Low-Complexity Moving Object Detection Algorithm in Dynamic Background

Yifan Zheng
Information Science and Engineering School, Shandong University,
Qingdao, China
serephz@163.com

Yujun Li
Information Science and Engineering School, Shandong University,
Qingdao, China
liyujun@sdu.edu.cn

Laibo Zheng
Information Science and Engineering School, Shandong University,
Qingdao, China
zhenglaibo@sdu.edu.cn

***Abstract*—The scale of the monitoring system is becoming larger and larger. In order to perform intelligent video processing in surveillance systems, we need to detect moving object in image sequence. Some methods in the literature can achieve a valid detection result, but usually they have high computational complexity. In the outdoor scenes, the background is usually dynamic, and the dynamic background makes it difficult to detect moving object. In order to solve these problems, we propose a new method with low computational complexity using mass center coordinate to expand the mask image. The proposed method can remove the interference of dynamic background in the detection. Experiment results show that our method can mask the dynamic background more completely while ensuring fast computation and consuming less hardware resources. The method can be used in massive video intelligent processing.**

Keywords-moving object detection; dynamic background; frame difference method; mask extension

I. INTRODUCTION

Moving object detection refers to the detection of objects that produce spatial movement in a continuous image of a video. It is an important procedure in video processing and can be used in a number of industrial applications such as surveillance systems and intelligent transportation [1]. Currently, there are two main approaches to detect moving object, traditional image processing methods [2] and deep learning methods.

There are three main methods used in traditional image processing for moving object detection: frame difference [3-4], optical flow [5], background subtraction [6].

The frame difference method is one of the more widely techniques; it has a simple calculation principle and a relatively stable outcome, making it useful in a variety of real-world situations. The frame difference method applies a difference calculation between two adjacent video frames, setting a threshold for the difference image. The values above the threshold set to 1 and below the threshold set to 0. The area with a value of 1 is considered to be the presence of a moving object. And the remaining part with a value of 0 is considered to be the background part of the image. However, the disadvantages of the frame difference method are obvious, it is not possible to effectively remove background wobbles and different types of noise. So, the frame difference detection method relies heavily on expert experience and is often used as a supplementary mean in other detection methods [7-8].

Optical flow is a pixel motion vector between adjacent images in an image sequence. The key step of detecting moving object in image by optical flow method is to analyze the motion vector of each pixel. Then we can distinguish the background and moving target from the motion vector of corresponding pixels. Create a vector matrix of the same size as the image to store motion vector of each pixel in two consecutive images. Each motion vector of the matrix is analyzed to determine if there is a moving object.

The background subtraction method uses a static background frame and the detection frame to subtract for moving object. The two algorithms most commonly used are GMM (Gaussian mixture model) algorithm [9-11] and Vibe (visual background extractor) algorithm [12]. The GMM algorithm is characterized by the distribution of background pixel values in the time domain showing a Gaussian distribution. Pixels within a certain threshold are judged as background, and those that do not conform to this distribution are judged as foreground, that is moving object. The GMM algorithm requires a large number of training frames to update the parameters, making it fitted with proper static background frame. Thus, GMM algorithm possesses a higher complexity.

The Vibe algorithm uses the first frame image as the background for initialization. For each pixel point in the video image within the joint neighborhood randomly selected values are used as background sample values. Vibe is a parameter-free model, so its computing speed is relatively fast. Since the Vibe algorithm uses only a single frame for the initialization of the background model, the selection of the first frame image is more demanding. The Vibe algorithm cannot be effectively initialized in the presence of dynamic background noise or moving objects in the initial frame image, and this will introduce the ghost area and cause the error detection of the moving object. There is a lot of subsequent work has been done to upgrade the algorithm and further research for this problem [13-16].

The deep learning methods [17-19] focus on using neural networks to extract various objects features in images. Then output the moving object and target attributes in the image by processing those feature data. However, compared with traditional image processing methods, the computational complexity of the neural network model is much higher. Applying it to a large-scale monitoring system requires extensive hardware requirements. Therefore, this paper will not go into details in this aspect.

