Proceedings of the 33rd Chinese Control Conference
July 28-30, 2014, Nanjing, China

Moving Object Detection in Dynamic Background

LIU Ting, CHI Hai-hong, HONG Chao，ZHAO Meng-shou

College of Automation, Harbin Engineering University, Harbin 150001, China
E-mail: chi_hon@hrbeu.edu.cn

Abstract: A new method of detecting moving object in dynamic background is proposed in this paper. At first, an adaptive threshold Harris algorithm is proposed in this paper to extract feature points, then, SIFT algorithm is used to describe these extracted feature points. The similarity function is used to match feature points and RANSAC algorithm is used to eliminate the pseudo matches. According to the correct matches, we get the affine transformation matrix which used to compensate the motion of background caused by camera motion, and update the dynamic background with the background model. Finally, the moving object can be detected by background subtraction method. Experimental results show that the method presented in this paper improves the accuracy of feature point extraction and detects moving target in dynamic background accurately.

Key Words: Harris-SIFT Algorithm, Motion Compensation, Background Modeling, Moving Object Detection

1 Introduction

Moving object detection has been the very important field in computer vision. At present, the main methods of moving object detection in dynamic background are optical flow and motion compensation. In addition, there are methods like motion segmentation proposed in [1][2] , movement region integration, and another type of method based on the difference between the feature points of the moving object and the ones of the background, which usually classify these two kind of points with related algorithm of pattern recognition, mentioned in [3][4]. The disadvantage of optical flow method is large calculation and high requirements for hardware. Motion compensation is widely used, such as in [5][6][7], and there are also various methods of obtaining the motion model parameters, such as gray projection algorithm, feature algorithm, the classic block-matching algorithm and the bit-plane matching algorithm. In this paper, we mainly focus on feature algorithm. Gray, area, edge, feature point or corner are the commonly used feature space. The earlier corner extraction method is Moravec algorithm [8], proposed by Moravec in 1979. It is simple, but it does not have rotational invariance and is sensitive to noise, which lead to rarely used now. One of the most widely used corner detection algorithm is Harris algorithm, proposed in [9]. Though it spends more time in computing than Moravec algorithm, it solves the existing problems of the former. While Harris algorithm has its own limitations: fixed threshold and no scale invariance. For the fixed threshold problem, reference [10] proposes to set the threshold to be 0.01 times the maxima of the corner response，and reference [11] improves the 0.01 to be a variable *P*. A dual threshold method is proposed in [12]. The above methods only utilize the maxima of the corner response function. In this paper, we make full use of the local maxima of corner response function, and propose to set the threshold to be *k* times the mean of the local maxima of corner response function, here *k* is a constant. Compared with the traditional Harris algorithm, experimental results show that this method improves the accuracy of detection. Weighing the detection accuracy and the real-time property, we detect the corners with adaptive threshold Harris algorithm mentioned above and use SIFT algorithm to describe these corners. When using background subtraction to extract moving object, the key step is background updating. The method we used in this paper improves the extraction efficiency.

This work is supported by the Fundamental Research Funds for the Central Universities HEUCF0414

2 Adaptive Threshold Harris Algorithm

To reduce the impact of noise on the image, a Gaussian window is used to smooth the image at the beginning. For any point (x, y) in the smoothed image, the changed value of gray it moves a small displacement in any direction, showed in (1).

$$E_{x,y} = (x, y)M(x, y)^T \tag{1}$$

where the expression of matrix M is

$$M = W_{x,y} * \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} \tag{2}$$

$*$ denotes convolution operation in x and y. Define the corner response function of pixel (x, y) as (3).

$$R = \det(M) - k \bullet tr^2(M) \tag{3}$$

where $\det(M)$ denotes the determinant of matrix M and $tr(M)$ denotes the trace of matrix M . k is a constant whose value usually be 0.04-0.06. When R is the local maxima within a field and greater than the set threshold value, this pixel is marked as a corner. In this paper we propose setting the threshold to be *k* times the mean of the local maxima of corner response function. The value of *k* is 2.65. In order to make a compare with traditional Harris algorithm, we choose a common image. Experimental results are shown in Figure 1.

