Research on Moving Target Recognition for Vehicle Driving Robot with Remote Operation Based on OpenCV

Junxian Li, Zhen Zhang, Yu Meng
Jilin University, No. 5988 Renmin Street, Changchun, Jilin, China, Changchun, Jilin, China
Jilin University, No. 5988 Renmin Street, Changchun, Jilin, China, Changchun, Jilin, China
Jilin University, No. 5988 Renmin Street, Changchun, Jilin, China, Changchun, Jilin, China
18843109017, islijunxian@gmail.com
17843124839, zhangzhen2016@mails.edu.jlu.cn
18843110351, 2388736956@qq.com

Abstract: Owing to the need of vehicle driving with remote operation and driverless technology, it has become a hot topic to deal with the real environment returned by the camera placed in the vehicle which serves as a sensor. In order to realize the recognition of moving objects, such as motor vehicles in the returned data stream, the C++ interface of OpenCV can be employed to program and detect the dynamic target by using GMM algorithm, achieving the effect of real-time detection of the environment picture returned by the camera and marking moving targets such as vehicles.

1. Introduction
In terms of remote operation vehicles, if a moving object has been marked on the real-time environment screen returned by the controlled vehicle's driving side, it will play a favorable assistant role for the driver at the actual operation end to judge the driving environment, which involves dynamic target detection technology. Classical dynamic object detection methods include Temporal Difference, Optical Flow, and Background Subtraction. Temporal Difference extracts the moving object in the image by thresholding the pixel-based time difference between adjacent frames, which is insensitive to light, and has good self-adaptability. However, it is not possible to effectively detect objects that have a high speed of motion or a large internal color; Optical Flow performs the motion detection through the characteristics of the optical flow of the moving object over time. The detection effect is good but the calculation is large and the real-time performance is poor, so the background difference method is eventually adopted. The video module of OpenCV contains several commonly used background subtraction methods, in which the Gaussian of Mixture Models (GMM) method works well.
2. Principle and Realization of GMM

2.1 Principle of GMM

GMM is an object model based on a Gaussian probability density function, namely, a normal distribution probability density function. Through data statistics and sample analysis, to find the appropriate parameters (mean $\mu$ and standard deviation $\sigma$) to model things, you can have more accurate and in-depth description and understanding of the nature of things. The probability density function given by the GMM is a weighted sum of several Gaussian probability density functions:

$$p(x|\theta) = \sum_{k=1}^{K} A_k \phi(x|\theta_k)$$

where $K$ —— Total number of models

$A_k$ —— Non-negative weight coefficient, $\sum_{k=1}^{K} A_k = 1$

$\phi(x|\theta_k)$ —— Normal distribution probability density

Hence the basic distribution mixed by the GMM is still Gaussian distribution. Similarly, through data statistics and sample analysis, and an appropriate algorithm, the parameters of the K sub-models, $\mu_k$ and $\sigma_k$, can be found, while a Gaussian subdivision model represents an independent class. Each data in the sample is in each Gaussian. Subprojection on the model gives the probability that the sample is subordinate to each sub-model. At this time, the class with the highest probability of selection is used as the result of determining the subordination, and the attribution response of a single sample can be judged according to the selected parameter.

In terms of computer vision, each frame in the video data stream is divided into two parts: foreground and background. The background is the scene that occupies a relatively high position in the picture and the update rate is slow, and the foreground is a dynamic object whose position update rate is faster in the picture relative to the background. The mixed Gaussian background modeling is based on pixel (gray value) sample statistics and can rely on the background representation method to describe the law of each pixel (gray value) in the time domain with the application of a Gaussian distribution.

The mixed Gaussian distribution probability density function for a single sampling point $x_t$ is:

$$p(x_t) = \sum_{i=1}^{K} \omega_{i,t} \times \eta(x_t|\mu_{i,t},\sigma_{i,t})$$

where $K$ —— Total number of distribution patterns

$\omega_{i,t}$ —— The weight of the i-th Gaussian distribution at time t, namely, the probability of projection on the i-th Gaussian distribution

$\eta(x_t|\mu_{i,t},\sigma_{i,t})$ —— The i-th Gaussian distribution at time t, $\mu_{i,t}$ is its mean, and $\sigma_{i,t}$ is its standard deviation.

