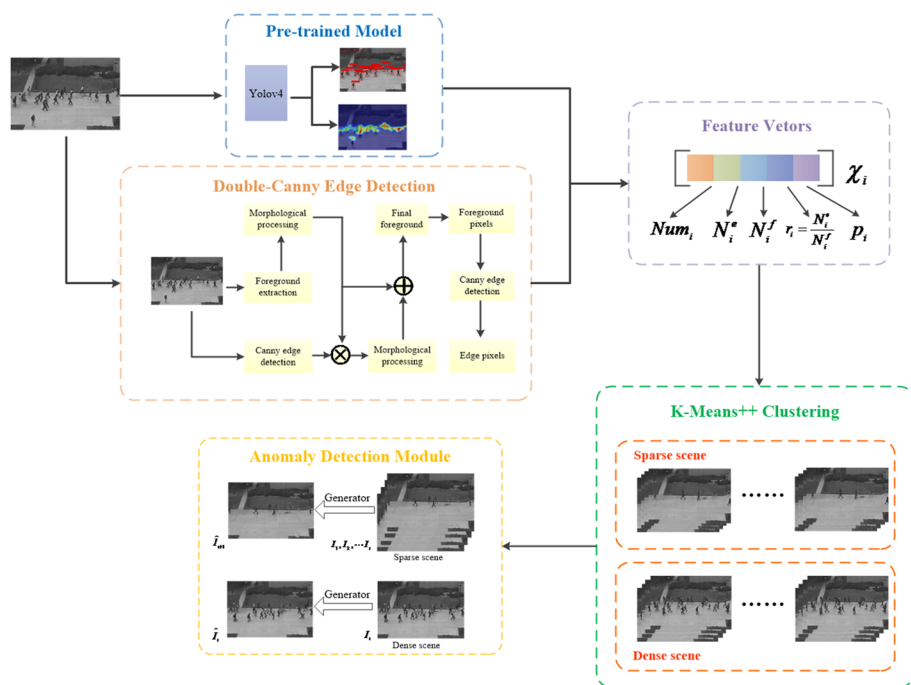
### 3 Method

The detection performance of video abnormal behavior is mainly affected by the complexity of the environment and the types of anomalies, among which the complexity of the environment is closely related to the density of moving objects in the scene. In recent years, the idea of reconstruction or future frame prediction is predominant for detecting anomalies. Reconstruction methods try to minimize the reconstruction error of training data. But due to the full learning capability of neural network, once the model is familiar with the background and the appearance characteristics of people during training, it can easily reconstruct the scene from the input frame itself. Therefore, in sparse scenes with simple crowd distribution, even abnormal samples can be reconstructed well. Future frame prediction methods predict what will happen next by gaining previous information. In the pedestrian street scene, the model learns the motion characteristics of pedestrians from the training data set. Once the stacked frames of runners are input into the model, it can only predict one person to keep on walking, which means there will be a large gap between the predicted frame and the real frame. However, the method is quite sensitive to noise and any changes in previous frames. When the crowd distribution changes in a dense scene with a large number of people and a complex crowd distribution, even if no abnormal events occur, the value used to evaluate the abnormality drops rapidly, resulting in more normal frames being judged as abnormal frames. Based on the above analysis, we adopt different methods to detect abnormal behavior in sparse and dense scenes, that is, the reconstruction based method is adopted in dense scenes, while the prediction based method is adopted in sparse scenes. Thus, we propose an anomaly detection method based on scene classification, which consists of two main modules, the scene classification module combining Double-Canny and pre-trained YoloV4 and the dual branch anomaly detection module, as shown in Fig. 1.

#### 3.1 Double-Canny Edge Detection (DECD) for pixel features

In sparse scenes, there is a strong linear relationship between crowd density and some statistical features of moving foreground (including foreground pixels, edge pixels, etc.), which can be used as one of the features to distinguish sparse scenes from dense scenes. Due to the influence of weather, lighting and other factors, the captured video needs to be pre-processed by graying and median filtering before extracting pixel features. For the extraction of foreground objects, the common optical flow method is complex in calculation, large in calculation and time-consuming, thus we choose background subtraction and frame difference method to extract foreground objects. At the same time, we use the erosion and opening operation in the morphological processing method to eliminate the noise in the obtained foreground image, so as to avoid errors in the next step of foreground pixel feature extraction. In addition to the foreground pixels, the edge pixels is also an important feature reflecting the number of people. Compared with other edge detection methods, canny edge detection operator has good denoising ability, and the extracted edges are more complete and accurate. Thus, we propose a Double-Canny Edge Detection(DECD) method. The method framework is shown in the orange box in Fig. 1.

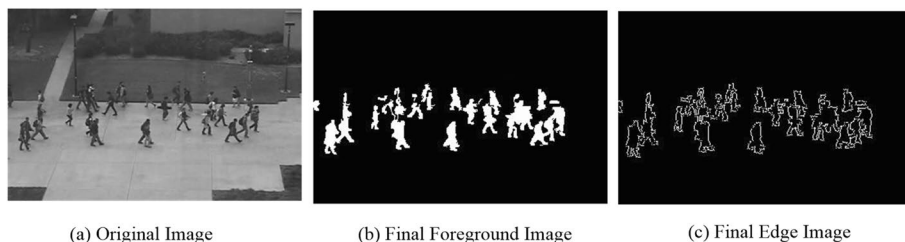
Given the original input frame, the canny operator is first executed to obtain the preliminary edge image, and the edge image and the preliminary foreground image are "sum" pixel by pixel to remove irrelevant edges and effectively capture the missing information in



**Fig. 1** The pipeline of our classification and detection structure

the foreground. After morphological processing of the result, the final foreground image is generated by pixel "or" operation with the preliminary foreground image. Finally, canny edge detection is performed on the final foreground image to obtain the final edge effect image. Figure 2 shows the final foreground image and edge image generated by the proposed DCED. The foreground image of the crowd is binarized with a foreground pixel value of 255 and a background pixel value of 0. The number of foreground pixels can be obtained by finding the number of pixels with a pixel value of 255, as follows:

$$N_f = \sum_{h=0}^H \sum_{w=0}^W [(f(h, w)) \oplus Mor(f(h, w) \otimes g(h, w))] \quad (1)$$



**Fig. 2** An example of the the proposed Double-Canny-Classification

$$N_e = \sum_{h=0}^H \sum_{w=0}^W [f(N_f(h, w))] \quad (2)$$

where  $N_f$  denotes the number of foreground pixels,  $W$  and  $H$  are the width and height of the image, respectively.  $f(\cdot)$  denotes the function of edge extraction while  $g(\cdot)$  denotes the combined function of the foreground extraction and  $Mor(\cdot)$ .  $f(h, w)$  and  $g(h, w)$  are the pixel value of foreground image generate by original image and overall edge image, respectively.  $\otimes$  and  $\oplus$  denote the “and” and “or” operation, respectively.  $Mor(\cdot)$  denotes the morphological processing.

