Journal of Software Engineering and Applications, 2014, 7, 452-460
Published Online May 2014 in SciRes. http://www.scirp.org/journal/jsea
http://dx.doi.org/10.4236/jsea.2014.75042

A Recognition Method of Pedestrians' Running in the Red Light Based on Image

Min Zhang, Chao Li Wang, Yun Feng Ji

Department of Control Science and Engineering, University of Shanghai for Science and Technology, Shanghai, China
Email: mixue.2012@163.com, clwang@usst.edu.cn, jyf123456789@126.com

Received 4 April 2014; revised 4 May 2014; accepted 12 May 2014

Abstract

It is dangerous for pedestrians to run when the traffic shows a red light, but in some cases the pedestrians are breaking the rules. This system will be a meaningful thing if the jaywalking behaviors of pedestrians in the road crossing through the monitoring cameras could be recognized. Then drivers can be informed of the situations in advance, and they can take some actions to avoid an accident. The characteristic behavior is the non-construction, and furthermore, due to the change of sunlight, temperature, and weather in the outside environment, and the shaking of cameras themselves, the background images will change as time goes by, which will bring special difficulties in recognizing jaywalking behaviors. In this paper, the method of adaptive background model of mixture Gaussian is used to extract the moving objects in the video. On the base of Histograms of Oriented Gradients (HOG), the pedestrians images and car images from MIT Library are used to train our monitoring system by SVM classifier, and identify the pedestrians in the video. Then, the color histogram, position information and the movement of pedestrians are selected to track them. After that we can identify whether the pedestrians are running in the red lights or not, according to the transportation signals and allocated walking areas. The experiments are implemented to show that the proposed method is effective.

Keywords

SVM Classifier, Histogram, Background Modeling, Objects Tracking

1. Introduction

Due to the rapid development in urban road traffic, the numbers of pedestrians and the vehicles are increasing continuously. And the phenomenon of pedestrians' running in the red light is increasing too. It brings bad influence to the road traffic safety and leads to road traffic jam for all the time. To improve the efficiency of urban

How to cite this paper: Zhang, M., *et al*. (2014) A Recognition Method of Pedestrians' Running in the Red Light Based on Image. *Journal of Software Engineering and Applications*, **7**, 452-460. http://dx.doi.org/10.4236/jsea.2014.75042

traffic and to protect the people, we need to do some work to detect the pedestrians who are running in the red light and give some alerts to them. It can help to improve the consciousness of people and give the drivers some important reminder to avoid traffic accidents. This is an economic and effective method to use the surveillance video of the modern transportation network to detect the pedestrians running in the red light.

We can use the frame difference to extract moving objects in the video. Though it has a good real-time performance to get the moving objects by subtracting two adjacent images, it is too sensitive to the environmental noise. The selected threshold has great impact on the detected results. For the big and color unanimous moving objects, it is easy to get internal voids from the detected objects, then it is hard to extract the complete moving targets. Optical-flow method can estimate the moving field based on the temporal and spatial gradients of the image sequence, then it can distinguish the moving objects and the background, but it is susceptible to the outside environment influence and it needs some complex mathematical calculation. Background subtraction is to get the difference from the background image to the current image, so if we want to extract the moving objects, we need to acquire the background first. Median filter, linear filter, linear Kalman filter and the Stauffer's mixture Guassian [1] model are all adaptive background models.

At the moment, recognizing based on the gait characteristics of moving is a method for identifying pedestrians. The human's waking gait has a certain periodicity, so we can analyze the periodicity of the video sequence to distinguish the pedestrian and other objects. Wohler [2] had adopted this feature to combine the adaptive time delay neural network to recognize pedestrians. Dalal proposed using the histograms of oriented gradients [3] [4] feature and combining the support vector machine to recognize pedestrians.

Multi-objects tracking in video sequence can be based on moving detection [5], particle filter [6]. Color, texture and motion information can be combined to track multi-objects [7], and it has achieved good results in moving objects tracking.

We choose the adaptive background model of mixture Guassian to characterize the background and then extract the moving objects from the foreground. HOG descriptors are chosen to describe the moving objects. In this system the linear SVM classifier is used for the training and then to classify the pedestrians. The training images are pedestrian images and car images from MIT Library. For the extracted pedestrian targets, color histograms, location and trajectory were used to construct matching matrix to track the objects. After pedestrians are detected and tracked, we can judge whether the pedestrians are running when the red light is on or not, according to traffic signal and alarmed area.

