DESIGN OF INTELLIGENT VIDEO SURVEILLANCE SYSTEM FOR INTELLIGENT ORCHARD BASED ON DEEP LEARNING NETWORK

Yangyang LIU[1], NiuNiu YIN[1], Yan SUN[1], Leijing CHEN[1.2], Fansheng MENG[3,*], Pengyang ZHANG[1], Huimin REN[1], Ruizhuo FENG[1]

Aiming at the problem of low intelligence level of orchard management in China, this study develops an open intelligent video surveillance system for intelligent orchards based on deep learning target monitoring and behavior recognition algorithms. Based on the open-source distributed framework gRPC, the system improves the main network structure of YOLOv5 and optimizes the abnormal behavior recognition algorithm based on human skeleton. It not only realizes the dynamic allocation of servers and the number of cameras for multi-platform and multi-host communication and maximum utilization of existing computing resources, but also improves the recognition rate of small and partially occluded targets in multi-channel video and reduces the amount of network computations while ensuring the real-time monitoring accuracy of abnormal behavior. The test results show that the monitoring system's accuracy is higher, the illegal operation monitoring accuracy is over 91.25%, and the abnormal behavior monitoring accuracy is above 61.78%. The surveillance system can meet the monitoring needs of intelligent orchards, reduce the cost of orchard management, and realize the scale and intelligent management of orchards, which is of great significance to promote the upgrading of orchard industry.

Keywords: Intelligent orchard; Target monitoring; Behavior recognition; Video surveillance; Deep learning network

1. Introduction

The intelligent orchard is a modern agricultural production mode based on video monitoring network, information technology, automatic control technology, intelligent equipment and facilities, etc. It is an inevitable choice for the development of a modern orchard [1-3]. However, conventional video surveillance systems lack automatic recognition and early warning mechanisms. Additionally, they require a lot of manpower to monitor the images when security personnel are needed, making it impossible for them to respond to accidents in a timely manner [4]. Intelligent video surveillance embedded computer vision algorithms, deep

[1] School of Engineering, Anhui Agricultural University, Hefei, Anhui, China;
2 Electronic information School, Sichuan University, Chengdu, China
3 School of mechanical engineering, Yangzhou University, Yangzhou, China
* Corresponding author: Fansheng MENG, master, engineer, mfs21721277@163.com

learning technology, and alarm mechanisms into the surveillance system, which can monitor abnormal situations and has important application value for the development of intelligent orchards [5].

At present, many image processing algorithms based on deep learning are composed of various convolutions, and the convolution operation consumes a lot of computing resources [6]. For the massive data output by the video surveillance system, a single server can no longer meet system requirements. An effective way to address this issue is to use distributed computing technology to connect multiple servers or embedded computing platforms to the video surveillance system, and flexibly configure system computing resources [7, 8]. Domestic and foreign scholars have made some progress on video surveillance. The Hadoop framework developed by Apache Foundation has excellent storage and management capabilities of cluster resources and can realize distributed storage of massive data with high scaling capability [9, 10]. However, this framework uses a number of nodes to maintain the data volume, resulting in high delay and poor real-time data processing performance [11]. Ji et al. [12] proposed a distributed high concurrency server system, which uses distributed deployment of multiple types of servers to change the server cluster mode and improve the concurrency of data requests. Joydip et al. [13] developed the distributed framework gRPC to realize the serialization and deserialization of data transmission between systems and improve the real-time performance of services. However, different intelligent analysis algorithms are required in different scenarios and have high storage and computing resource requirement. Therefore, optimization is required in terms of resource allocation such as computing power. In target monitoring, the network proposed by Zhang et al. [14] can learn from images to extract sample images for feature labeling, saving a large amount of dataset production costs. However, this method is a single-stage monitoring framework, which cannot separate monitoring and proposal, resulting in low accuracy and recall rate. Liu et al. [15] monitored targets using a two-level target monitoring method, and pre-processed regional proposals before monitoring to improve accuracy and recall. Region-based Convolutional Neural Networks (RCNNs) solve the target monitoring problem as a regression problem, allowing the image to infer the target category, location coordinates, confidence level, and other information through one inference [16]. However, this method has a complex structure and is time-consuming in monitoring. YOLO series of algorithms, with their simple structure and ability to quickly detect objects, are not as accurate as the two-step monitoring methods based on candidate regions [17]. In behavior recognition, Wang et al. [18] applied the Model-Agnostic Meta-Learning （MAML） method to train the single-target monitoring model as a tracker, which could realize online updating. Orchard operations often involve multiple people simultaneously, so the single target monitoring function cannot meet the needs of intelligent orchards [19, 20]. Wu et al. [21] proposed a video behavior recognition method of spatio-temporal correlation with depth, independent of perspective. However, this study requires high computing power support, which will increase the complexity of the orchard’s video monitoring system. Yan et al. [22] proposed a new behavior recognition algorithm based on Spatial Temporal Graph Convolution Networks (ST-GCNs), which recognized human behaviors using a spatio-temporal GCN and achieved a high recognition rate in the standard action recognition dataset. However, the operation of an orchard is non-standard, so it cannot be directly applied to intelligent video monitoring of intelligent orchards.