If crowd density estimates are made directly using foreground pixels  $N_f$ , estimation errors can easily result from individual scale changes. The foreground and edge pixels in severe occlusion can not effectively represent the characteristic information of dense crowd. While the ratio of the edge pixels  $N_e$  to the foreground pixels  $N_f$  can reflect the degree of crowd occlusion. Therefore, we calculate the ratio of  $N_e$  to  $N_f$ , as follows:

$$r = \frac{N_e}{N_f} \quad (3)$$

In summary, we combine  $N_f$ ,  $N_e$  and  $r$  as pixel features to distinguish sparse and dense scenes, which can reflect the number of people in the current frame and the degree of occlusion. As an example, we select several videos in the training set of the three benchmark dataset, plotting the curves of  $N_e$  and  $r$  to each frame in the video, as shown in Fig. 3.

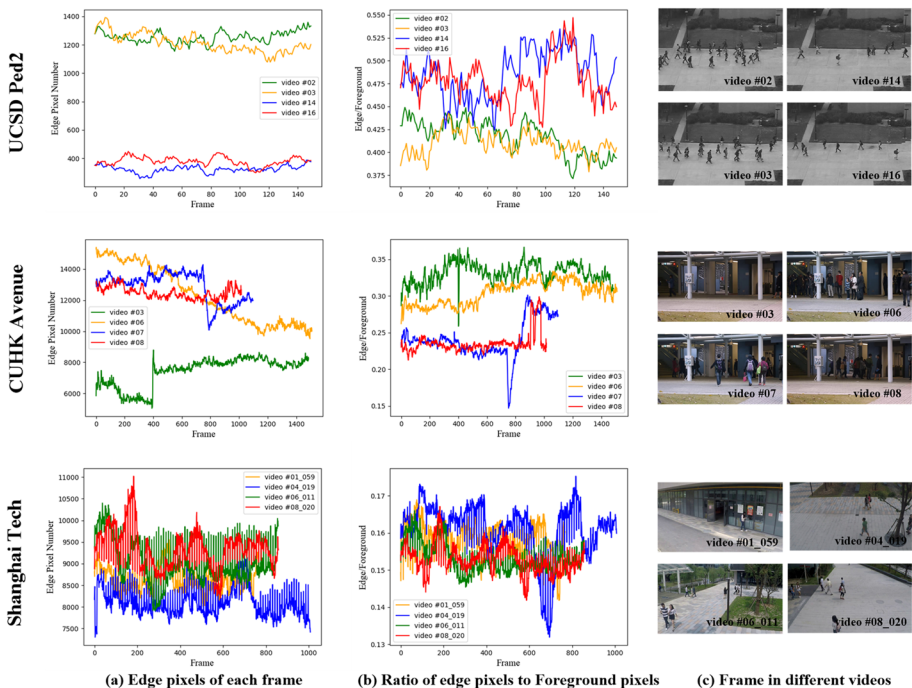


Fig. 3 Comparison of different video scenes on three datasets



When there is no occlusion in the crowd, the foreground pixels and edge pixels will increase with the number of people. However, when the number reaches a certain level, occlusion will occur in the crowd. At this situation, due to the serious crowd occlusion, the and cannot increase or even decrease as the crowd as it increases. The occlusion relationship between people can be seen in the ratio  $r$  of edge pixels to foreground pixels. In some extent, the ratio of to can reflect how people are occluded from each other. The ratio will decrease as the occlusion degree increases. Figure 3a is a comparison of the number of edge pixels in different clips of testing videos of three datasets. As can be seen, the trend of change in edge pixels in each video is basically the same as in Fig. 3c. Figure 3b is the comparison of the ratio. In terms of the UCSD Ped2 dataset, as can be seen that, the ratio of video #14 and video #16 are basically high while video #2 and video #3 are basically low. Thus video #2 and video #3 have some occlusion and the crowd is sparse in video #14 and video #16. The results are in line with the actual situation of four test videos in Fig. 3c. Combined with the above analysis, there are obvious differences between dense and sparse scenes in terms of the number of edge pixels, foreground pixels, and the ratio. As a result, pixel features can be extracted to provide a basis for subsequent scene classification. Notably, the Shanghai Tech dataset is more complex than the other two datasets. The crowd distribution is more volatile, so the corresponding curve are more fluctuating. In this complex situation, our approach effectively classifies subsequent scenes by distinguishing between the number of people in each video and the degree of occlusion.

### 3.2 YoloV4 for number of people and heatmap

Different from the mapping relationship between pixel features and crowd density, the number of people in the video frame can directly reflect the density of the crowd. Considering the importance of detection accuracy and speed, we use YoloV4 [6] to detect the number of people in video frames. This method has a good performance in terms of speed and accuracy. CSPDarknet53 is selected as the backbone network in this method, so as to better balance the input network resolution, convolution layer and parameters, and solve the problem of gradient information duplication in network optimization in the backbone network. As a result, the parameters and flops of model are reduced, guaranteeing reasoning speed and accuracy while also shrinking the overall size of model. In addition to using YoloV4 to detect the number of people in the video frame, in order to obtain the crowd distribution, we also generated a corresponding heatmap. By calculating the proportion  $p_i$  of the highlighted part in the heatmap, we can reflect the degree of crowd aggregation. In most cases, the larger the proportion of the high heat part, the more serious the crowd concentration.

Figure 4 shows the effectiveness of YoloV4 used in pedestrian detection and heatmap in sparse and dense scenes. The first column is the original video frame, and the other two columns are the pedestrian detection results and heatmap. As illustrated in Fig. 4, the YoloV4 algorithm we employ is capable of detecting pedestrians accurately in both sparse and dense scenes. In addition, YoloV4 can also generate precise heatmaps that represent the degree of crowd aggregation based on crowd distribution in various situations. In general, YoloV4 has excellent performance and provides a solid foundation for further scene classification.