Three main sections of this system are extracting moving objects, the recognition of pedestrians, pedestrians tracking and behavior judgment. The execute solution frame is showed in **Figure 1**.

2. Extract the Moving Objects

The background of the video images will change frequently as the outdoor environments changes, such as light, temperature, and wind. And the camera's shaking also contributes to this factor. So we choose the adaptive background model of mixture Gaussian [1] to cope with the changing environmental conditions. Extract the foreground from the video sequence, and make it binary. Then using morphological processing we can find the connected areas, and extract the moving objects from the foreground.

2.1. Adaptive Background Model of Mixture Gaussian

We build K mixture Gaussian models for every pixel in an image. While a new picture of the video sequence comes, we do the matching processing for every pixel in it. If the pixel matched successfully, it will be classified into background. Otherwise it will be classified as foreground, and we use the information this pixel to update the mixture Gaussian models. The history values of pixel $\{x_0, y_0\}$ is $\{X_1, X_2, \ldots, X_t\} = \{I(x_0, y_0, i): 1 \le i \le t\}$, where $I(x_0, y_0, i)$ represents the pixel value of the location $\{x_0, y_0\}$ at the i^{th} frame of the video sequence. Firstly, we build K mixture Gaussian models based on the history values of the pixels. We always arrange the K Gaussian models as the priority $\rho_{i,t} = \dfrac{\omega_{i,t}}{\sigma_i}$ descending. The probability of the current pixel value as X_t is:

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}). \tag{1}$$

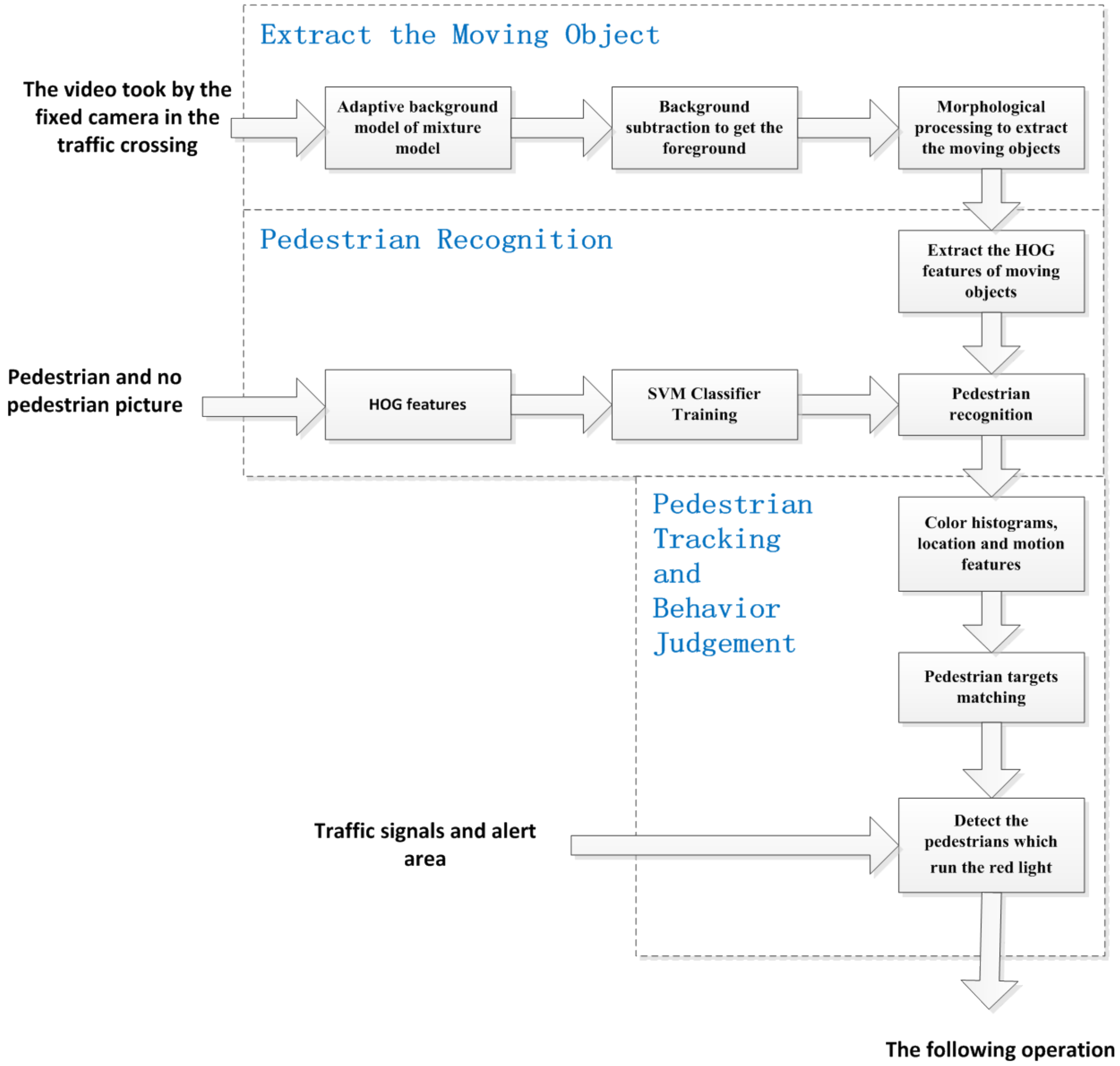
Extract the Moving Object
The video took by the fixed camera in the traffic crossing
Adaptive background model of mixture model
Background subtraction to get the foreground
Morphological processing to extract the moving objects
Pedestrian Recognition
Extract the HOG features of moving objects
Pedestrian and no pedestrian picture
HOG features
SVM Classifier Training
Pedestrian recognition
Pedestrian Tracking and Behavior Judgement
Color histograms, location and motion features
Pedestrian targets matching
Traffic signals and alert area
Detect the pedestrians which run the red light
The following operation

