Detecting moving objects on a video having a dynamic background

FX Arinto Setyawan, Joo Kooi Tan, Hyoungseop Kim, Seiji Ishikawa
Dept. of Control Engineering, Kyushu Institute of Technology,
1-1 Sensui-cho, Tobata-ku, Kitakyushu-shi, Fukuoka, 804-8550, Japan
E-mail: {arinto,etheltan,ishikawa}@ss10.cntl.kyutech.ac.jp, kim@cntl.kyutech.ac.jp
tel: 093-884-3191

Abstract

This paper proposes a method of detecting moving objects in a video having a dynamic background using a method which infers the background sequentially. The proposed method performs the update of the pixel values in the background which are influenced by the value of the current pixel. The aim is to cope with changes in the value of the pixels in the background caused by the movement of the background objects such as the leaves swaying on trees, the water droplets of the rain or the change in light intensity according to the time lapse. The performance of the proposed method is shown experimentally using the video taken on a rainy and windy day.

Keywords: Object detection, foreground detection, background inference, Gaussian distribution.

1. Introduction

The increasing number of crimes and accidents that occur nowadays requires a reliable video surveillance system. The surveillance system can be realized by installing cameras in places vulnerable to crime and accidents. There are 3 types of video surveillance activities, i.e. manual (conventional), semi-autonomous and fully autonomous [1].

The conventional video surveillance systems can record what they see, but cannot find what is seen. In conventional surveillance systems, the task of video surveillance review was performed by trained security personnel. The increase of surveillance video data makes security officer jobs increasingly heavy. Solution by adding more security personnel is an option that spends much cost. A better solution is to replace a conventional video surveillance system by a fully autonomous system.

On the fully-autonomous system, the input is a video sequence taken in the spot where surveillance is done, and, without human intervention, object detection and object tracking are done using computer [2]. The rapid development of computer, especially in terms of processing speed and a large amount of memory, allows the implementation of a fully autonomous surveillance system.

Detection of moving objects is one of the important tasks in many computer vision applications including a video surveillance system. A moving object detection system is a system that detects moving objects in a video taken with the use of surveillance cameras. A general approach used in the detection of moving objects is background subtraction [3,4,5]. The idea of the background subtraction is to compare the image scene at the current time with a reference of the background image model. Usually a reference of the background image model is the first image frame of the video and is updated all the time.

The other application of the detection of moving objects is for a traffic control system, application in the field of sports, hospital patient monitoring, monitoring of students in a class, and others. Aslani and Nasab [6] performed a research on the application of detecting moving objects for a traffic monitoring system. Application on detecting moving objects in the field of sports was done by Manikandan and Ramakrishnan [5].

Much research on the detection of moving objects using a static camera has been done using the methods other than the background subtraction. Stauffer and Grimson [7] use Gausian Mixture Model to address changes in the background such as changes in light intensity, slow-moving objects and the effects of moving elements in a scene. Aslani and Nasab [6] employ optical flow to detect and track a moving object. Keerthana, Ravichandran and Santhi [8] use Fuzzy-Extreme Learning Machine for detecting a moving object. Zhou, Yang and Yu [9] detect a moving object by a method of Detecting Contiguous Outliers in the Low-Rank Representation (DECOLOR).

In this paper, we propose a method of detecting a moving object in a video having a dynamic background. Pixel values on the background at all times are updated to get the model background. The updated value depends on how often the pixels are recognized as background or foreground. After moving object is detected, morphological operations are performed on the image frame in order to get better detection results. This method was tested on videos taken during a rainy and windy day in order to get a video that has a dynamic background.

2. Method

The flowchart on the method used in detecting moving objects in a video having a dynamic background is shown in **Fig. 1**. The first step of the proposed method is to convert an image at time T (abbr., image T) from RGB to gray level. The purpose of this conversion is to reduce the computational load. Standard formula to convert RGB to grayscale is given as follows;

$$I = 0.2989R + 0.5870G + 0.1140B\,. \tag{1}$$

The second step is to determine the normal distribution model, $N_T(f;\,\mu,\,\sigma)_{(x,y)}$, of a pixel at (x,y) in the image frame. The normal distribution is formed using the following equation;

$$N_T(f;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2}\left(\frac{f-\mu}{\sigma}\right)^2} \tag{2}$$

Here f is the value of the pixel intensity at pixel (x,y), and μ and σ are the mean and the variance of the pixel intensity. The value of the initial mean is the value of the intensity of the pixel in image T(=0) and the value of the initial variance is determined as 1 in the experiment.