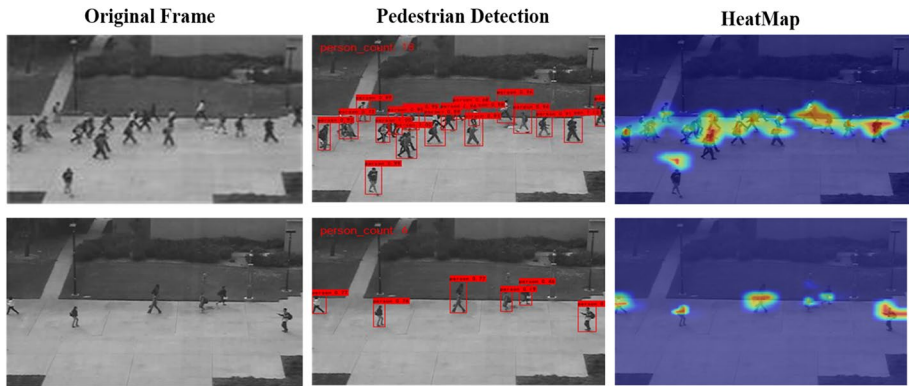


Fig. 4 Pedestrian detection results of YoloV4 on UCSD Ped2

### 3.3 K-Means clustering for scene classification

In order to adaptively distinguish sparse and dense scenes, we express the three pixel features extracted above, the number of people obtained by using YoloV4, and the proportion of the high heat part in the heatmap as a feature vector  $\chi_i = \left[ Num_i, N_i^e, N_i^f, r_i = \frac{N_i^e}{N_i^f}, p_i \right]$ , where  $Num_i$  is the number of the  $i$ th frame,  $N_i^e$  and  $N_i^f$  are the number of foreground pixels and edge pixels respectively,  $r_i$  is the ratio of  $N_i^e$  and  $N_i^f$ , and  $p_i$  represents the proportion of the highlighted part in the heatmap. Furthermore, introduce K-Means clustering method to automatically cluster the scenes in the video into two categories. In this algorithm, we set K to 2. The clustering results of the three datasets are shown in Fig. 5, and Table 1 show the proportion of sparse and dense scenes on training sets of three datasets.

### 3.4 Anomaly detection module

In the anomaly detection phase, proposes a dual-branch network structure based on GAN. Specifically, in sparse scenes, the prediction model is selected, and U-Net is used as a generator to predict the the next frame, while the reconstruction model is applied in dense

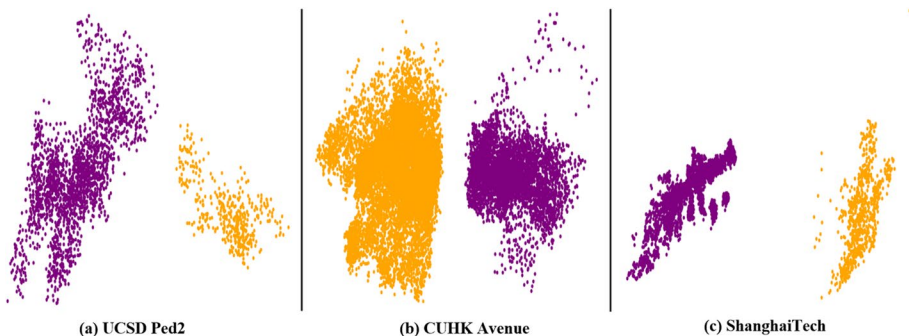


Fig. 5 Comparison of different video scenes

**Table 1** Proportion of sparse and dense scenes on training sets of three datasets

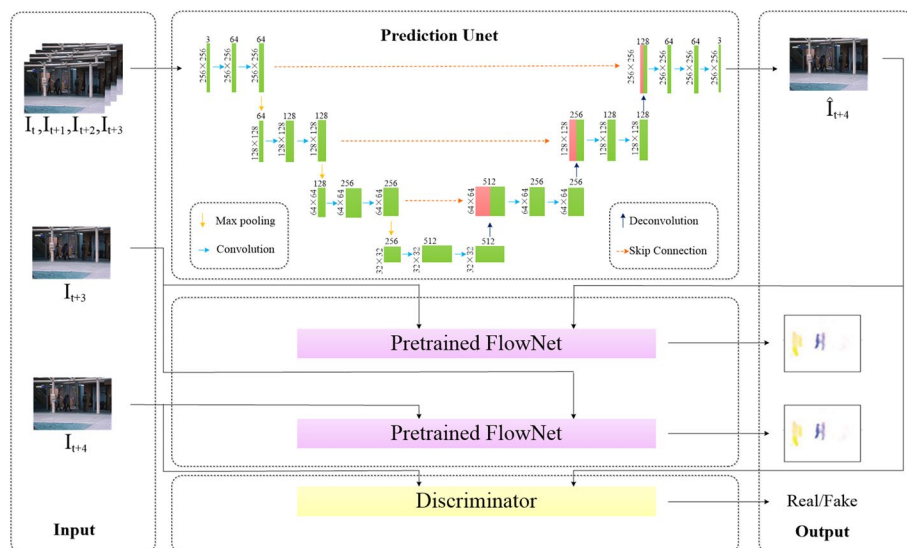
Dataset	Sparse	Dense
UCSD Ped2	0.58	0.42
CUHK Avenue	0.65	0.35
Shanghai Tech	0.61	0.39

scenes, and its generator is a U-Net without skip connection, which is used to reconstruct video frames.

### 3.4.1 Prediction-based abnormal behavior detection for sparse scenes

In sparse scenes, the distribution of the crowd is more scattered and the number of people is small. In order to generate higher quality future frames during the prediction process, in addition to adding intensity and gradient constraints, optical flow constraints are introduced to make the optical flow between the predicted frames and the ground truth frames consistent. For the convolution and deconvolution layers, the kernel size is  $3 \times 3$ , and for the maximum pooling layer, the kernel size is  $2 \times 2$ , the structure of our prediction network as shown in Fig. 6.