Figure 1. The framework of the system.

where K represents the number of Gaussian models, $\omega_{i,t}\left(\sum_{i=1}^{K}\omega_{i,t}=1\right)$ is the weight corresponding to Gaussian model, and η represents the i^{th} Gaussian model probability density function:

$$\eta(X_t,\mu,\Sigma)=\frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}}e^{-\frac{1}{2}(X_t-\mu)^T\Sigma^{-1}(X_t-\mu)} \tag{2}$$

For an image of RGB channels, we assume that the three channels are independent of each other and have the same variance σ^2, that means $n=3$, the mean value $\mu=\left(\mu_r,\mu_g,\mu_b\right)^T$, and the covariance matrix is $\Sigma=\sigma^2 I$.

Background modeling procedures are as follows:

1) Initializing the mixture Gaussian models (using the median background modeling firstly)

Initialize a bigger variance σ_{init}^2 for every background mixture Gaussian. The weight of every models are initialized as $\omega_{init} = 1/K$. The mean values μ of every mixture Gaussian models are initialized by the pixel value of the first image, so the first image has a great influence on the background models. In our experiment we find that if the first frame varies a lot with the real background, it will take a much longer time to model the background as the real background. To solve this problem, we introduce the median background modeling to initialize the mean values μ. We get the beginning *N* frames of the sequence firstly. And then calculate the median value of every pixel. Initialize the mean values μ as the median values of every pixel.

2) Matching the background models

The K Gaussian models are sorted by descending order priority $\rho_{i,t} = \frac{\omega_{i,t}}{\sigma_i}$. In the order of the Gaussian models, we do the match for the new coming frames. If the pixel value meet the condition that $\left|X_t - \mu_{i,t-1}\right| < 2.5\sigma_{i,t-1}$, then we can consider that it matches successfully according to the Gaussian equation. And the we can set $M_{i,t} = 1$, and set the other models $M_{i,t} = 0$. If none of the models can be matched, we change the last Gaussian model's mean value as the current pixel value and initialize a big variance.