The next step is the judgment if each pixel of the current image is in the background or on the foreground. This judgment is done by subtracting intensity of the current image pixel by the mean of previous image pixel. For the judgment, the following equation is employed;

$$\frac{\left|f_{T+1} - \mu_T\right|}{\sigma} \le Th \tag{3}$$

Here $f_{\mathrm{T+1}}$ is the value of the pixel intensity in a current image frame and μ_{T} is the mean value of the background pixel model. Th is a threshold. A pixel is regarded as the pixels in the background, if it satisfies Eq.(3): Otherwise it is regarded as the pixel on the foreground or on a moving object.

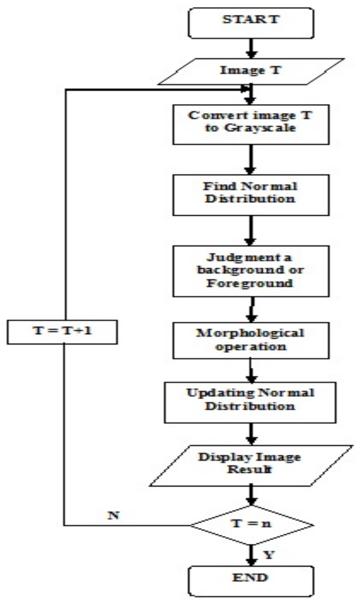


Fig. 1. Flowchart of the system.

After obtaining a set of pixels which represents a moving object, then a morphological operation is performed to those pixels on the image. The morphological operation used in the proposed method is the opening operation. The opening operation is a combination of erosion and dilation operations. They are performed in sequence, i.e., erosion is done to the original image and then dilation is applied to the result. The opening operation of an image *f* by a structuring element *s* is defined using the following equation:

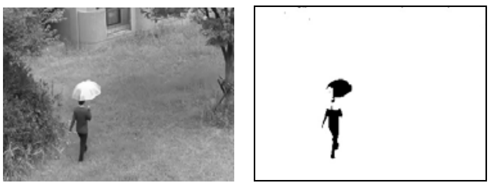
$$g(x,y) = (f(x,y)\Theta s) \oplus s \tag{4}$$

An example of the result of the judgment followed by the morphological operation is shown in **Fig. 2**.

The next step is to update the normal distribution at each pixel in the background. The purpose of this update is to overcome the disturbance that occurs in the background caused by the change in light intensity, swaying leaves of trees, and slow-moving objects, etc. The mean and the variance of the normal distribution are updated using the following equations;

$$\mu_{T+1(x,y)} = \begin{cases} \rho * f_{T+1(x,y)} + (1-\rho) * \mu_{T(x,y)} \\ \quad \text{if } f_{T+1(x,y)} = background \\ (1-\beta) * f_{T+1(x,y)} + \beta * \mu_{T(x,y)} \\ \quad \text{if } f_{T+1(x,y)} = foreground \end{cases} \tag{5}$$

$$\sigma_{T+1(x,y)}{}^2 = \begin{cases} \rho * (f_{T+1(x,y)} - \mu_{T+1(x,y)})^2 + (1-\rho) * \sigma_{T(x,y)}{}^2 \\ \quad \text{if } f_{T+1}(x,y) = background \\ (1-\beta) * (f_{T+1(x,y)} - \mu_{T+1(x,y)})^2 + \beta * \sigma_{T(x,y)}{}^2 \\ \quad \text{if } f_{T+1(x,y)} = foreground \end{cases} \tag{6}$$



(a) (b)

Fig. 2. Example of the performance: (a) Original image, (b) the moving object detection followed by the morphological operation.

$$\rho = c * N(f_{T+1}; \mu_T, \sigma_T{}^2) \tag{7}$$

$$\beta = \frac{1}{1 + k * C_{T+1}{}^2} \tag{8}$$

Here ρ is a variable learning rate. Constant *c* is defined so that the maximum value of ρ is 1. C_{T+1} is the number of successive frames where the pixel $p(x,y)$ has been judged as a foreground pixel, and *k* is a constant. Each pixel is updated with different values depending on how often pixels are judged as foreground or background. These updated values affect the next judgment on the background or the foreground.