In large campus surveillance systems, most outdoor scenes of monitors are with dynamic backgrounds, such as waving leaves, greenery, and swinging flags. These objects are not our targets of interest. Person, car and other targets are the objects we need to detect. The number of cameras of the surveillance system on campus is very large. If the full surveillance video is to be processed in real time, the detection of moving objects of interest has to be achieved using as few computational resources as possible. Therefore, it is necessary to reduce the computational complexity of the detection algorithm while ensuring a certain accuracy rate.

Based on the above characteristics and the problems of detection methods mentioned above in outdoor scenes with dynamic backgrounds. We propose a dynamic background removal method using a center-of-mass extension mask. This method uses multi-frame differential images to calculate the initial mask area, and then uses a center-of-mass expansion algorithm to expand the mask area. Then mask out the dynamic background and calculate the moving area of the differential image. Using this method to determine whether there is a moving target of interest in the image.

This paper is structured as follows. Section II gives the procedure of the proposed method. Section III shows the results of experiment, and Section IV concludes the whole paper.

II. BACKGROUND MASK GENERATION METHOD

The frame difference method is the simplest one to detect the moving object. But it can not satisfy the requirement of detection, for it unable to output an accurate result. Others methods have high computational complexity, and can not used in a large intelligent surveillance system. To keep the computational speed advantage of frame difference method, we use the difference in pixel counting characteristics between dynamic backgrounds and moving objects in the images.

A. Mask image initialization

First, we get the surveillance video data, then convert each frame from RGB format to grayscale. The value of each pixel could calculate by

$$Gray = 0.299 * G + 0.584 * R + 0.114 * B. \tag{1}$$

Each frame size changed from $3 \times H \times W$ to $1 \times H \times W$. This can reduce the amount of calculation in the subsequent differential operations.

Differential images can highlight the edge of moving objects. Dynamic backgrounds are usually moving in the image, and position of them in the image is fixed. We use this property to extract moving objects from dynamic backgrounds. Then we calculate the differential image of 200 consecutive grayscale frames, and binarize each image. $D_n(x, y)$ is given as

$$D_n(x,y) = \begin{cases} 0, & |G_n(x,y) - G_{n+1}(x,y)| < T_D \\ 1, & |G_n(x,y) - G_{n+1}(x,y)| \geq T_D \end{cases}, \tag{2}$$

where T_D is the threshold value for the binarization of differential image, which can filter out the pixels with a large difference between two grayscale images. These pixels correspond to the parts of the image that are in constant motion, i.e., the dynamic background. Then set these pixels to 1, and get 200 two-dimensional arrays of the same size as the image to get initial mask image $M(x, y)$ by using

$$M(x,y) = \begin{cases} 0, & \sum_{n=1}^{200} D_n(x,y) < T_M \\ 1, & \sum_{n=1}^{200} D_n(x,y) \geq T_M \end{cases}, \tag{3}$$

where T_M is the threshold value for binarization of the mask image. The pixels in mask image $M(x, y)$ with data of 1 are the dynamic background to be masked.

B. Mask extension using mass center

Differential image $D_n(x, y)$ can sensitively perceive changes in the edges of single connected regions. To achieve effective masking of dynamic backgrounds, we need extend mask image $M(x, y)$ to cover the middle part of one of the discrete mask regions. Thus, we propose a method using mass center to extend mask area. Then we calculate the mass center in the mask $M(x, y)$. The calculation of the mass center coordinates is given as

$$x_m = \frac{\sum m_i x_i}{\sum m_i},\ y_m = \frac{\sum m_i y_i}{\sum m_i}. \tag{4}$$

For all value of the pixels in mask are 1, we can use the followed formula to calculate the coordinates of the mass center (x_m, y_m),

$$x_m = \frac{\sum_{i=0}^{W}\left(\sum_{j=0}^{H} M(x_i, j) * x_i\right)}{\sum_{i=0}^{W}\left(\sum_{j=0}^{H} M(x_i, j)\right)}, \tag{5}$$

$$y_m = \frac{\sum_{i=0}^{H}\left(\sum_{j=0}^{W} M(y_i, j) * y_i\right)}{\sum_{i=0}^{W}\left(\sum_{j=0}^{H} M(y_i, j)\right)}. \tag{6}$$

Before extending the mask area, we use open operation in morphology to change initial mask image, and then get image $M'(x, y)$. The open operation includes two steps which are erosion and dilation. The formulas are given as

$$erode(M(x,y)) = \min_{(x',y'):element(x',y')\neq 0} src(x+x', y+y'), \tag{7}$$

$$dilate(M(x,y)) = \max_{(x',y'):element(x',y')\neq 0} src(x+x', y+y'), \tag{8}$$

where $src(x + x', y + y')$ is the convolution operation with the image using a 7×7 two-dimensional array $B(x', y')$. Select the center of the array as the anchor point. The operation of dilation is setting the pixel value of anchor point to the maximum value that $B(x', y')$ covers in $M(x, y)$. Erosion is setting to the minimum value.