Under the premise that the pixels are mutually independent, the sample observation values of the pixel gray are analyzed, and dynamic adaptive modeling is performed to find the appropriate parameters of the sub-model (mean value and standard deviation), and the Gaussian construction is performed by mixing the pixel gray values. The model determines the attribution model of the pixel, and the pixel that does not belong to any one of the sub-models is determined to be the foreground pixel.

2.2 GMM algorithm

The core of GMM is to address the parameters $\mu_k$ and $\sigma_k$ of each sub-model. If the maximum
likelihood method is used for parameter estimation, the calculation process would be very complicated and it will be difficult to obtain an optimal solution. Therefore, dynamic adaptive modeling of EM algorithm can be used. The two parameters of variance and mean in the model are updated in real time. The EM algorithm idea is explicated as follows:

K (3 to 5) Gaussian distributions are used to add up each pixel in the image in the nearest t-frame gray value \{X_1, X_2, X_3, ..., X_t\}. The prior estimation parameters and weights of the K models are given during initialization. The parameters and weights of each Gaussian model are iterated from the previous step. This step is given as each step has been given, called step E. After obtaining a new frame of image, its sample observations \{x_1, x_2, x_3, ..., x_t\} are got. According to the 3σ principle commonly used in the project (2.5σ is more reliable here) and the given mixture Gaussian model is matched. If it is successful, the pixel is determined as the background point, otherwise it is the foreground spot. After the determination, the parameters are updated according to the new sample observation value, ready to receive new data. This step optimizes the E-step parameter, which is called M-step. The above two steps are repeated until the algorithm converges so that the parameters reach a local optimum. The specific implementation of the algorithm is explicated as follows:

1. Each new sample value is matched with the current K models according to Equation 2.2.1. If the difference between the mean values of any one of the models satisfies the following Equations, the matching will be considered to be successful, that is, the pixel is the background point, otherwise it is the foreground points.

\[
|X_{t+1} - \mu_{i,t}| \leq 2.5\sigma_{i,t} \quad (2.2.1)
\]

2. If the match is successful, the finished matched model parameters are updated according to 2.2.2, 2.2.3 and 2.2.4.

\[
\rho = \alpha \cdot \eta(X_{t+1}|\mu_k^{(t)}, \sigma_k^{(t)}) \quad (2.2.2) \quad \mu_{k,t+1} = (1-\rho)\mu_{k,t} + \rho X_{t+1} \quad (2.2.3)
\]

\[
\sigma_{k,t+1}^2 = (1-\rho)\sigma_{k,t}^2 + \rho(X_{t+1} - \mu_{k,t+1})^T(X_{t+1} - \mu_{k,t+1}) \quad (2.2.4)
\]

Equation 2.2.2 is called "learning rate" and the value is between 0 and 1.

3. If the matching is unsuccessful, the sub-model parameters remain unchanged, and the weights of all K sub-models are updated according to Equation 2.2.5.

\[
\omega_{k,t+1} = (1-\alpha) \times \omega_{k,t} + \alpha \times M_{k,t+1} \quad (2.2.5)
\]

The sub-model matching the current pixel \(M_{k,t}=1\), the remaining K-1 sub-model \(M_{k,t}=0\).

3. Vehicle detection algorithm

The vehicle is a dynamic target in the driving process. Therefore, a Gaussian model can be used to perform GMM on the image data returned by the camera to determine the dynamic target. In the experiment, due to the limited conditions of the prototype and the danger of the experimental environment, it is possible to process the real-time screen data returned by the camera instead of the video of a dynamic vehicle.

The specific algorithm implementation flow chart is shown in Figure 3.1.
4. Realization of vehicle recognition algorithm based on OpenCV

Two versions of the Gaussian Mixture-based Background/Foreground Segmentation Algorithm are implemented in the OpenCV computer vision library. The call interface is clear and the results are very good.

Taking BackgroundSubtractorMOG2 as an example, a prototype constructor for the class BackgroundContractorMOG2 with a formal parameter is:

```
BackgroundSubtractorMOG2::BackgroundSubtractorMOG2(int history, float varThreshold, bool bShadowDetection=true)
```

Where history is the number of used historical frames, varThreshold is a threshold for determining whether it is a background (this value does not affect the background update rate), and bShadowDetection indicates whether shadow detection is used (enabled by default).