3. Experimental Results

For experiment, we use 2 video scenes. The video frame rate and the size of an image are 30 fps and 320 × 240 pixels, respectively. The experimental environment is as follows: The operating system is Windows 7 Enterprise, the processor is Intel® core™ 2 Duo E7500, 4GB RAM, and the used software is MS Visual Studio 2008. The result of moving object detection on video 1 is shown in **Fig. 3**.

The effectiveness of the proposed method is evaluated by comparing the results with the ground truth, as shown in **Fig. 4**. In the resultant image of comparison between the result of the proposed method and the ground truth, the red areas are true positive (TP) that is an overlap part between the ground truth and detection results. Blue means the part which is included in the ground truth but not in the detection result and this part is False Negative (FN); Green means the part which is included in the detection result but not in the Ground Truth and this part is False Positive (FP).

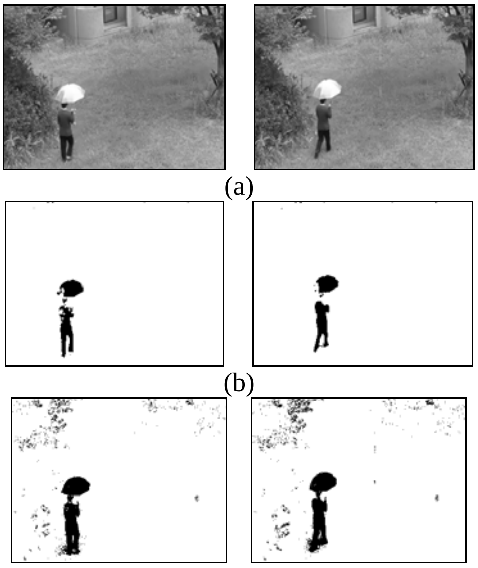
The sensitivity of the proposed method is expressed using the popular parameters, Recall (R) and Precision (P), whereas the accuracy of the method is calculated using the F measure. They are defined by the following formula;

$$R = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{9}$$

$$P = \frac{N_{TP}}{N_{TP} + N_{FP}} \tag{10}$$

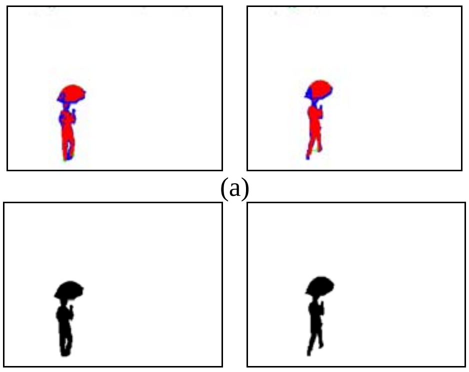
$$F = 2\frac{Recall \times Precision}{Recall + Precision} \tag{11}$$

Here N_{TP} is the number of pixels in the true positive area; and N_{FP} is the number of pixels in the false positive area; N_{FN} is the number of pixels in the false negative area. The result on the evaluation of the proposed method is given in **Table 1.**



(a) (b) (c)

Fig. 3. The result of the moving object detection in video 1: (a) Original frames, frame 130 (the left) and frame 140 (the right), (b) the result of detection by the proposed method, (c) the result of detection using the background subtraction.

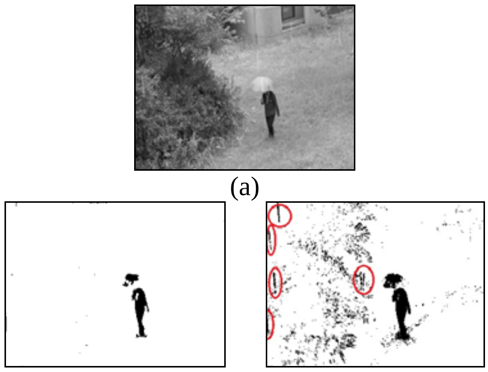


(a) (b)

Fig. 4. Evaluation on the result shown in Fig. 3b: (a) Display of the TP (red), FN (blue) and FP (green), (b) the ground truth images.