The open operation can eliminate isolated small points outside the target area in the image, and has a valid effect on removing Gaussian noise. Open operation can smooth the boundaries of larger objects without significantly changing their area.

When observing the initial mask image $M(x,y)$ we find that the shape of the mask is not a complete single-connected area, because differential image always highlights the edge in the origin image. Monitor system produce a variety of noise while transmitting the data due to many objective factors. These noises can cause dramatic pixel change and be captured by differential image $D_n(x,y)$. Discrete mask regions will be generated in the $M(x,y)$ image, and it can be eliminated by open operation effectively.

After the open operation we get the mask image $M'(x,y)$. Then we extend the mask region in each of the two vertical directions towards the mass center. The extension of the mask region to the mass center stop at its coordinate (x_m, y_m). For each column in the image, we set $y_1 = min(y_{min}, y_m)$ and $y_2 = max(y_{max}, y_m)$, and for each row, we set $x_1 = min(x_{min}, x_m)$ and $x_2 = max(x_{max}, x_m)$. The process of extension is

$$M_h'(x,n) = \{1 | x \in (x_1, x_2),\ x_1 \neq x_2\}, \tag{9}$$

$$M_v'(n,y) = \{1 | y \in (y_1, y_2),\ y_1 \neq y_2\}, \tag{10}$$

where $M_h'(x,n)$ is the expansion process in the horizontal direction. On each vertical array, all values in the (x_1, x_2) interval are set to 1. x_1 is the value of the smallest horizontal coordinate among all data with value 1 in $M_h'(x,n)$. x_2 is the value of the biggest horizontal coordinate among all data with value 1 in $M_h'(x,n)$. When $x_1 = x_2$ indicates that all data on this vertical array are not inside the mask region. Similarly, the parameters in the vertical direction formula are the same. After the extension we get the final mask image $M_e'(x,y)$.

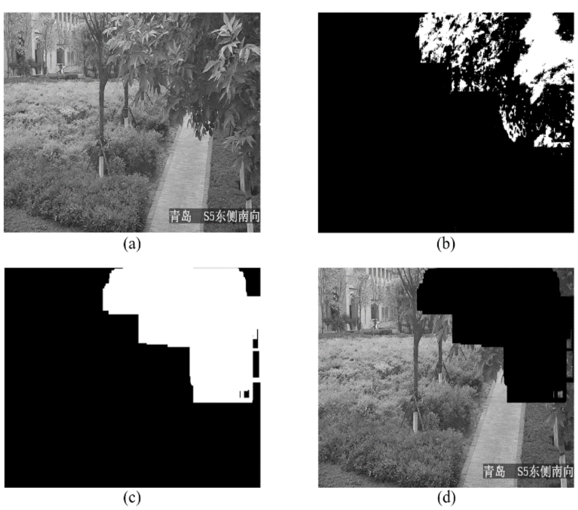

Fig 1. Example of the propose method: (a) input frame, (b)mask without expansion, (c)mask after expansion, (d)frame cover with final mask

C. *Moving object detection*

We use 200 images in the initial sequence to calculate the mask image. After masking the dynamic background in the image, we use noise reduction for the rest part of the image, and then count the number of moving pixels with differential image.

The percentage of moving pixels in the remaining part $R(n)$ is used to determine whether there is a moving object in the image,

$$R(n) = \frac{\sum\sum\left(D_n(x,y) - M_e'(x,y)\right)}{H*W - \sum\sum M_e'(x,y)}. \tag{11}$$

where $\sum\sum(D_n(x,y) - M_e'(x,y))$ is the number of moving pixels in differential image after masking with final mask $M_e'(x,y)$. When the percentage of $R(n)$ is higher than 15.4%, it indicates that there is a moving object in the image.