3) Update the mixture Gaussian models

If the model is matched, (*i.e.*, $M_{i,t} = 1$), we should update the mixture Gaussian model as,

$$\begin{aligned} &\mu_{i,t} = (1-\beta)\mu_{i,t-1} + \beta x_t \\ &\sigma_{i,t}^2 = (1-\beta)\sigma_{i,t-1}^2 + \beta\left(x_t - \mu_{i,t}\right)^T\left(x_t - \mu_{i,t}\right). \\ &\beta = \alpha/\omega_{i,t-1} \end{aligned} \tag{3}$$

For all the models update their weights $\omega_{i,t} = (1-\alpha)\omega_{i,t-1} + \alpha M_{i,t}$, where α is model learning rate, and β is parameter learning rate, which react with the parameter convergence speed of the adaptive background model.

4) Background generation:

We choose the former B Gaussian models to generate the background models, which meet $B = \arg\min_b\left(\sum_{k=1}^{b}\omega_k > T\right)$, in our experiments we choose $T = 0.85$. According to the weights of these Gaussian models, get the weighted average of mean values for every Gaussian models as the background. Such as **Figure 2(b)**.

2.2. Foreground Moving Object's Extracting

According to the adaptive background model of mixture Gaussian, we can get the background and foreground of the video sequence, as show in **Figure 2(b)**, **Figure 2(c)**. The foreground is a binary image, the moving object is represented by 1 and the background is represented by 0 in **Figure 2(c)**. We use the morphological processing such as expansion and corrosion to remove the empty in the foreground object and the small noise in it. Then extract the blob block of connected areas. Extracted moving object from the original image is showed in **Figure 2(d)**.

3. The Recognition of Pedestrians

Histograms of oriented gradients (HOG) is a feature descriptor of objected detection in computer vision and image processing. Dalal [4] proposed that, combination of HOG descriptor and the linear SVM classifier will recognize the pedestrians more effectively and we can achieve good results from it. So in our experiments, we choose the HOG descriptor to extract the features of moving objects and use the liner SVM classifier to distinguish the pedestrians from other objects.

3.1. HOD Descriptor

The appearance and shape of the local area of an object can be characterized by the local distribution of local intensity gradients and the directions. The descriptor of histograms of oriented gradients is based on this idea. It's a statistical characteristic of the intensity gradients and directions to the local area of the image.

In our experiments, we get our HOG descriptors using the suggested parameters of reference paper [4] to recognize pedestrians. For an image of 64 × 128 pixels, we divide it into small cells of 8 × 8 pixels. And the ad-