In this study, the open-source distributed framework is used to design the monitoring system, and different algorithms are used to analyze the different scenes of camera surveillance. The structure of YOLOv5 network is improved by adding an attention mechanism module, which reduces the number of network parameters and improves the target recognition accuracy. The abnormal behavior identification method based on the skeleton node information is then designed, the relevant source code of OpenPose is developed and modified to convert video data into text information of the node, and the relevant human behavior dataset is made to realize the monitoring and warning function of the intelligent orchard.

2. Materials and methods

The purpose of the intelligent video monitoring system of the intelligent orchard in this study is to monitor the illegal operations and abnormal behaviors that may occur during orchard management. The software architecture of the system is implemented based on the distributed RPC (Remote Procedure Call) framework [23]. It realizes the camera login, display, and management functions, reasonably allocates computing tasks to the corresponding server, and realizes the function of multi-algorithm task for parallel processing of multi-channel surveillance video.

2.1 Abnormal target monitoring

2.1.1 Optimization of YOLOv5 target monitoring network

YOLOv5 target monitoring network can output target category location and degree of confidence at the same time, with high accuracy and speed [24], as shown in Fig. 1.

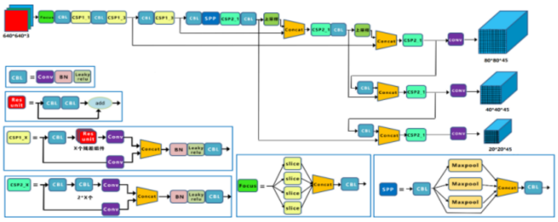
640*640*3
Focus
CBL
CSP1_1
CSP1_3
SPP
CSP2_1
上采样
Concat
CONV
80*80*45
40*40*45
20*20*45
CBL = Conv BN Leaky relu
Res unit = CBL CBL add
CSP1_X = CBL Res unit Conv Conv Concat BN Leaky relu CBL
X个残差组件
CSP2_X = CBL CBL Conv Conv Concat BN Leaky relu CBL
2*X个
Focus = slice slice slice slice Concat CBL
SPP = CBL Maxpool Maxpool Maxpool Concat CBL

Fig. 1. YOLOv5s network structure diagram.

As the application scene of this study is a complex orchard, with more occlusions and higher camera resolution, the target size is smaller after entering the target monitoring network for image scaling. YOLOv5s has a low recognition rate for partially occluded targets and small targets. In this study, the Channel Attention mechanism module (CA) is introduced [25]. The SE (Squeeze-and-Excitation Networks) module [26] is added to the YOLOv5 target monitoring network to selectively enhance useful channel feature information and suppress useless information. The SE module mainly consists of Squeeze and Excitation. The Squeeze operation is used first. As shown in (1), the global average pooling is performed first. The Excitation operation is then carried out, as shown in (2). The dependence on each channel is learned, and then the scale adjustment of the feature map is carried out according to the dependence level to obtain the original size feature map, which mainly includes Rectified Linear Unit (ReLU) and Sigmoid operations. The process is shown in Fig. 2. A feature F of H×W×C (H×W represents height×width) is first input, and global maximum pooling and average pooling of a space are respectively carried out to obtain two channels of 1×1×C. They are then sent into a two-layer shared neural network, with the number of neurons in the first layer being C/r, the activation function being ReLU, and the number of neurons in the second layer being C. After adding the two features, the weight coefficient Mc is obtained by Sigmoid activation function. Finally, Mc × input feature F produces the new scaled feature.