**Loss function.** To minimize the difference between the predicted frame and the original frame, intensity loss, gradient loss and optical flow loss are used as constraints. The intensity constraint compares the value of each pixel between the two frames, while the gradient constraint compares the gradient of the pixel values at the same position in the two frames and sharpens the generated frame. Both constraints are based on appearance, whereas for anomalous behavior detection, the accuracy of motion prediction needs to be guaranteed,

**Fig. 6** The structure of our prediction network for sparse scenes

hence optical flow estimation via FlowNet and the introduction of optical flow loss. Considering the instability of GAN training, we use Least Squares GAN (LSGAN) to predict the next frame. The intensity loss is the L2 distance between the predicted frame  $\hat{I}$  and the real frame  $I$ .

$$L_{\text{int}}(\hat{I}, I) = \|\hat{I} - I\|_2^2 \quad (4)$$

$$L_{\text{gd}}(\hat{I}, I) = \sum_{i,j} \left| \|\hat{I}_{i,j} - \hat{I}_{i-1,j}\| - \|I_{i,j} - I_{i-1,j}\| \right|_1 + \left| \|\hat{I}_{i,j} - \hat{I}_{i,j-1}\| - \|I_{i,j} - I_{i,j-1}\| \right|_1 \quad (5)$$

Optical flow loss is defined as the distance between the optical flow of the predicted frame and the optical flow of the actual frame, and flowNet is denoted by  $\phi$ .

$$L_{\text{op}}(\hat{I}_{t+1}, I_{t+1}, I_t) = \|\phi(\hat{I}_{t+1}, I_t) - \phi(I_{t+1}, I_t)\|_1 \quad (6)$$

**Adversarial training.** In the GAN architecture, the discriminator tries to distinguish between real frames and generated frames, thus facilitating the self-renewal of the generator  $G_p$ . The continuously optimized generator tries to fool the discriminator with images that are closer to the real ones, thus facilitating the discriminator to improve its discriminative ability. Assuming that zero is set as the label of the generated image and 1 represents the real image, the discriminator aims to correctly label the corresponding image, defining the adversarial loss of  $D_p$  as shown in Eq. (10), and the generator aims to generate images that can be labelled as 1. When training  $G_p$ , the weight of  $D_p$  is fixed, defining the adversarial loss of  $G_p$ , as shown in Eq. (8).

$$L_{\text{adv}}^{D_p}(I, \hat{I}) = \frac{1}{2}(D_p(I) - 1)^2 + \frac{1}{2}(D_p(\hat{I}) - 0)^2 \quad (7)$$

$$L_{\text{adv}}^{G_p}(\hat{I}) = \frac{1}{2}(D_p(\hat{I}) - 1)^2 \quad (8)$$

**Objective function.** Combining the constraints of appearance, movement and adversarial training to acquire the objective function, as in Eq. (9).

$$L_{G_p} = \lambda_{\text{int}} L_{\text{int}}(\hat{I}_{t+1}, I_{t+1}) + \lambda_{\text{gd}} L_{\text{gd}}(\hat{I}_{t+1}, I_{t+1}) + \lambda_{\text{op}} L_{\text{op}}(\hat{I}_{t+1}, I_{t+1}, I_t) + \lambda_{\text{adv}} L_{\text{adv}}^{G_p}(\hat{I}_{t+1}) \quad (9)$$

When training  $D_p$ , the loss function is as in Eq. (10).

$$L_{D_p} = L_{\text{adv}}^{D_p}(\hat{I}_{t+1}, I_{t+1}) \quad (10)$$

In training network,  $\lambda_{\text{int}}$ ,  $\lambda_{\text{gd}}$ ,  $\lambda_{\text{op}}$ ,  $\lambda_{\text{adv}}$  are set to 1.0, 2.0, 2.0, 0.05 respectively.

### 3.4.2 Reconstruction-based abnormal behavior detection for dense scenes

In dense scenes, targets that reappear after occlusion are easy to miss. The reconstruction-based approach only reconstructs based on the current frame, so the probability of missing an occluded target is less. For the reconstruction network, except that there is no skip connection,

other settings are basically consistent with the prediction network, the structure of the generator as shown in Fig. 7.

The loss function of the reconstruction network is defined as Eq. (11). At the same time, the loss function for the adversarial training is shown Eq. (12).

$$L_{adv}^{G_r}(\hat{I}) = \frac{1}{2}(D_r(\hat{I}) - 1)^2 \quad (11)$$

$$L_{adv}^{D_r} = \frac{1}{2}[(D_r(I) - 1)^2] + \frac{1}{2}[(D_r(\hat{I}) - 0)^2] \quad (12)$$

This is then combined with the reconstruction loss  $L_{adv}^{G_r}$ , the constraints of appearance  $L_{int}$ ,  $L_{gd}$  and movement  $L_{op}$  like the prediction model to obtain the objective function for the whole model:

$$L_{G_r} = \lambda_{int}L_{int}(\hat{I}_{t+1}, I_{t+1}) + \lambda_{gd}L_{gd}(\hat{I}_{t+1}, I_{t+1}) + \lambda_{op}L_{op}(\hat{I}_{t+1}, I_{t+1}, I_t) + \lambda_{adv}L_{adv}^{G_r}(\hat{I}_{t+1}) \quad (13)$$

In the training phase, the coefficients  $\lambda_{int}$ ,  $\lambda_{gd}$ ,  $\lambda_{op}$ ,  $\lambda_{adv}$  are set to 1.0, 2.0, 2.0, 0.05 respectively.

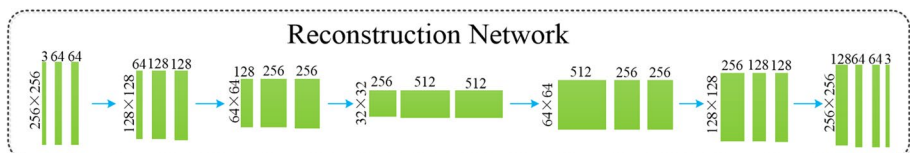
### 3.4.3 Anomaly score

Assuming that normal events can be well reconstructed or predicted, anomaly detection can be performed using the difference between generated and real frames. Peak Signal to Noise Ratio (PSNR) is used to represent the ratio between the maximum possible power of a signal and the power of destructive noise that affects the fidelity of its representation. It is mainly used to quantify the reconstruction quality of images and videos affected by lossy compression.