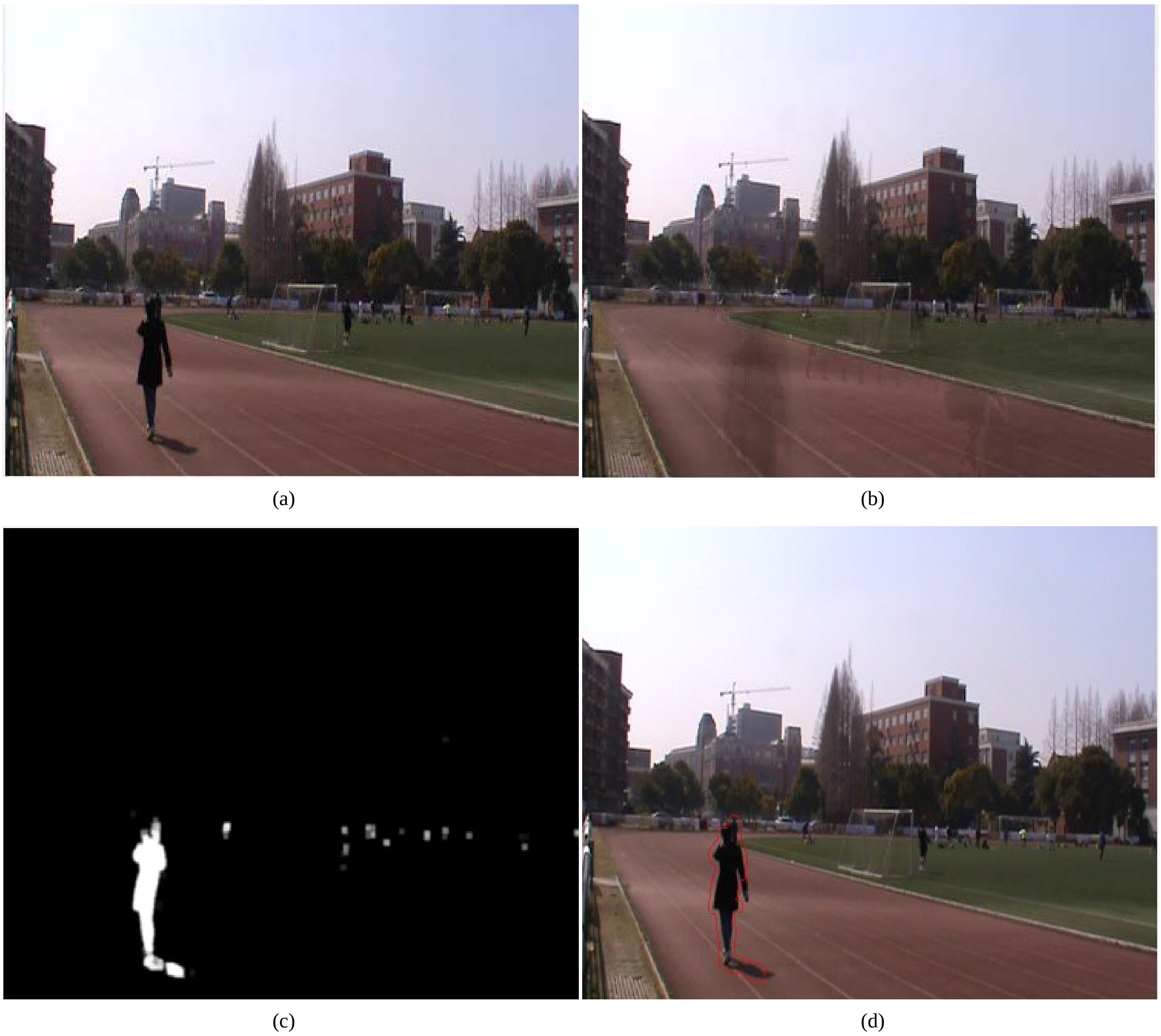

Figure 2. Campus playground: (a) the 40th frame. (b) The background by the mixture Gaussian models. (c) The binary foreground model. (d) The extracted moving object.

jacent four cells combined into one block. The block slides by one cell to the adjacent area once and the entire image is scanned to get the HOG descriptors. The blocks and cells are showed in **Figure 3**. As to the gradients, divide the 1800 into 9 directions (bin) equally. For the 9 directions, get the histograms of a cell as a 9-dimensional feature vector. Combine the four cells of each block into one descriptor of 36 dimensions. And the block slide in one direction by one cell for one time, so we can get 112 blocks in this 64 × 128 image. Combine all the blocks' descriptors into one descriptor to get the final HOG descriptor of 112 × 36 dimensions.

In order to eliminate the effects of illumination, etc. Before we combine all the blocks' descriptors, we need to normalize the descriptor for every block. We adopt the Lowe-style clipped L2 norm. Firstly, L2-norm: $\mathrm{v} \to \mathrm{v} \Big/ \sqrt{\|\mathrm{v}\|_2^2 + \varepsilon^2}$, ε is a very small number to avoid the meaningless for this equation. $\|\mathrm{v}\|_2^2 = \sum v_i^{\,2}$ is the 2 norm of the vector v. Then do the truncation, limit for the maximum value of v as 0.2. Then do the L2-norm again.

The gradient histogram for each cell is calculated by statistical devoting. Divide the 0˚ - 180˚ into 9 bins {(0 - 20), (20 - 40), (40 - 60), (60 - 80),… (160 - 180)}, the center point of every bin value is $\{x_i = 20 * i - 5 \mid i = 1, 2, \cdots 9\}$, which represents the horizontal axis of the histograms. Assume the direction of the gradient α is between x_i and x_{i+1}, and the intensity of the gradient is G, the pixel will influence the bin of i^{th}

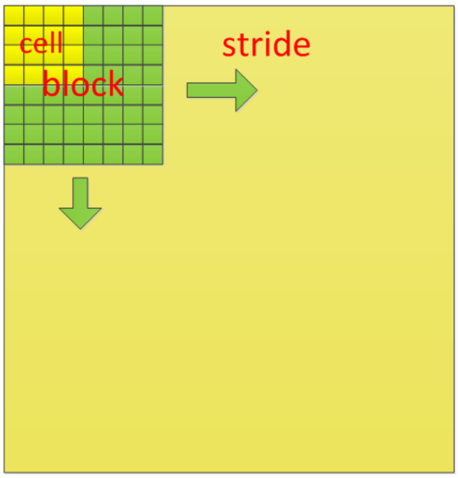

Figure 3. 64 × 128 pixel image, cell (8 × 8) and block (16 × 16), block stride one cell once.

and the (*I* + 1)th. The *i*th bin will be added by $h_i = G*|x_{i+1} - \alpha| / |x_{i+1} - x_i|$ and the (*I* + 1)th bin will be added by $h_{i+1} = G*|\alpha - x_i| / |x_{i+1} - x_i|$.