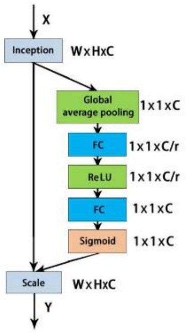

Fig. 2. SE module structure diagram.

$$F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \,, \tag{1}$$

$$F_{\mathrm{ex}}(z,W) = \sigma(g(z,w)) = \sigma(W_2 \delta(W_1 z)) \,, \tag{2}$$

where σ and δ are the Sigmoid and ReLU activation functions, respectively.

This method will increase the number of network parameters. In order to improve the model loading speed, it is necessary to compress the number of model parameters without affecting the monitoring effect. In this study, Ghostbottleneck is used instead of Cryptoservice Provider 1(CSP1) structure to realize feature graphs obtained with a small number of nonlinear convolutions, and a smaller model is produced by eliminating redundant features [27].

2.1.2 Behavior recognition network optimization based on ST-GCN

ST-GCNs (Spatio-Temporal Graph Convolutional Networks) models the dynamics of human skeleton based on the time series of human node positions, and monitor human behaviors by analyzing the changes of node coordinate vectors over time [28]. In this study, the skeleton of each person in each frame of the video is constructed in a certain order, so that the skeleton constitutes a time series. Represented by a set of matrices, as shown in (3), each node information is composed of two-dimensional coordinates and the degree of confidence of each node, as shown in (4):

$$V = \{v_{ti} \mid t = 1,\ldots,T, i = 1,\ldots,N\}, \tag{3}$$

$$F(v_{ti}) = (x, y, c), \tag{4}$$

where T is the number of consecutive frames, N is the number of skeleton joints of the human body, F is the information of the node, x is the key point, Y is a two-dimensional coordinate, and c is the confidence level, that is, the credibility of the measured parameters.

In ST-GCN, ST represents space and time. The overall network framework is shown in Fig. 3.

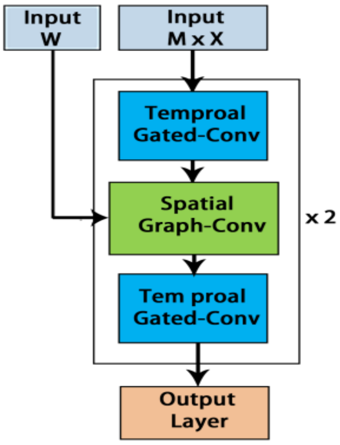

Fig. 3. ST-GCN network framework.

The network input is the eigenvector *X* and the adjacency matrix *W* of the graph of *M* skeleton nodes, and each skeleton is composed of *N* points. The characteristic information at time *T* is obtained through two space-time convolution modules and an output layer. Each space-time module contains a time domain convolution block and a spatial domain convolution module. Time domain convolution carries one-dimensional convolution according to the information of each node and time as the convolution dimension, as shown in (5). Y_T is then obtained by activating the gating mechanism activation function (GLU), as shown in (6). GLU is given in (7), where *w*, *b*, *v*, and *c* are all learnable parameters.

$$[PQ] \in R^{(M-K_T+1)\times 2}, \tag{5}$$

$$Y_T \in R^{(M-K_t+1)\times n}, \tag{6}$$

$$f(x) = (X * w + b) * (X * v + c) \quad . \tag{7}$$

Spatial domain convolution is carried out on the graph at every moment, independent of time, and the network input is $X \in R^n$, *n* is the number of skeleton nodes, and then Chebyshev graph convolution is carried out to obtain the characteristic results Y_S, as shown in (11) where $T_i(L)$ is Chebyshev polynomials [29] in (8), convolution kernel in (9), and Chebyshev graph convolution in (10).

$$\begin{cases} T_0(x) = 1 \\ T_1(x) = x \\ T_{n+1}(x) = 2xT_n(x) - T_{n\text{-}1}(x) \end{cases}, \tag{8}$$

$$\begin{cases} g\theta^{'}(\Lambda) \approx \sum_{k=0}^{K} \theta^{'} T_k(\tilde{\Lambda}) \\ \tilde{\Lambda} = \frac{2}{\lambda_{\max}}\Lambda - I_N \end{cases} \tag{9}$$

$$\begin{cases} g\theta^{'} * X \approx \sum_{k=0}^{K} \theta^{'}{}_k T_k(\tilde{L})X \\ \tilde{L} = \frac{2}{\lambda_{\max}} L - I_N \end{cases} \tag{10}$$

$$Y_S = \sum_{i=0}^{K-1} \theta_i T_i(L) X \tag{11}$$

The output layer contains a time domain convolution layer and a fully connected layer. According to the time domain convolution calculation module, each through time domain convolution of a one-dimensional convolution, the data on the time dimension reduces by $2(K_t - 1)$, before the output layer outputs $Y \in R^{(M-4(K_t-1)) \times n \times C_0}$ following two space-time convolution modules. The output layer of the time domain convolution kernels is of size $\Gamma \in R^{(M-4(K_t-1)) \times n \times C_0}$, the number of convolution kernels is C_O, and the outputs are $\hat{v} = Zw + b, Z \in R^{n \times C_O}, w \in R^{C_O}$ through the full connection layer.