$$PSNR(I, \hat{I}) = 10 \lg \frac{[\max_i]^2}{\frac{1}{N} \sum_{i=0}^N (I_i - \hat{I}_i)^2} \quad (14)$$

The smaller the PSNR, the worse the image quality and the greater the likelihood that the frame is an abnormal frame. Once the PSNR of all test video frames have been calculated, all PSNR values are normalized to the [0,1] range and the score for each frame is calculated using Eq. (15).

$$S(t) = \frac{PSNR(I_t, \hat{I}_t) - \min_t PSNR(I_t, \hat{I}_t)}{\max_t PSNR(I_t, \hat{I}_t) - \min_t PSNR(I_t, \hat{I}_t)} \quad (15)$$



**Fig. 7** The structure of our reconstruction network for dense scenes

## 4 Experimental results

### 4.1 Datasets and evaluation metric

To validate the proposed anomaly detection model for practical applications of video surveillance, this section evaluates the performance of the proposed model on the three popular benchmark datasets UCSD Ped2, CHUK Avenue and ShanghaiTech.

#### 4.1.1 UCSD Ped2

It is captured from outdoor cameras having 10fps with an image size of  $240 \times 360$ . The crowd density in the walkways was variable, ranging from sparse to very crowded. It contains 16 training videos and 12 testing videos with a total of 4560 images that containing 12 anomalies. In the normal setting, the video contains only pedestrians. Commonly occurring anomalies include bikers, skaters, small carts, and people walking across a walkway, or in the grass that surrounds it.

#### 4.1.2 CHUK Avenue

It contains 16 training videos and 21 test videos with a resolution of  $360 \times 640$ . And as the camera position and the angle change, the size of the captured portrait also changes. There are 14 unusual events including running, throwing objects, and loitering. The training videos contain only normal events, and the testing videos contain both normal and abnormal events.

#### 4.1.3 Shanghai Tech

The training set contains 330 training videos, more than 270,000 training frames, and the testing set contains 107 test videos, 130 abnormal events, with a resolution of  $480 \times 856$ . It contains 13 scenes with multiple perspectives. In terms of data volume and scene diversity, it can be said to be unique. In addition, the dataset introduces the anomalies caused by sudden motion, such as chasing and brawling, which are not included in existing datasets. These characteristics make the dataset more suitable for real scenes.

In the field of abnormal behavior detection, a common evaluation metric is to calculate the receiver characteristic operating curve (ROC) by gradually varying the threshold of the conventional score, and then accumulate the area under the curve (AUC) as a scalar for model performance evaluation, with higher values indicating better detection performance.

### 4.2 Setup

Before training, the size of each frame is resized to  $256 \times 256$ . The learning rates of the prediction model and the reconstruction model generator and the discriminator are the same,  $1e-4$  and  $1e-5$ , respectively. For parameter optimization, we choose the Adam optimizer to train the whole model end to end, where the decay rate is  $1e-4$ , the learning rate is  $1e-5$ , the momentum is 0.9, the batch size is 16.

All experiments are carried out on a dedicated GPU server with Intel Xeon Silver 4216 CPU running at 2.1 GHz, 32 GB of RAM, a NVIDIA TITANX GPU and 24G video memory. We use the Pytorch framework to implement our anomaly detection architecture with Windows 10, Python 3.7, Cuda 10.0.

### 4.3 Visualization of DCED

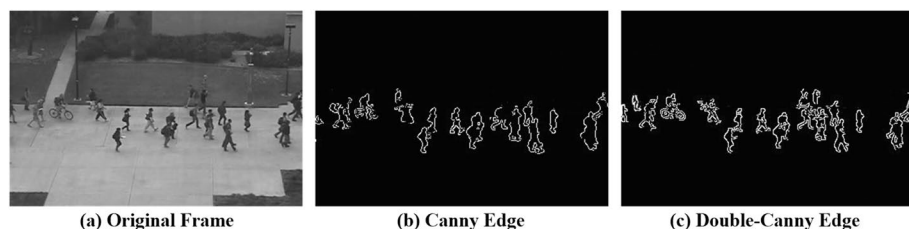
To demonstrate the effectiveness of Double-Canny we used in the classification stage, we visualized some examples on the UCSD Ped2 dataset to fully demonstrate the effectiveness of the methods we used. Figure 8 visualizes an example of the Double-Canny used, where (a) represents the original video frame, and (b) and (c) represent the edge detection results generated when using one and Double-Canny operations, respectively.

As can be seen from Fig. 8, the result of using single canny operation is inferior to that of using Double-Canny operations. This is because the background subtraction method and the frame difference method are unable to capture the slow moving or stationary pedestrians when the moving speed is small or there is noise, thus there are certain defects in the edge map obtained. The Double-Canny algorithm can effectively capture the pedestrian edges that move slowly or not significantly, and generate a clearer edge effect map. It is important to note that the additional canny operation runs concurrently with the original operation, preserving the original edge detection time and requiring low calculation.

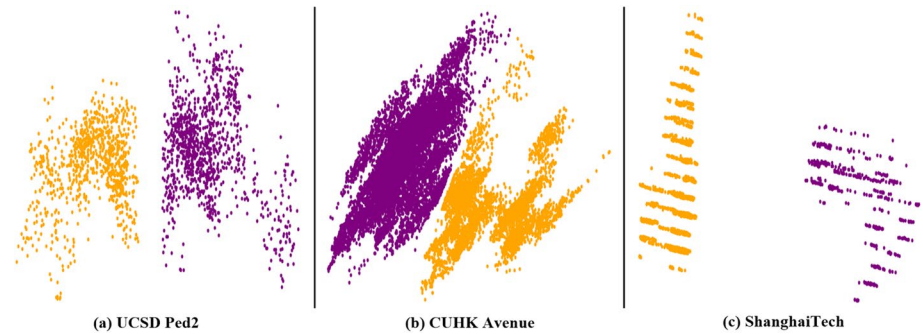
### 4.4 Classification results

In the testing stage, we first divide scenes of each dataset into sparse and dense using the proposed classification framework. Specifically, we obtain the number of people and heat-map in the scene through YoloV4, and the foreground pixels, edge pixels and their ratios are obtained by Double-Canny. The feature vectors generated by these five features are clustered, and the final sparse and dense partitions are obtained. As shown in the Fig. 9, we use PCA to map the classified feature vectors into the two-dimensional space. Furthermore, we also give the proportion of sparse and dense scenes in each dataset in Table 2.