Table 1. Evaluation of the method.

Video	Evaluation values		
	R(*Recall*)	*P*(*Precision*)	*F*(accuracy)
Video 1	70.41	97.63	81.76
Video 2	67.84	86.66	75.97



(a) (b) (c)

Fig. 5. Elimination of raindrops: (a) Original image containing raindrops, (b) the result by the proposed method, (c) the result by the background subtraction. The raindrops are marked in red.

4. Discussion and Conclusion

The method used to detect a moving object on a video having a dynamic background is a background sequential inference. This method uses updating of the pixel values of the background based on how often the pixel is recognized as a background or foreground. The updating value of each pixel on a background is different with every pixel on the background. On the other hand, the threshold value, appearing in Eq.(3), to determine whether a pixel is in the background or on the foreground should probably be different with every region on the given image. But it is actually given an identical value such as 1 (equivalent to σ) in the experiment and it gives satisfactory results.

The effectiveness of the proposed method was calculated using the parameters defined by Eqs.(9), (10) and (11). The method is considered effective, if the values of sensitivity, i.e., R and P, are greater than 50%. The used method achieved a high level of the precision

(P) and hence the accuracy (F) is also high. It has achieved the precision level greater than 80% and the accuracy greater than 75%.

Raindrops on a rainy day should be included in the background, since they don't have a particular meaning as objects. The proposed method can reduce raindrops effectively as shown in **Fig. 5**. Many raindrops are observed in the image of **Fig. 5a**. They are eliminated by the proposed method as shown in **Fig. 5b**, but some of them remain as in **Fig. 5c**, if a simple background subtraction is employed. The fields marked in red in **Fig. 5c** are raindrops.

The color of a moving object greatly affects the success of this technique. The color of an object that is very different from the background will be more easily recognized. The speed of a moving object also affects the success of this technique. If the object moves slowly or even it stands still in an image, it will be recognized as part of the background. But it is acceptable, as the topic of this paper is 'moving' object detection**.**

References

[1] I. S. Kim, H. S. Choi, K. M. Yi, J. Y. Choi, S. G. Kong, Intelligent visual surveillance - A survey, *International Journal of Control, Automation, and System*, **8**(5), (2010), pp. 926-936.

[2] K. A. Joshi, D. G. Thakore, A survey on moving object detection and tracking in video surveillance system, *International Journal of Soft Computing and Engineering*, **2**(3), (2012), pp. 44-48.

[3] T. Bouwmans, Recent advance on statistical background modeling for foreground detection: A systematic survey, *RCPS*, **4**(3), (2011), pp. 147-176.

[4] C. Guyon, T. Bouwmans, E. Zahzah, Foreground detection based on low-rank and block-sparse matrix decomposition, *Proc. IEEE International Conference on Image Processing*, (2012), pp. 1225-1228.

[5] R. Manikandan, R. Ramakrishnan, Human object detection and tracking using background subtraction for sports applications, *International Journal of Advanced Research in Computer and Communication Engineering*, **2**(10), (2013), pp. 4077-4080.

[6] S. Aslani, H. M. Nasab, Optical flow based moving object detection and tracking for traffic surveillance, *International Journal of Electrical, Robotics, Electronics and Communications Engineering*, **7**(9), (2013), pp. 774-777.

[7] C. Stauffer and W. E. L. Grimson, Adaptive background mixture models for real time tracking, *Conference on Computer Vision and Pattern Recognition*, Vol. 2, (1995), pp. 246-252.

[8] N. Keerthana, K. S. Ravichandran, B. Santhi, Detecting the moving object in dynamic backgrounds by using fuzzy-extreme learning machine, *International Journal of Engineering and Technology*, **5**(2), (2013), pp. 749-754.

[9] X. Zhou, C. Yang, and W. Yu, Moving object detection by detecting contiguous outliers in the low-rank representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **35**(3), (2013), pp. 597-610.