2.1.3 Human joint point extraction based on Openpose

The behavior recognition function of the study is realized by two deep learning networks, Openpose and ST-GCN. Openpose human skeleton node monitoring network is developed based on Caffe framework, which can extract posture information such as human movement, facial expression, and finger movement, and is applicable to single or multi-person situations [30]. Openpose is a bottom-up monitoring algorithm, which first monitors all the nodes in the image, then distributes the monitored nodes to the corresponding people according to their correlation, and finally connects all the nodes in a certain order to form the human skeleton. The algorithm extracts features through VGG-19, and then enters multiple CNN modules through two branches to obtain the degree of confidence information and association information of the closed nodes respectively. The correlation of the closed nodes is represented by Part Affinity Field (PAF) [31]. Finally, the coordinates and confidence level of all people's joints in the image are obtained.

ST-GCN performs behavior inference classification through the node data of continuous video frames. The higher the monitoring accuracy is, the more frames the network input is, and the more computation is required. In this study, the maximum number of network input frames is set to 300. When there are more than 5 people in each frame, only the 5 people with the highest average confidence level are selected for calculation. In order to verify the accuracy of the Openpose+ST-GCN optimization network studied in this paper for violation monitoring, so this study selects at least five people with the highest average confidence in each frame of the video for calculation, so as to ensure that the four behaviors including climbing, running, throwing, and smoking can be calculated and monitored within the set maximum network input frame of 300 frames. When a small number of people with high confidence level is selected for each frame, it will make the feature dimension in the picture too small and the model too simple, resulting in the fitted function not satisfying the training set error and causing the underfitting phenomenon, while too many people will make too many feature latitudes, too complex model assumptions, too many parameters and too much noise. Different accuracy and speed are obtained by changing the number of video frames contained by different chips to reduce the amount of calculation, as shown in Table 1.

Table 1

Performance comparison of different frame numbers

Frame numbers	mAP (%)	FPS
50	47	83
100	71	77
150	86	61
200	88	55
250	90	50
300	90	44

It can be seen from Table 1 that the mean average precision (mAP) is positively correlated with frame number, while FPS is negatively correlated. To meet the overall performance of the system, the number of frames entered by the network needs to be adjusted according to the computing GPU usage of the server. The specific adjustment method is shown in (12):

$$\begin{cases} N = \dfrac{50}{R^2}, R > 0.5 \\ N = 230, R <= 0.5 \end{cases} \tag{12}$$

where, R is GPU computing resource occupancy rate (%) and N is the number of frames.

According to (12), the number of frames in the input video is set in the range between 50 and 230, which can dynamically match the accuracy rate, FPS, and GPU computing resources and improve the overall performance of the system.

Behavior recognition has a high requirement for computing power resources. In order to improve the loading efficiency of model files, the algorithm model loading program is separated from the inference program in this study, and the corresponding model file is loaded into GPU when the system runs the initialization code. The cameras that the system uses are webcams. To reduce the coupling between the behavior recognition algorithm in this work and other programs, this study combines the video stream reading, decoding, and other programs with the proposed algorithm. The camera IP address, data stream format, password, and other information are directly input to get the current video behavior monitoring results. After processing, it is sent to the client via RPC for overlay display.

2.2 Establishment of target data setting

According to the orchard scene characteristics and operation specifications, this study monitored 5 types of operation scenarios of daily orchard management, as shown in Table 2.

Table 2

Daily management abnormal target monitoring requirements

	Pruning	Fertilization	Application of Pesticide	Bagging	Picking
Overalls	√	√	√	√	√
Caps	√	√	√	√	√
Gloves	√	√	√	√	√
Masks	√	√	√	√	√
Climbing	×	×	×	×	×
Running	×	×	×	×	×
Throwing	×	×	×	×	×
Smoking	×	×	×	×	×

Orchard management dataset is created by field shooting and labeling of orchards. The label file used in this study is XML format of VOC dataset, and LableImg, an open-source and convenient annotation tool, is used for annotation. Each image corresponds to a tag file that describes the category, location, and size of the object in the image.