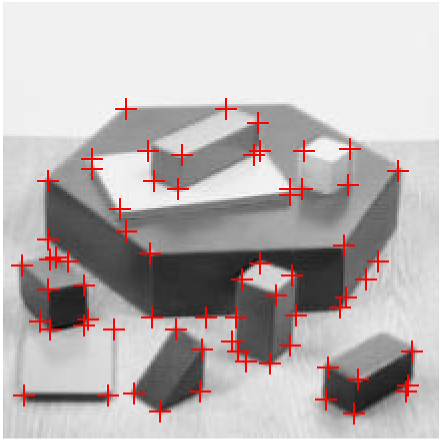

Fig. 1: Result of adaptive threshold Harris algorithm proposed in this paper

The comparison of experiment result of the adaptive threshold Harris algorithm in this paper and the Harris algorithm proposed in [11] as well as the traditional Harris algorithm shown in Table 1.

Table 1: Comparison of Experiment Results

Algorithm	Correct Corners	Missed Corners	False Corners
In [11]	51	9	4
In [12]	46	14	10
In this Paper	52	8	4

As can be seen from Table 1, the adaptive threshold Harris algorithm proposed in this paper detects more correct corners and less missed corners, that is improving the detection accuracy.

3 Improved SIFT Algorithm

SIFT algorithm is proposed in 1999 by Canadian D. G. Lowe in [13] and summarized in 2004 in [14]. This section we will briefly describe the SIFT algorithm, and then introduce how to combine SIFT algorithm with the adaptive threshold Harris algorithm proposed above, which is the improved SIFT algorithm.

3.1 SIFT Algorithm

The first step of SIFT algorithm is scale-space extrema detection. The expression of variable-scale Gaussian function is

$$G(x,y,\sigma)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (4)$$

where σ is scale-space factor. Using $G(x,y,\sigma)$ to convolve with original image $I(x,y)$ to obtain the scale space function:

$$L(x,y,\sigma)=G(x,y,\sigma)*I(x,y) \qquad (5)$$

According to the size of original image and top layer image, the number of layers of the Gaussian pyramid can be determined. Then create a Gaussian pyramid through down-sampling, and make the difference of two nearby scales separated by a constant multiplicative factor k, we get $D(x,y,\sigma)$:

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma)-G(x,y,\sigma))*I(x,y) \\ &= L(x,y,k\sigma)-L(x,y,\sigma) \end{aligned} \qquad (6)$$

The local maxima and minima of $D(x,y,\sigma)$ may be the potential interest points that are invariant to scale and orientation. The process of detecting local extrema is as follow: compare the center point of the 3×3 window with the rest 8 points and nine neighbors in the scale above and below, 26 points in total.

The second step of SIFT algorithm is keypoints localization. the Taylor expansion of $D(x,y,\sigma)$ is

$$D(X)=D+\frac{\partial D^T}{\partial X}X+\frac{1}{2}X^T\frac{\partial^2 D}{\partial X^2}X \qquad (7)$$

where X is

$$X=(x,y,\sigma)^T \qquad (8)$$

Set the derivative of the Taylor expansion of $D(x,y,\sigma)$ to zero and get the offset of extrema $\hat{X}$ as follow:

$$\hat{X}=-\frac{\partial^2 D^{-1}}{\partial X^2}\frac{\partial D}{\partial X} \qquad (9)$$

Then, adjust the position of current keypoint according to this offset and interpolate at the new position repeatedly until it converges.

DOG operator will generate a strong edge response, so we have to eliminate the edge response points. Get the Hessian matrix

$$H=\begin{pmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{pmatrix} \qquad (10)$$

Let α and β, the eigenvalues of matrix H, denote the gradient of x and y direction respectively. According to the mathematical knowledge we have

$$tr(H)=D_{xx}+D_{yy}=\alpha+\beta \qquad (11)$$

$$det(H)=D_{xx}D_{yy}-(D_{xy})^2=\alpha\beta \qquad (12)$$

Assume that α is greater than β, and let $\alpha=r\beta$. Then,

$$\frac{Tr(H)^2}{Det(H)}=\frac{(\alpha+\beta)^2}{\alpha\beta}=\frac{(r+1)^2}{r}=r+\frac{1}{r}+2 \qquad (13)$$

The larger the value of r is, the larger gap between α and β will be, that is the situation of edge pixel. So we set a certain value of r as threshold, if the potential feature point satisfies (10), it will be saved, or else eliminate it.