When detecting moving objects in the subsequent images, we calculate the new mask image at the same time, which enables continuous updating of the mask image while camera is working.

III. Experimental Results

In order to evaluate the function of proposed method, we implement algorithm on an Intel i5-9500T 2.2GHz CPU and 8GB memory. Microsoft Visual Studio 2008 and C++ are used.

The problem solved in this paper is the interference of dynamic background in the outdoor scene image. The final output is the presence or absence of moving objects in the image. GMM, Vibe, and Optical Flow methods are compared with the proposed one. Their final output is a binarized image with the pixels corresponding to the moving objects marked. To compare properly, we uniform the output of all methods by calculating $R(n)$ to determine whether there is a moving object in the current image.

We uses our own surveillance video of outdoor scenes as the dataset. Due to the huge size of the surveillance data, we select 40 typical videos of 5 minutes in length, and mark the time periods with and without moving objects. Each of them has dynamic background interference in the image. The image size of videos is 576×720. The detection results of all methods in some videos are shown in Figure 2. From Figure 2 we find that GMM and Vibe cannot remove the interference of dynamic background. The optical flow method could suppress part of interference. But in most cases, dynamic background still makes the detection results incorrect. There is a part of the waving leaves at the top right corner of image in the first video. As it shown in Figure 1, the generated final mask using our methods can cover the most part of dynamic background. Similarly, the mask image also works in the second and third video. The average time for each algorithm to process single frame is shown in Table I.

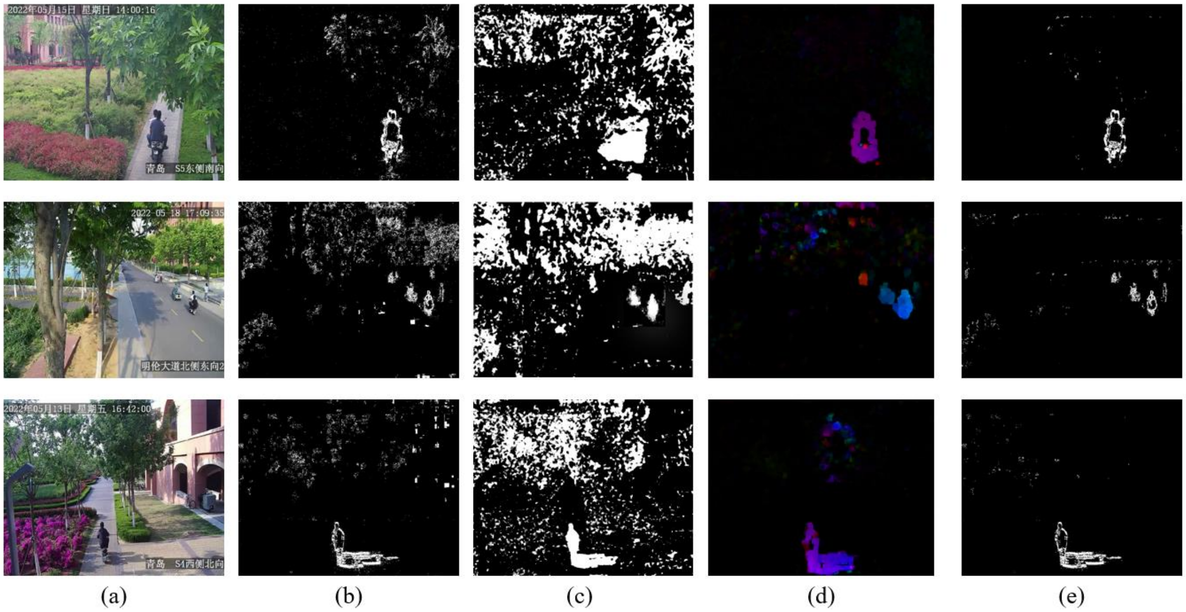

Fig 2. Binarization image of each method: (a) input frame, (b)GMM method, (c)Vibe method, (d)Optical Flow method, (e)Our methods

Table I. Average Processing Time

Methods	GMM	Vibe	Optical Flow	Proposed
Average Processing Time (ms)	36.4	41.4	62.9	2.3

Experimental results show that the proposed method in this paper can effectively suppress the effect of dynamic background on the moving object detection, and maintain the advantage of frame difference method in terms of computational speed at the same time.