2.3 Test Scheme

2.3.1 Violation operation monitoring test

In order to test the superiority of the YOLOv5 optimized network model in this study, SSD, YOLOv3, YOLOv5 and optimized YOLOv5SE models are adopted in this paper. The model performance comparison test of self-built dataset is conducted first, and then the violation operation monitoring test is conducted. The monitoring test of illegal operation was divided into five groups: pruning, fertilization, application of pesticide, bagging, and picking. The monitoring accuracy is compared according to the four operating standards of wearing overalls, working caps, gloves, and masks in each group. Three hundred photos are selected for each illegal operation.

The dataset is divided into training and test sets according to the ratio of 9 to 1. Three indicators, mAP, FPS, and recall rate, are used to evaluate model performance. mAP is the average accuracy for all categories and FPS is the number of frames processed per second. Frames Per Second (FPS) considers the entire processing time including image reading, cropping, overlaying, and displaying of prediction frames, etc. Recall rate is the ratio of the number of correctly detected targets to the total number of targets in the actual image.

During the test, the computer operating system was Ubuntu18.04, the CPU was Intel (R) Core (TM) I7-9700 CPU@3.00GHz ×8, the memory size was 32GB, and the display card was the dual-channel GeForce RTX 2080TI 12GB. The deep learning PyTorch framework was used to build the optimized YOLOv5 model. As for the parameter settings, the number of epochs was set to 300, the batch size was 16, the input image dimensions were 640×640, and the initial learning rate was 0.01. The warMP-UP method was used to adjust the learning rate, with a momentum of 0.973 and weight decay of 0.0005.

2.3.2 Abnormal behavior monitoring test

In order to verify the superiority of abnormal behavior monitoring model, four kinds of behaviors including climbing, running, throwing, and smoking are monitored by C3D, Alphapose+LSTM, Openpose+ST-GCN, and Openpose+ST-GCN optimization network in this study, respectively, and the monitoring accuracy is determined and compared. Each behavior is tested with 300 photos.

In this study, each video is transformed into a picture, by segmenting the video into images by the length of a single frame, the joint point information and behavior category information with high confidence in the images are selected and the dataset is divided into training and test sets in a ratio of 9 to 1. Then, label information is added to each file according to the category, so that each JSON file contains the human body node information with high degree of confidence in the video and the behavior category information to which the video belongs. Specific performance was tested through 10 videos containing 0-6 people with a size of 1920 ×1080.

Training parameter configuration: the batch size is 64, that is, 64 video clip files are read at the same time, the number of epochs is 100, and the initial learning rate is 0.1. Data picture set in network format: the number of epochs is set to 16, and the initial learning rate to 0.005. When the epoch number is a multiple of 4, the learning rate is reduced to 10% of its value.

In this study, the overall skeleton of a human is represented by 18 nodes, and each node contains a two-dimensional coordinate and a confidence level. The identification accuracy of human joints directly affects the final behavior identification accuracy. When there are many unusable data in the training set, the monitoring effect of behavior identification model will be seriously affected. Because the number of people in each frame and the degree of confidence of each node are different, there are partial occlusion of human body and background noise. In this study, the following four filtering principles are proposed for data extraction and preservation to filter data and avoid false detection: (1) Save the key data of the two people with the highest confidence levels. (2) If the occlusion is more than half of the total data, the data is considered invalid. (3) If the position difference of the same joint point between adjacent frames is too large, the data is considered invalid.

As shown in Formula (13), when D is greater than the threshold value, this data is considered invalid:

$$\begin{aligned} D &= \sum_{\mathrm{i}=1}^{K} d_i \\ d &= |x_1 - x_2| + |y_1 - y_2| \end{aligned} \tag{13}$$

where, D is the distance of the node (mm). (4) When the confidence of all key points in the human body is lower than the threshold, the data is not available.

The original Openpose network only extracts the joint data of a single picture, while the data in this study is the node data corresponding to continuous video frames. Data filtering, formatting, and preservation modules are added on the basis of the source code, and the resulting process is shown in Fig. 4.