The gradient is calculated by the template $[-1, 0, 1]$. The horizontal and vertical gradient values of pixel (x, y) are

$$\begin{aligned} G_x &= H(x+1, y) - H(x-1, y) \\ G_y &= H(x, y+1) - H(x, y-1) \end{aligned}. \quad (4)$$

where $H(x, y)$ is the pixel values of (x, y).

The intensity and direction of the gradient are

$$\begin{aligned} G(x, y) &= \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \\ \alpha(x, y) &= \tan^{-1}\left(\frac{G_y(x, y)}{G_x(x, y)}\right) \end{aligned}. \quad (5)$$

For an image of RGB channels, we calculate the gradients for every channel. Choose the biggest intensity gradient vector as the gradient of the pixel.

3.2. Support Vector Machine Classifier (SVM)

Support Vector Machine Classifier is a statistics method for classifying. We choose the penalized parameter C = 0.01 for the linear SVM. In the urban traffic junctions, the moving objects are mainly pedestrians and cars . Other types of moving objects are relatively less than these two types. So we choose the pedestrians' pictures in MIT library as positive training samples and the cars' pictures in MIT library as the negative training samples. Here 924 pedestrians' pictures and 516 cars' pictures were used.

Use the SVM classifier result to identify the extracted moving objects in the video sequence. Distinguish the pedestrians from the moving objects and using the recognition results for the next procedure. **Figure 4** shows the recognition results. The red outline stands for the moving objects extracted by adaptive background model of mixture Gaussian. The green rectangles are the moving objects identified as pedestrian.

4. Pedestrians Tracking and Behavior Judgment

After distinguish pedestrians from other moving objects, we need to track them to get their location and moving direction trend, then judge whether the pedestrian is running the red light or not. The pedestrians' clothes color is almost fixed when they are running the road cross. So the color histograms will be a very good feature for dis-

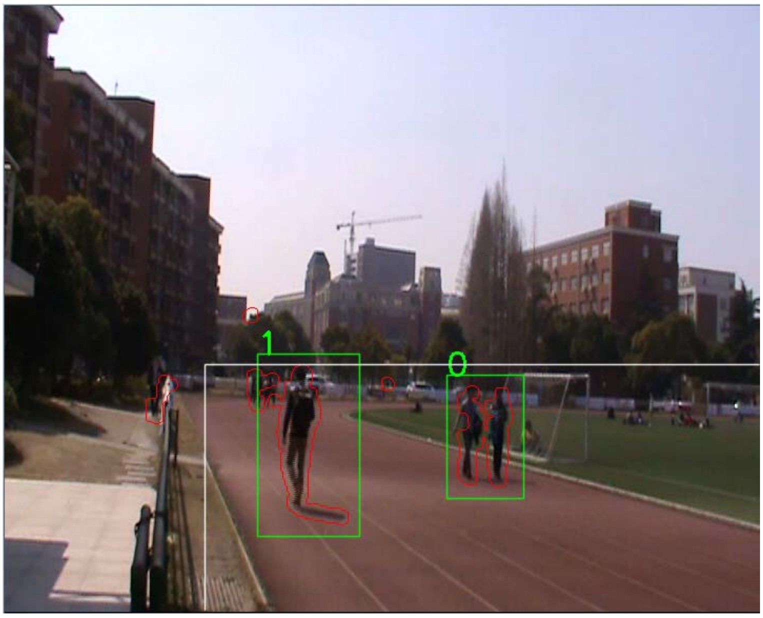

Figure 4. The recognition result of the video sequence.

tinguishing different moving objects as a color feature. The motion of a pedestrian is some kind of continuous in trajectory and space. So we can choose the location and trajectory as the motion features of the pedestrian. We combine the color histograms, location and trajectory of the pedestrian to track the moving pedestrian.