IV. CONCLUSION

In this paper, we propose a moving object detection method based on dynamic background removal. We use a sequence of images to generate the mask image and then extend the mask area with mass center coordinates of the original mask region. This method exploits the statistical properties of the pixel values of dynamic background differential images. It reduces the interference of dynamic backgrounds on results in outdoor scenes, while its complexity is very low. By using this method, we can detect moving objects in the image correctly. However, in our monitor system testing, we found that there are very few extreme scenarios where results of proposed method are not ideal. This is our future work.

REFERENCES

[1] Zhao X, Wang G, He Z, et al. A survey of moving object detection methods: A practical perspective[J]. Neurocomputing, 2022, 503: 28-48.

[2] Li H, Wang Y, Liu W. Moving object detection based on HFT and dynamic fusion[C]//2014 12th International Conference on Signal Processing (ICSP). IEEE, 2014: 895-899..

[3] Sengar S S, Mukhopadhyay S. A novel method for moving object detection based on block based frame differencing[C]//2016 3rd International Conference on Recent Advances in Information Technology (RAIT). IEEE, 2016: 467-472.

[4] Shi G, Suo J, Liu C, et al. Moving target detection algorithm in image sequences based on edge detection and frame difference[C]//2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC). IEEE, 2017: 740-744.

[5] Agarwal A, Gupta S, Singh D K. Review of optical flow technique for moving object detection[C]//2016 2nd international conference on contemporary computing and informatics (IC3I). IEEE, 2016: 409-413.

[6] Kalsotra R, Arora S. Background subtraction for moving object detection: explorations of recent developments and challenges[J]. The Visual Computer, 2021: 1-28.

[7] Xu Z, Zhang D, Du L. Moving object detection based on improved three frame difference and background subtraction[C]//2017 International Conference on Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII). IEEE, 2017: 79-82.

[8] Yin J, Liu L, Li H, et al. The infrared moving object detection and security detection related algorithms based on W4 and frame difference[J]. Infrared Physics & Technology, 2016, 77: 302-315.

[9] Li Y H, Xiong C Z, Yin Y X, et al. Moving object detection based on edged mixture Gaussian models[C]//2009 International Workshop on Intelligent Systems and Applications. IEEE, 2009: 1-5.

[10] Bian Z, Dong X. Moving object detection based on improved Gaussian mixture model[C]//2012 5th International Congress on Image and Signal Processing. IEEE, 2012: 109-112.

[11] Ali S T, Goyal K, Singhai J. Moving object detection using self adaptive Gaussian Mixture Model for real time applications[C]//2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE). IEEE, 2017: 153-156.

[12] Barnich O, Van Droogenbroeck M. ViBe: A universal background subtraction algorithm for video sequences[J]. IEEE Transactions on Image processing, 2010, 20(6): 1709-1724.

[13] Yan Q, Wang J. An Improved Moving Target Detection Algorithm Based on Vibe[C]//2020 International Conference on Computer Network, Electronic and Automation (ICCNEA). IEEE, 2020: 16-20.

[14] Zhang Y, Zhao X, Tan M. Motion detection based on improved Sobel and ViBe algorithm[C]//2016 35th Chinese Control Conference (CCC). IEEE, 2016: 4143-4148.

[15] Gao J, Zhu H. Moving object detection for video surveillance based on improved ViBe[C]//2016 Chinese Control and Decision Conference (CCDC). IEEE, 2016: 6259-6263.

[16] Qiu S, Tang Y, Du Y, et al. The infrared moving target extraction and fast video reconstruction algorithm[J]. Infrared Physics & Technology, 2019, 97: 85-92.

[17] Qu Y, Jiang L, Guo X. Moving vehicle detection with convolutional networks in UAV videos[C]//2016 2nd International Conference on Control, Automation and Robotics (ICCAR). IEEE, 2016: 225-229.

[18] Shahbaz A, Jo K H. Moving object detection based on deep atrous spatial features for moving camera[C]//2020 IEEE 29th International Symposium on Industrial Electronics (ISIE). IEEE, 2020: 67-70.

[19] Zhu H, Yan X, Tang H, et al. Moving object detection with deep CNNs[J]. Ieee Access, 2020, 8: 29729-29741.