$$\frac{Tr(H)^2}{Det(H)}<\frac{(r+1)^2}{r} \qquad (14)$$

The third step of SIFT algorithm is orientation assignment. For the keypoints detected above, collect the gradients and directions of pixels within their 3σ neighborhood window to build histograms. The maximum value of the histogram is the main direction of the keypoint. The gradient magnitude, $m(x, y)$, and orientation, $\theta(x, y)$,of pixel (x, y) are as follows.

$$m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2} \quad (15)$$

$$\theta(x,y)=\tan^{-1}\left(\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)}\right) \quad (16)$$

Finally, the keypoints containing information of location, scale and orientation will be described with a vector, used for corner matching in next step.

3.2 SIFT Algorithm Combined With Harris Algorithm

Though SIFT feature is invariant for translation, rotation, scaling and brightness, its feature points detection process is complicated and time-consuming, which is bad for real-time system. To ensure the accuracy and the detection speed of the feature point, in this paper, we detect the corners or feature points with adaptive threshold Harris algorithm proposed in section 2, and then, establish feature descriptor vectors containing information of location, scale and orientation for those corners with SIFT algorithm. Finally, the matching process is completed based on the vectors. The improved SIFT algorithm take less time, as it do not need to search the whole image, but only focus on the corner that already detected by the adaptive threshold Harris algorithm.

4 Corner Matching

There are several traditional feature point matching algorithms such as sum of squared differences (SSD)[15], cross correlation(CC) [16] and normalized cross correlation (NCC)[17]. SSD is simple but very sensitive to changes in illumination. Instead of using the gray value of the feature point's neighborhood, CC relay on the cross correlation of the gray value of the feature point's neighborhood. This method is still relatively sensitive to changes in illumination. On this basis, the normalized cross-correlation is proposed to eliminate the influence of illumination. Taking the calculation amount into account, Euclidean distance will be used to measure the similarity of two corners. When the feature descriptor vectors of two images are established, take one corner in the first image and compute the Euclidean distance with corners in second image to find the two nearest one. If the ratio of the nearest distance and the second nearest distance less than a certain threshold, the nearest distance corner will be taken as the matching corner. The threshold become smaller, the numbers of matching corner will be less but more robust. The recommended threshold by Lowe is 0.8. This paper, when we take 0.45 it reach the best results.

5 Background Global Motion Compensation

Numbers of matching corners will be obtained after the above steps. This section we mainly introduce how to use these matching corners to obtain the affine transformation matrix M , and realize global motion compensation.

5.1 Parameters estimation

The two-dimensional affine transformation model is used to describe the motion of image sensor. For a certain corner, assuming (x, y) is the coordinate at time T_{k-1} and (x', y') at time T_k . The transformation relationship between them is

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_0 & a_1 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_2 \\ a_5 \end{pmatrix} \quad (17)$$

Write the affine transformation matrix M as follow:

$$M = \begin{pmatrix} a_0 & a_1 & a_2 \\ a_3 & a_4 & a_5 \\ 0 & 0 & 1 \end{pmatrix} \quad (18)$$

In order to obtain the above six parameters, theoretically speaking, only three non-collinear matches are needed. But if the location of one corner among the three matches exist a large error, the affine transformation matrix based on these three matches also exist a large error. In practical application, the least squares method is usually used to get approximate solution to more than $N(N \geq 3)$ matches.

5.2 Mismatches Remove with RANSAC Algorithm

There are some local feature points, such as the one be covered or located in the target or the one out of the field of view. While all we want are global feature points that correctly matched and located in background area. The least squares method can not distinguish these two feature points and eliminate local feature points. Therefore, we need a robust method to estimate the global motion parameters and the RANSAC algorithm is one of the most representatives. Specific application process is as follows:

1）The feature points set of reference image is $P=\{(x_i, y_i) | i = 1, 2, \cdots, n\}$ and the set of target image is $Q=\{(x_i', y_i') | i = 1, 2, \cdots, n\}$. Randomly select three matches as a sample subset S , which used to initialize matrix M ;

2）Select those points that meet the requirements of the initialized M ,that is the inliers set S^* ;

3）If the number of feature points in set S^* is greater than N , according to the least squares method to estimate the six parameters with all feature points in set S^* .