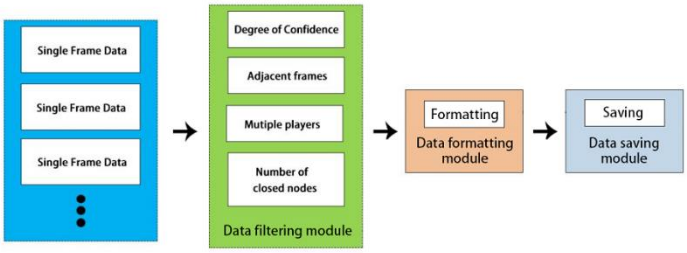

Fig. 4. Data extraction of key nodes.

3. Results and Discussion

3.1 Test results and discussion of violation operation monitoring

Table 3

Network performance comparison of self-built datasets

Model	Size	FPS	mAP (%)	Recognition Rate	Parameters
SSD	300×300	79	88.23	82.1	76.3MB
YOLOv3	640×640	72	93.13	91.64	256MB
YOLOv5	640×640	153	95.15	91.42	52.8MB
YOLOv5SE	640×640	142	95.43	92.47	31.3MB

As can be seen from Table 3, YOLOv5 has great advantages in accuracy, recognition rate, and model parameters compared with SSD and YOLOv3 networks. Compared with the original YOLOv5SE, the optimized YOLOv5SE has improved the mAP and recognition rate, increased the recognition rate of small targets and occlusion targets and can more accurately monitor small targets. In addition, the number of model parameters can be reduced to meet the real-time requirements of intelligent video surveillance algorithms. It is proved that the method of algorithm optimization is feasible and effective.

Table 4

Monitoring results of illegal operations

Groups	Craft	Irregularities	SSD	YOLOv3	YOLOv5	YOLOv5SE
1	Pruning	Without overalls	88.27	92.23	94.19	95.36
		Without caps	88.04	92.15	94.01	95.32
		Without Masks	86.65	90.18	92.66	93.95
		Without gloves	81.59	86.79	90.12	91.31
2	Fertilization	Without overalls	88.32	92.33	94.27	95.48
		Without caps	88.13	92.26	94.10	95.37
		Without Masks	86.66	90.29	92.74	94.11
		Without gloves	81.68	87.05	90.84	92.09
3	Applation of Pesticide	Without overalls	88.34	92.34	94.29	95.51
		Without caps	88.15	92.28	94.11	95.37
		Without Masks	86.65	90.29	92.75	94.10
		Without gloves	82.69	87.07	90.88	92.15
4	Bagging	Without overalls	88.18	92.20	94.15	95.35
		Without caps	88.07	92.21	94.01	95.30
		Without Masks	86.55	90.19	92.68	93.96
		Without gloves	82.06	86.77	90.10	91.31
5	Picking	Without overalls	88.13	92.14	94.13	95.35
		Without caps	88.05	92.08	92.98	95.28
		Without Masks	86.01	90.04	91.55	93.88
		Without gloves	81.53	86.65	88.89	91.25

It can be seen from Table 4 that in the experiments of pruning, fertilization, application, bagging, and picking, the monitoring accuracy of YOLOv5SE model after overall optimization is the highest, the monitoring accuracy of YOLOv5 is the second highest, and the monitoring result of SSD network is the worst. YOLOv5SE was 1.4%, 3.7%, and 7.9% better than YOLOv5, YOLOv3 and SSD, respectively. Therefore, the use of YOLOv5 model is as the basic network is justified, and the proposed optimization method can enhance monitoring accuracy, reduce financial losses, and improve the standardized scale of orchard management.

Violations can easily cause injuries, and every percentage point improvement in the accuracy of monitoring results can prevent a great deal of unnecessary human and financial losses. In general, the YOLOv5SE model focuses on the four types of operation specifications without overalls, without working caps, without gloves and masks. Each group had the highest monitoring accuracy for the illegal operation without overalls, especially in the process of fertilization and application, up to 95.51%. Because the uniform area is large, it is difficult to be shielded when workers apply fertilizer and pesticides along the tree line. Whereas pruning, bagging, and picking operations require workers to go between tree branches, especially in the picking. Fruit trees have the lushest growth, providing more shelter. As a result, the monitoring accuracy is relatively low. The monitoring accuracy of illegal operation without gloves and masks is low, especially the monitoring accuracy of illegal operation without gloves in picking operation, which is only 91.25%, mainly due to the hand movement between branches where there is more cover.

3.2 Abnormal behavior monitoring test results and discussion

Table 5

Network performance comparison.