First, it can be seen from Fig. 9 that the proposed K-Means clustering method performs well on each dataset. The classification results on UCSD Ped2 are relatively scattered, because the UCSD Ped2 dataset itself carries obvious noise, which will affect the classification results to some extent. Additionally, compared to CUHK Avenue and Shanghai Tech, its clustering results are less compact due to the fact that its sparse and dense scenes include a lot of different scenes. In the Shanghai tech dataset, we visualized the classification result of the third scene. Similar to the classification results in the training set,



**Fig. 8** Edge detection results of the proposed Double-Canny on UCSD Ped2



**Fig. 9** Classification results on three datasets

**Table 2** Proportion of Sparse and Dense Scenes on testing sets of three datasets

Dataset	Sparse	Dense
UCSD Ped2	0.74	0.26
CUHK Avenue	0.55	0.45
Shanghai Tech	0.81	0.19

**Table 3** AUC of different methods on the three datasets

Method	UCSD Ped2	CHUK Avenue	ShanghaiTech
ALOCC [25]	94.3	84.7	-
Ada-Net [27]	90.7	<b>89.2</b>	70.0
Adversarial Discriminator [24]	<u>95.5</u>	-	-
Liu et al. [21]	95.4	85.1	<u>72.8</u>
MemAE [9]	94.1	83.3	71.2
LSA [1]	95.4	-	72.5
ST-CaAE [20]	92.9	83.5	-
Ours	<b>95.9</b>	<u>85.9</u>	<b>73.4</b>

the proposed K-Means method can successfully classify them into sparse and dense. Compared with the single scene of UCSD Ped2 dataset, CUHK Avenue dataset contains more scenes, hence its classification results are slightly confused. In addition, it can be seen from Table 2 that the proportion of the divided sparse scenes is larger than that of the dense scenes, which is consistent with the actual results in the dataset. Table 2 shows that the proportion of divided sparse scenes is higher than dense scenes, which is consistent with the actual results in the dataset.

#### 4.5 Comparison with existing methods

To prove the effectiveness of our method, We compare it with the state of the art [1, 9, 20, 21, 24, 25, 27], on the publicly available datasets.



From the AUC results of the proposed method and other methods in Table 3, we observe three things: (1) The proposed method shows the first two performance on all three datasets, achieving the average AUC of 95.9%, 85.9%, and 73.4%, respectively. Especially on UCSD Ped2, our method appears at least 0.4% (95.9% vs 95.5%) higher than other methods. This demonstrates the effectiveness of the proposed method to classify crowd scenes for anomaly detection. (2) Compared with the Liu et al. [21], the proposed method has 0.5%, 0.8% and 0.6% improvement on three datasets, respectively. Compared with MemAE [9], which had superior performance by using the combination of GAN and autoencoders to learn feature representations, the proposed method achieves about 1.6% and 1.2% AUC increases for UCSD Ped2 and Avenue datasets. This can be attributed that, for one thing, LSGAN is superior to GAN, for the other thing, the scene classification does help the model enhance the ability of detection. (3) Ada-Net [27] combines an autoencoder network and a GAN model that is used to benefit enhancing the reconstruction ability of the autoencoder, the performance of which is superior to the proposed method on CHUK Avenue (89.2.% vs 85.9%). This is because it integrates an attention model into the decoder to dynamically select informative parts of encoding features for decoding and the attention mechanism is helpful to preserving important information for learning intrinsic normal patterns. Although the attention module it used is very useful, the simple basic model limit its performance and our method outperforms it on the other two datasets (95.9% vs 90.7% & 73.4% vs 70.0%).

To qualitatively analyze the anomaly detection performance of our model, we plot the ROC curves of the baseline algorithms for comparison, as shown in Fig. 10. By varying the threshold parameter, we can obtain a series anomaly detection results and their corresponding false positive rates (FPR) and true positive rates (TPR). Thus, the ROC curve can be plotted by the series of coordinate points composed of FPRs and TPRs (The ROC curves of the comparison methods are taken from the original papers). It can be seen that our method outperforms both MemAE [9] and Liu et al. [21] on all three datasets. All in all, the proposed scene classification does make the model to be improved on the original basis.

## 4.6 Ablation study

To evaluate how the proposed scene classification method affects the anomaly detection performance, several ablation experiments are conducted on three datasets, as shown in Table 4. To begin with, we test the detection performance of the proposed Reconstruction

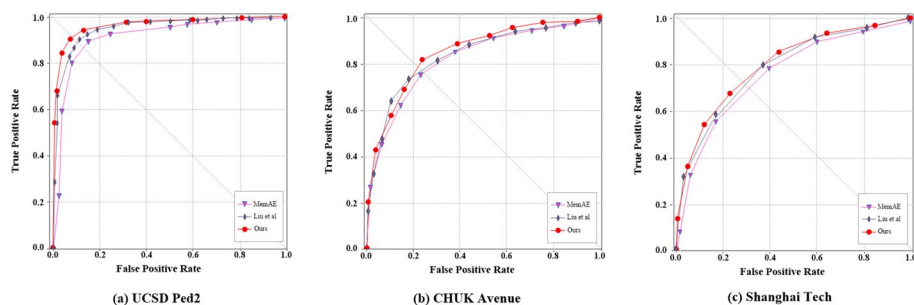


Fig. 10 ROC comparison of different methods on three datasets

**Table 4** Ablation study on three datasets

Detection	Prediction	-	√	√
	Reconstruction	√	-	√
Classification		-	-	√
AUC (%)	Ped2	94.5	95.2	95.9
	CHUK Avenue	84.2	85.1	85.9
	Shanghai Tech	71.3	72.7	73.4

and Prediction method. Further, a comprehensive study with two detection methods and the scene classification to be used is conducted.

Taking the UCSD Ped2 dataset as an example, it can be seen from Table 4 that the AUC results of the proposed reconstruction and prediction methods presents a significant performance of 94.5% and 95.2% respectively on UCSD Ped2. With the addition of the scene classification, the final detection performance can achieve 1.4% (95.9% vs 94.5%) and 0.7% (95.9% vs 95.2%) improvement when compared to the two detection methods. For CUHK Avenue and Shanghai Tech datasets, the proposed method can substantially enhance the model. In a word, the remarkable performance on the three datasets reveals that the proposed scene classification can make a good distinction between sparse and dense scenarios, and uses more appropriate methods to detect anomalies for different scenes, effectively improving overall exception detection performance.