Its user call interface is an overloaded operator:

```
void BackgroundSubtractorMOG2::operator()(InputArray image, OutputArray fgmask, double learningRate=-1)
```

Where image is the current frame image, fgmask is the output foreground mask, and learningRate
is the background learning rate.

Auto recognition can be easily implemented using the hybrid Gaussian model interface provided by OpenCV.

5. Results demonstration
By processing the image data returned by the camera with the application of GMM, the dynamic objects in the scene will be marked. Take a section of road traffic video as an example, Figure 5.1 and Figure 5.2 show the car images before and after processing, respectively.

![Figure 5.1. Vehicle image before processing.](image1)

![Figure 5.2. Vehicle image after processing.](image2)

Because of the requirement of real-time performance, the accuracy of recognition will inevitably decrease in applications where the processing speed is required to be high. And due to factors such as the judgment threshold, the recognition success rate cannot be 100%.

When the learning rate is 0.2 and the threshold varThreshold is 20, the detection results are collected. The results are shown in Table 5.1.

| Detection time /s | Passed targets/vehicle | Marked number/vehicle | Detection time /s |
|-------------------|------------------------|-----------------------|-------------------|
| 1                 | 3                      | 1                     | 33.3%             |
| 2                 | 4                      | 3                     | 75%               |
| 3                 | 4                      | 3                     | 75%               |
| 4                 | 5                      | 4                     | 80%               |
| 5                 | 3                      | 3                     | 100%              |
| 6                 | 7                      | 7                     | 100%              |
| 7                 | 7                      | 6                     | 85.7%             |
| 8                 | 5                      | 5                     | 100%              |
| 9                 | 5                      | 5                     | 100%              |
| 10                | 4                      | 4                     | 100%              |

6. Conclusion
Through calling the GMM interface of OpenCV, real-time detection of dynamic targets in the video data stream can be achieved. After marked, the dynamic objects including pedestrians and vehicles
will become conspicuous, which actually helps to remind the staff at the remote operation end of the remote operation vehicle driving robot to reduce the accident rate and improve driving safety coefficient.

References
[1] Li Bo. Research on several key technologies of vehicle identification and tracking based on image analysis[D]. Wuhan: Huazhong University of Science and Technology, 2011:9-10.
[2] Tian Penghui. Research on the detection and tracking of moving objects in video images[D]. Xi'an: Chang'an University, 2013:16-18.
[3] KaewTraKulPong, Pakorn, and Richard Bowden. "An improved adaptive background mixture model for real-time tracking with shadow detection."[J] Video-Based Surveillance Systems. Springer US, 2002. 135-144.
[4] Stauffer C, Grimson W. Adaptive background mixture models for real-time tracking[C]. Ft Collins, CO, USA: Proceeding of Computer Vision and Pattern Recognition, 1999: 246-252.
[5] Krupp A, Mueller W. Classification trees for random tests and functional coverage[C]. Proc of the Conference on Design, Automation and Test in Europe, 2006: 1031-1032.
[6] Alekseev S, Tiede R, Tollkuhn P. Systematic approach for using the classification tree method for testing complex software systems[C]. Proceeding of the IASTED Conference on Software Engineering, 2007: 13-15.
[7] Conrad M, Krup A. An extension of the classification-tree method for embedded systems for the description of events[J]. Electronic Notes Theory Computer Science, 2006, 164(4): 3-11.
[8] LI En-ke, LIU Shang-qian, MA Yan-xuan, et al. Novel algorithm for adaptive image segmentation of moving targets[J]. Journal of Xidian University, 2008, 35(3): 491-494.
[9] DING Zhong-xiao. Survey on moving object detection methods for video surveillance images[J]. TV technology, 2008, 32(5): 72-76.
[10] Veenman C, Reinders M, et al. Resolving motion correspondence for densely moving points. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001, 23(1): 54-72.
[11] Yilmaz A, et al. Contour based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(11): 1531-1536.
[12] Collins R, Gross R, Silhouette-based Human Identification from Body shape and Gait. 5th International Conference on Automatic Face and Gesture Recognition, 2002.