Model	mAP	Whether to support multiple players	FPS	Storage resource occupation
C3D	71.4	F	28.3	10.9GB
Alphapose+LSTM	47.3	T	11.1	5499MB
Openpose+ST-GCN	55.4	T	14.3	2974MB
Openpose+ST-GCN optimizing	67.1	T	14.3	2974MB

As can be seen from Table 5, the C3D model has the worst effect, occupies the most memory resources, and does not have the ability of simultaneous monitoring of multi-person behaviors. However, other networks can monitor multi-person behaviors simultaneously. In addition, the average accuracy of the proposed Openpose+ST-GCN optimized model is high when multi-person simultaneous monitoring is conducted. It is proved that the suggested method of algorithm optimization is feasible and effective.

Table 6

Abnormal behavior monitoring results

Abnormal behavior	Monitoring accuracy/%		
	Alphapose+LSTM	Openpose+ST-GCN	Openpose+ ST-GCN optimizing
Climbing	41.33	49.46	61.78
Running	41.51	51.12	63.53
Throwing	40.07	49.88	62.85
Smoking	38.89	48.15	61.91

According to Table 6, although the accuracy of abnormal behavior monitoring is generally low, the Openpose+ST-GCN optimized model monitoring accuracy is the highest. Before optimization, the monitoring accuracy of Openpose+ST-GCN network is 9.2% higher than that of Alphapose+LSTM network. It proves that the basic algorithm selected in this study has good accuracy. After the proposed optimization of Openpose+ST-GCN network, the monitoring accuracy is improved by 12.87%, with a large increase range. It proves that the effect of the proposed optimization on the algorithm is relatively significant. In particular, the accuracy of smoking behavior monitoring increased by 13.76%. There are many leaves in the orchard, and smoking can easily cause fire. The proposed model optimization improves the accuracy of smoking behavior monitoring, which is very important for orchard safety. The optimized Openpose+ST-GCN network has the highest monitoring accuracy of running behavior, which is 63.53%. Orchard running can easily cause safety accidents, so the optimized algorithm can strengthen the orchard safety management and prevent human and financial losses.

4. Conclusions

Based on the construction of distributed software framework for video surveillance systems and research on intelligent algorithm design, this study realizes the monitoring functions of illegal operation and abnormal behavior in the exclusive scene of the orchard. Based on YOLOv5, this study introduces the channel domain attention mechanism module to optimize and improve the recognition rate of small and occlusion targets. Besides, the ST-GCN network structure is used as the basic network of behavior recognition, and the behavior recognition method based on deep learning is designed to realize behavior recognition combined with Openpose extraction of human key points. The orchard video monitoring network based on a distributed framework can dynamically change the ST-GCN model input video frames according to service computing resources, separate algorithm model loader and inference program, achieve the balance between speed and accuracy, and improve behavior recognition performance.

Through the data collection and annotation of scene and working process, the orchard target dataset is created. The feasibility and effectiveness of this study on the optimization of the monitoring network and behavior recognition algorithm for violation operation are verified through performance comparison tests using the self-built dataset. The optimization algorithm is verified using illegal operation and abnormal behavior monitoring tests. It can provide accurate monitoring of illegal operation and abnormal behavior, which meets the demands of intelligent orchard for daily management of video monitoring. It also provides a reference for reducing the cost of orchard management and realizing the development of orchard scale and intelligent management.

REFERENCES

[1]. Y. Lin. Design of the application system to smart leisure agriculture based on the Internet of Things. Office Informatization, 2020. 25: 24-26.

[2]. H. Wu. Design of water and fertilizer integration system based on Internet of Things technology. Journal of Heilongjiang Vocational Institute of Ecological Engineering. 2019, 32: 37-38.

[3]. Z. Zhou, J. Zhang, H. Guo, et al. Development and Testing of Intelligent Sensing and Precision Proportioning System of Water and Fertilizer Concentration. Smart Agriculture, 2020, 2: 82.

[4]. U. Gawande, K. Hajari, Y. Golhar. Pedestrian detection and tracking in video surveillance system: issues, comprehensive review, and challenges. Recent Trends in Computational Intelligence, 2020: 1-24.

[5]. H. Luo, J. Liu, W. Fang. P.E.D. Love, Z. Lu. Real-time smart video surveillance to manage safety: A case study of a transport mega-project. Advanced Engineering Informatics, 2020, 45: 101100.

[6]. K. Rezaee, S.M. Rezakhani, M.R. Khosravi, M.K. Moghimi. A survey on deep learning-based real-time crowd anomaly detection for secure distributed video surveillance. Personal and Ubiquitous Computing, 2021: 1-17.