#### 4.7 Effectiveness of YoloV4 and DCED

To further explore the impact of each component of our proposed classification framework on anomaly detection performance, we carried out a series of experiments on the CUHK Avenue dataset. Specifically, we first conducted experiments on the impact of YoloV4 and DCED modules on the performance of model anomaly detection, then introduced K-Means clustering strategy and explored its impact on the performance of the model, as shown in Table 5.

First of all, we can see that when YoloV4 is used alone for crowd classification without using K-Means, the final anomaly detection performance of the network can reach 85.2%; If the DCED module is used alone, it can reach 85.3%. Compared with a single reconstruction model or prediction model, these two methods can bring considerable improvement to the model, especially when the two methods are combined for classification tasks, the performance of the model can further reach 85.5%. When using K-Means, we can observe that: 1) The combination of K-Means and any module can further improve the performance of the model. 2) When K-Means, YoloV4 and DCED are used simultaneously, the performance of the model will reach the best state—85.9%.

**Table 5** Ablation study on CUHK Avenue

K-Means	-	-	-	√	√	√
YoloV4	√	-	√	√	-	√
DCED	-	√	√	-	√	√
AUC on CUHK Avenue	85.2	85.3	85.5	85.5	85.7	85.9

**Table 6** AUC of different methods on UCSD Ped2

Sparse	Prediction	✓	-	-	-
	Reconstruction	-	✓	-	-
Dense	Prediction	-	-	✓	-
	Reconstruction	-	-	-	✓
AUC (%)	Ped2	96.4	94.0	94.3	95.4
	CHUK Avenue	86.4	83.4	83.8	85.1
	Shanghai Tech	73.6	71.6	70.9	72.8

**Table 7** Parameter, GFLOPs, and FPS for the proposed model

Architecture	Para.(M)	GFLOP/s	FPS
Classification	53.2	131.6	76.1
Detection	48.2	40.5	27.4
Classification + Detection	101.4	30.3	20.3

#### 4.8 Comparison of sparse and dense scenes

In this section, in order to further explore the applicability of different detection methods in different scenarios, we have conducted a series of exploration experiments. Specifically, for the proposed reconstruction and prediction methods, we divided the training set and the test set into sparse and dense scenarios in advance. According to the scene classification, there are 1487 sparse scenes in UCSD Ped2 dataset, with a proportion of 0.74, and 523 dense scenes, with a proportion of 0.26. The CUHK Avenue dataset contains 8428 sparse scenes, which account for 0.55; 6896 dense scenes, which account for 0.45. In the Shanghai Tech dataset, there are 34,735 sparse scenes that account for 0.81 and 8148 dense scenes that account for 0.19. Taking the proportion as the weights of the two networks, the final accuracy of the model is obtained after weighting and summing. We train both methods in sparse/dense training scenarios and test the performance of anomaly detection under sparse/dense testing scenarios, respectively.

As we can see from Table 6 that, the prediction method is superior to the reconstruction method for sparse scenes, on the three datasets, the performance difference is 2.4% (96.4% vs 94.0%), 3.0% (86.4% vs. 83.4%) and 2.0% (73.6% vs. 71.6%) respectively. Similarly, in dense scenarios, the performance of the reconstruction method is 1.1% (95.4% vs. 94.3%), 1.3% (85.1% vs. 83.8%) and 1.9% (72.8% vs. 70.9%) higher than that of the prediction method respectively. This can be attributed that the prediction method does not predict these occlusion scenarios well and is better suited to sparse scenarios. On the contrary, reconstructions can perfectly reconstruct these occlusion scenarios. Therefore, we use the prediction method in sparse scenarios and the reconstruction method in dense scenarios to make the model's detection performance even higher.

#### 4.9 Complexity of the proposed method

To explore the complexity of the proposed model, we report the parameters, GFLOPs, and FPS of the different components of the model in Table 7. FLOPs are calculated using the input size of  $256 \times 256$ . As shown in Table 7, both classification and exception detection

modules are lightweight in size and have lower computational costs. In the case of FPS, a total model that combines classification and exception detection modules can reach 20.3 FPS, i.e., requiring 0.0492 s for process of a frame.

## 5 Conclusion

In this paper, we have proposed a dual branch network based on scene classification for video anomaly detection, so as to solve the problem of uneven crowd distribution in the scene. We have proved that the proposed Double-Canny algorithm combined with YoloV4 detection can effectively divide different scenes into sparse scenes and dense scenes. We have also used different networks to detect anomalies in sparse and dense scenes, effectively reaping the benefits of different networks for anomaly detection. It opens up a new way for anomaly detection training schemes for GAN-like deep networks. Extensive evaluations on standard benchmarks show that our method outperforms existing methods by a large margin, which proves the effectiveness of our method in anomaly detection. Due to the variable angles captured by surveillance cameras in real scenes, it is harder to find a universal algorithm to classify scenes in a fine-grained manner. As a result, a new method of anomaly detection coupled with a more detailed edge detection classification network will be central to our future work.

**Funding** This work is supported in part by National Natural Science Foundation of China under Grant 61871241, Grant 61971245 and Grant 61976120, in part by Jiangsu Industry University Research Cooperation Project BY2021349, in part by Nantong Science and Technology Program JC2021131 and in part by Postgraduate Research and Practice Innovation Program of Jiangsu Province KYCX21\_3084 and KYCX22\_3340.

**Data availability** We provide original and editable data appearing in the submitted article, including figures, tables and experimental results.