4.1. The Introduction of Tracking Features

Color histograms [7] feature is a statistic feature of the whole image. It uses the global colors' distribution to describe a picture and can represent the image in a global aspect. Pedestrian is no-rigid moving object. Its shape and size will change unexpected in different frames, but its clothes color is almost stationary. And the color histograms can respect a pedestrian's clothes color very well. Since the RGB image cannot perceive the real world color space intuitively and can be easily influenced by sunlight, but the HSV color space is based on the perception of human eyes and can confirm the object's color information. So in our experiment, Initial step is to convert the RGB color image to the HSV color space. And then we used only the color histograms of the H channel. The distance between two objects' color histograms is calculated by Euclidean distance $D(x_1,x_2)=\sqrt{\sum_{i=1}^{N}(x_{1,i}-x_{2,i})^2}$, where i is the index of color histograms, $ix_{1,i}$ is the color histograms' value at index i.

Location means the center coordinate of the moving object, $l_i=(x_i,y_i)$ represent the i^{th} object location. The location distance between two objects is $D(l_1,l_2)=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$.

Trajectory information can be described by the smoothness of direction and velocity, the smoothness of the i^{th} object is

$$S_i=\frac{v_{i,t-1}\cdot v_{i,t}}{|v_{i,t-1}||v_{i,t}|}+\frac{2\sqrt{|v_{i,t-1}||v_{i,t}|}}{|v_{i,t-1}|+|v_{i,t}|}. \qquad (6)$$

where $v_{i,t}$ is the velocity $v_{i,t}=l_{i,t}-l_{i,t-1}$, the different location between the current frame and the former frame. In the equation, the first part is the smoothness of direction and the second part is the smoothness of velocity. We treat them equally, so add them without weights. In the distance matrix, we set trajectory distance as $D=\frac{1}{S_i}$.

4.2. Tracking and Matched Method

We construct a distance matrix $D_{I\times J}$ in this part. Set the targets which has been tracked as the matrix's rows. Set the new identified pedestrians as the columns. According to the multi-object tracking strategy in reference paper [8], we construct the correspond matrix of the distance matrix $M_{I\times J}=0$ as matching matrix.

The matching steps are as follows:

1) For every row of the matrix $D_{I\times J}$, search the position of the smallest element, and in the position of the correspond matrix element is increased by one.

2) For every column of the matrix $D_{I\times J}$, search the position of the smallest element, and in the position of the correspond matrix element is increased by one.

3) If the element in the matrix $M_{I\times J}$ is two, we will believe the corresponded row and column represent objects and match successfully. The other objects haven't been matched are the new objects coming into the image or have left the image.

The element of distance matrix $D_{I\times J}$ is $d_{i,j}$, which represents the distance between the *i*th tracked pedestrian and the *j*th pedestrian in the new frame. It contains the color histograms distance, location distance and trajectory smoothness information. $d_{i,j}=\sum_{f=1}^{3} d_{i,j,f}$, $d_{i,j,f}$ are the normalized distance of the *f*th features, the actual distance is $\hat{d}_{i,j,f}$, normalized method is applied for every column in the distance matrix, $d_{i,j,f}=\dfrac{\hat{d}_{i,j,f}}{\sum_{i}^{I}\hat{d}_{i,j,f}}$ (the distance between the *j*th object in the new frame and all the tracked objects should be calculated normalized).

4.3. Detect the Pedestrians Which Run the Red Light

After the objects are detected and tracked, we marked every pedestrian, see from **Figure 5**. We set an alarmed area firstly. While the red light is on, if we tracked the pedestrians run into the alarmed area, we can point it out that the pedestrians who are running in the red light and storing the pictures of the pedestrians running when there is a red light. Meanwhile we can give some good suggestions to the pedestrians, for educating them to be careful and don't break the red lights.