[7]. A.B. Mabrouk, E. Zagrouba. Abnormal behavior recognition for intelligent video surveillance systems: A review. Expert Systems with Applications, 2018, 91: 480-491.

[8]. G.F. Shidik, E. Noersasongko, A. Nugraha, et al. A systematic review of intelligence video surveillance: trends, techniques, frameworks, and datasets. IEEE Access, 2019, 7: 170457-170473.

[9]. J.M. Wu, G. Srivastava, M. Wei, et al. Fuzzy high-utility pattern mining in parallel and distributed Hadoop framework. Information Sciences, 2021, 553: 31-48.

[10]. X. Zhu, J. Wang. Research on source camera recognition method based on Hadoop. Modern Computer. 2020, 18, 88-92.

[11]. G. Zhang, M, Ye, Z. Wang, T. Zhou. Comparison and implementation of Core Technologies of Big Data Hadoop framework. Research and Exploration in Laboratory, 2021, 40, 145-148+176.

[12]. P. Ji, D. He. Design and implementation of distributed High concurrency server based on Erlang/OTP. Information & Computer (Theory Edition), 2015, 10, 29-31.

[13]. Joydip. K. How to build gRPC applications in ASP.NET Core. InfoWorld.com. 2020.

[14]. S. Zhang, L. Wen, X. Bian, et al. Single-Shot Refinement Neural Network for Object Detection. IEEE, 2018.

[15]. L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, et al. Deep Learning for Generic Object Detection. A Survey, 2019.

[16]. Y. Xiao, Z. Tian, J. Yu, Y. Zhang, S. Liu, S. Du, X. Lan. A review of object detection based on deep learning. Multimedia Tools and Applications, 2020, 79: 23729-23791.

[17]. W. Wang, G. Jiang, Y. Chu, Y. Chen. Review of Object Detection System from RCNN to YOLO series. Journal of Qilu University of Technology, 2021, 35, 9-16.

[18]. G. Wang, C. Lou, Z. Xiong, and W. Z. Tracking by Instance Detection: A Meta-Learning Approach. IEEE, 2020: 6288-6297

[19]. J. Peng, C. Wang, F. Wan, et al. Chained-Tracker: Chaining Paired Attentive Regression Results for End-to-End Joint Multiple-Object Detection and Tracking. Computer Vision – ECCV 2020, 2020, 145-161.

[20]. G. Ciaparrone, F.L. Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, and F. Herrera. Deep Learning in Video Multi-Object Tracking: A Survey. Neurocomputing, 2020, 381: 61-88.

[21]. P. Wu, X. Yang, B. Mao, L. Kong, Z. Hou. A Perspective-independent Method for Behavior Recognition in Depth Video via Temporal-spatial Correlating. Journal of electronics & information technology, 2019, 41, 904-910.

[22]. S. Yan, Y. Xiong, D. Lin. Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition. Thirty-second AAAI conference on artificial intelligence, 2018.

[23]. Y. Wu, Y. Guang, S. He, M. Xin. An industrial-based framework for distributed control of heterogeneous network systems. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2018, 50: 2120-2128.

[24]. W. Li, T. Zhu, X. Li, J. Dong, J. Liu. Recommending Advanced Deep Learning Models for Efficient Insect Pest Detection. Agriculture, 2022, 12(7): 1065.

[25]. C. Li, J. Ma, S. Zhao. A workshop personnel detection method based on spatial domain attention mechanism. Journal of Harbin University of Science and Technology, 2022, 27: 7.

[26]. A. Diba, M. Fayyaz, V. Sharma, et al. Temporal 3D ConvNets: New Architecture and Transfer Learning for Video Classification. arXiv preprint arXiv, 2017, 1711: 08200.

[27]. M. Defferrard, X. Bresson, P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. Advances in neural information processing systems, 2016, 29.

[28]. X. Shi, J. Huang, B. Huang. An Underground Abnormal Behavior Recognition Method Based on an Optimized Alphapose-ST-GCN. Journal of Circuits, Systems and Computers, 2022: 2250214.

[29]. H. Jie, S. Li, S. Gang, S. Albanie. Squfeeze-and-excitation networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 99.

[30]. T.N. Kipf, M. Welling. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv, 2016, 1609: 02907.

[31]. C.Sawant. Human activity recognition with openpose and Long Short-Term Memory on real time images. EasyChair Preprint, 2020, 2297.