**Code availability** We are pleased to share code that is used in work submitted for publication. Authors' contributions: All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Hongjun Li, Xulin Shen, Xiaohu Sun, Yunlong Wang, Chaobo Li, Junjie Chen. The first draft of the manuscript was written by Xulin Shen and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

## Declarations

**Conflicts of interest/Competing interests** None.

## References

1. Abati D, Porrello A, Calderara S et al (2019) Latent space autoregression for novelty detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 481–490. IEEE Computer Society, Long Beach, USA
2. Alafif Tarik et al (2022) Generative adversarial network based abnormal behavior detection in massive crowd videos: a hajj case study. *J Ambient Intell Humaniz Comput* 13(8):4077–4088
3. Alanazi AA, Bilal M (2019) Crowd density estimation using novel feature descriptor. *arXiv preprint arXiv:1905.05891*
4. Bhuiyan MR, Abdullah J, Hashim N et al (2022) A deep crowd density classification model for Hajj pilgrimage using fully convolutional neural network. *PeerJ Comput Sci* 8:e895


5. Bhuiyan R, Abdullah J, Hashim N et al (2022) Deep dilated convolutional neural network for crowd density image classification with dataset augmentation for hajj pilgrimage. *Sensors* 22(14):5102
6. Bochkovskiy A, Wang CY, Liao HYM (2020) Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*
7. Chen D, Yue L, Chang X et al (2021) NM-GAN: Noise-modulated generative adversarial network for video anomaly detection. *Pattern Recognit* 116:107969
8. Gong S, Bourenane EB (2019) A method based on texture feature and edge detection for people counting in a crowded area. In: *Digital Image and Signal Processing*
9. Gong D, Liu L, Le V et al (2019) Memorizing normality to detect anomaly: memory-augmented deep autoencoder for unsupervised anomaly detection. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp 1705–1714. Institute of Electrical and Electronics Engineers Inc, Seoul, Korea
10. Goodfellow IJ, Pouget-Abadie J, Mirza M et al (2014) Generative adversarial networks. *Adv Neural Inf Process Syst* 3:2672–2680
11. Hussain N, Yatim HSM, Hussain NL et al (2011) CDES: A pixel-based crowd density estimation system for Masjid al-Haram. *Saf Sci* 49(6):824–833
12. Huynh VS, Tran VH, Huang CC (2019) Iuml: Inception u-net based multi-task learning for density level classification and crowd density estimation. In: *IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp 3019–3024. IEEE
13. Jia D, Zhang C, Zhang B (2021) Crowd density classification method based on pixels and texture features. *Mach Vis Appl* 32(2):1–22
14. Jiang X, Zhang L, Xu M et al (2020) Attention scaling for crowd counting. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp 4705–4714. IEEE Computer Society, Seattle, USA
15. Lamba S, Nain N (2017) A large scale crowd density classification using spatio-temporal local binary pattern. In: *13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pp 296–302. IEEE
16. Lamba S, Nain N (2019) A texture based mani-fold approach for crowd density estimation using Gaussian Markov Random Field. *Multimedia Tools Apply* 78(5):5645–5664
17. Lazaridis L, Dimou A, Daras P (2018) Abnormal behavior detection in crowded scenes using density heatmaps and optical flow. In: *26th European Signal Processing Conference (EUSIPCO)*, pp 2060–2064. European Signal Processing Conference, Rome
18. Lee S, Kim HG, Ro YM (2020) BMAN: bidirectional multi-scale aggregation networks for abnormal event detection. *IEEE Trans Image Process* 29:2395–2408
19. Lei Z, Deng F, Yang X (2019) Spatial temporal balanced generative adversarial autoencoder for anomaly detection. In: *Proceedings of the 2019 International Conference on Image, Video and Signal Processing*, pp 1–7
20. Li N, Chang F, Liu C (2020) Spatial-temporal cascade autoencoder for video anomaly detection in crowded scenes. *IEEE Trans Multimed* 23:203–215
21. Liu W, Luo W, Lian D et al (2018) Future frame prediction for anomaly detection - a new baseline. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 6536–6545. IEEE Computer Society, Salt Lake City
22. Marsden M, McGuinness K, Little S et al (2017) Resnetcrowd: A residual deep learning architecture for crowd counting, violent behaviour detection and crowd density level classification. In: *14th IEEE international conference on advanced video and signal based surveillance (AVSS)*, pp 1–7. IEEE
23. Naeem H, Cheng X, Ullah F et al (2022) A deep convolutional neural network stacked ensemble for malware threat classification in internet of things. *J Circuits Syst Comput* 31:2250302
24. Ravanbakhsh M, Sangineto E, Nabi M et al (2019) Training adversarial discriminators for cross-channel abnormal event detection in crowds. In: *19th IEEE Winter Conference on Applications of Computer Vision*, pp 1896–1904. Institute of Electrical and Electronics Engineers Inc, Hilton Waikoloa Village
25. Sabokrou M, Khalooei M, Fathy M et al (2018) Adversarially learned one-class classifier for novelty detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 3379–3388. IEEE Computer Society, Salt Lake City, USA
26. Samriya JK, Tiwari R, Cheng X et al (2022) Network intrusion detection using ACO-DNN model with DVFS based energy optimization in cloud framework. *Sustain Comput Inform Syst* 35:100746
27. Song H, Sun C, Wu X et al (2019) Learning normal patterns via adversarial attention-based autoencoder for abnormal event detection in videos. *IEEE Trans Multimedia* 22(8):2138–2148
28. Wang P, Wang P, Fan E (2021) Violence detection and face recognition based on deep learning. *Pattern Recogn Lett* 142:20–24

29. Xiong G, Cheng J, Wu X et al (2012) An energy model approach to people counting for abnormal crowd behavior detection. *Neurocomputing* 83:121–135
30. Xu M, Ge Z, Jiang X et al (2019) Depth information guided crowd counting for complex crowd scenes. *Pattern Recogn Lett* 125:563–569
31. Zhu L, Li C, Yang Z et al (2020) Crowd density estimation based on classification activation map and patch density level. *Neural Comput Appl* 32(9):5105–5116

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

## Authors and Affiliations

Hongjun Li<sup>1,2</sup>  · Xulin Shen<sup>1</sup>  · Xiaohu Sun<sup>1</sup>  · Yunlong Wang<sup>1</sup>  · Chaobo Li<sup>1</sup>  · Junjie Chen<sup>1,2</sup> 

Xulin Shen  
2010310003@stmail.ntu.edu.cn

Xiaohu Sun  
2010310052@stmail.ntu.edu.cn

Yunlong Wang  
2110310014@stmail.ntu.edu.cn

Chaobo Li  
1811310007@yjs.ntu.edu.cn

Junjie Chen  
cjjcy@ntu.edu.cn

<sup>1</sup> School of Information Science and Technology, Nantong University, 9 Seyuan Road, Nantong 226019, Jiangsu Province, People's Republic of China

<sup>2</sup> Nantong Research Institute for Advanced Communication Technologies, Nantong 226019, Jiangsu, People's Republic of China