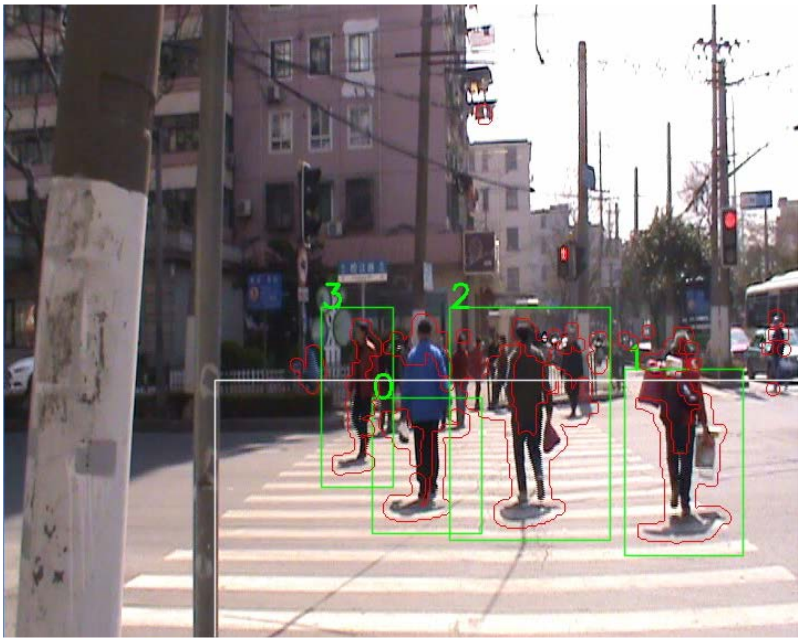

Figure 5. Four pedestrians marked out for walking into the alarmed area.

We have done our experiments in school playground and traffic junctions. The accuracy rate of detecting and tracking pedestrians in school playground can reach above 80%. Then we use it in the traffic junctions, through 6 video sequences which took from 3 traffic junctions, for every traffic junctions we took two 15 minutes surveillance video sequences. The accuracy rate of detecting pedestrians who run the red light is above 60%.

5. Conclusion

We apply the pedestrian recognition of video sequence to the road safety, through detecting the moving pedestrians in it and tracking them properly. While the red light is on, if we detect pedestrians running into the alarmed area, we alarm this information to the drivers and the pedestrians to improve the safety to road traffic. This system works well to some extend to the real time traffic that we have analyzed, so it is possible to apply the pedestrian recognition of video sequence to improve road safety. In the future we will focus on how to improve the accuracy of detecting and tracking. And we are planning to add the prediction module in our system, the main idea is that before the people run the red light, we give some conditions to judge whether the pedestrian will run the red light or give the probability of the pedestrians running the red light.

Acknowledgements

This paper was partially supported by The National Natural Science Foundation (61374040), Key Discipline of Shanghai (S30501), Scientific Innovation program (13ZZ115), Graduate Innovation program of Shanghai (54-13-302-102).

References

[1] Stauffer, C. and Grimson, W.E.L. (1999) Adaptive Background Mixture Models for Real-Time Tracking. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Fort Collins, 23-25 June 1999.

[2] Wohler, C., Kressel, U. and Anlaur, J.K. (2000) Pedestrian Recognition by Classification of Image Sequences Global Approaches vs. Local Spatio-Temporal Processing. 15*th International Conference on Pattern Recognition*, Barcelona, 3-7 September 2000, 540-544.

[3] Gavrila, D.M. (2000) Pedestrian Detection from a Moving Vehicle. In: Vernon, D., Ed., *Computer Vision—ECCV* 2000, Springer, Berlin, 37-49. http://dx.doi.org/10.1007/3-540-45053-X_3

[4] Dalal, N. and Triggs, B. (2005) Histograms of Oriented Gradients for Human Detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, 25-25 June 2005, 886-893.

[5] Chang, F.L., Ma, L. and Qiao, Y.Z. (2007) Human Oriented Multi-Target Tracking Algorithm in Video Sequence. *Control and Decision*, **22**, 418-422.

[6] Liu, G.C. and Wang, Y.J. (2009) An Algorithm of Muli-Target Tracking Based on Improved Particle Filter. *Control and Decision*, **22**, 317-320.

[7] Takala, V. and Pietikainen, M. (2007) Multi-Object Tracking Using Color, Texture and Motion. *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, 17-22 June 2007, 1-7.

[8] Yang, T., Pan, Q. and Li, J. (2005) Real-Time Multiple Objects Tracking with Occlusion Handling in Dynamic Scenes. *IEEE Conference on Computer Vision and Pattern Recognition*, 20-25 June 2005, 970-975.