4）If the number of feature points in set S^* is less than N , resample a new subset S and circulate step 2) and 3);

5）After a certain number of sampling , we get the maximum inliers set that used to calculate the final parameters with the least squares method.

5.3 Global Motion Compensation

When we get the affine transformation matrix M , we can compensate the motion of background for the current image. Since many of the coordinates calculated by compensation are not an integer, then the gray values can not be directly obtained, so the coordinates need to be adjusted. Let (u,v) denote the motion vector. Then the adjusted coordinates of pixel (x,y) in current image would be $(x+u, y+v)$ and we have

$$\begin{cases} x+u = x_0 + \mu_x \\ y+v = y_0 + \mu_y \end{cases} \tag{19}$$

where x_0 is the integer part of $x+u$ and μ_x is the fractional part, while y_0 is the integer part of $y+v$ and μ_y is the fractional part. Geometry method, including forward mapping and backward mapping, is commonly used when adjusting coordinate. In this paper, we choose backward mapping method, which is mapping from output to input. The output gradation value is calculated from the four input pixels.

5.4 Experimental Results and Analysis

The effectiveness of the parameters estimation and motion compensation is verified through experiments. Experimental platform is a PC with Intel（R）Core（TM）i3-2120 3.3GHz CPU, 4G memory and Windows XP OS. Fig. 2 denotes the two nearby original images. The detection results of these two images by improved SIFT algorithm are shown in Fig.3. The final matching points are shown in Fig.4 after eliminating the mismatches by RANSAC algorithm.

Fig.2 Original images

The experimental results show the proposed parameters estimation and motion compensation method used in this paper are effective. It can better compensate the offset of the background caused by the motion of camera, which is conducive to detect and track the moving target in dynamic background.

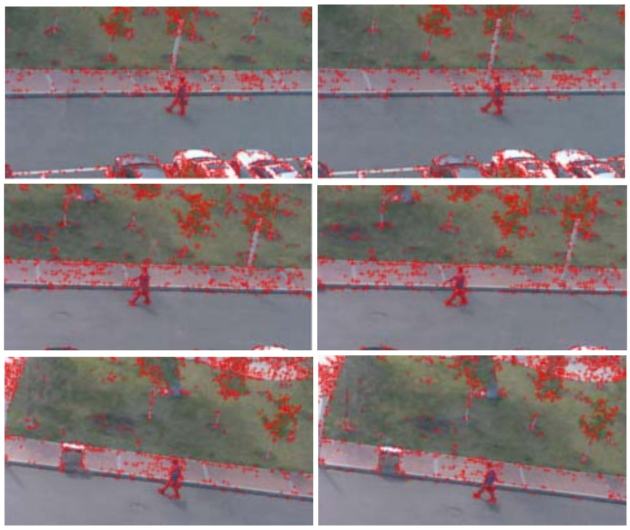

Fig.3 Corner detection results of the improved SIFT algorithm

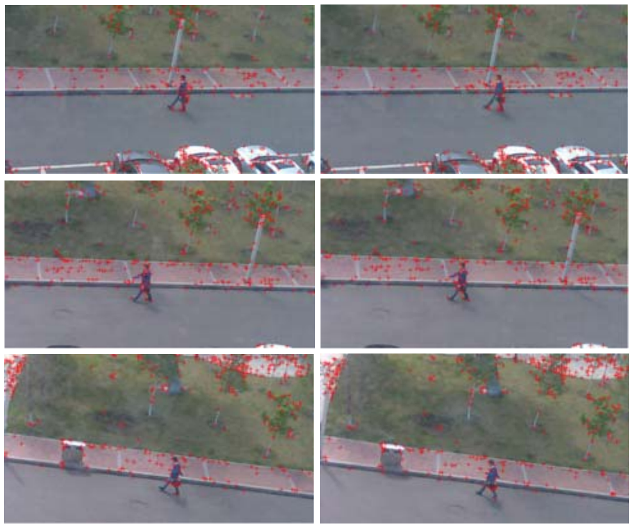

Fig.4: The final matching points

6 Moving Target Detection

Background subtraction method is applied to detect the moving object and we made a comparison of detection result between the one with motion compensation and the without.

6.1 Background subtraction method

In this paper, moving object is extracted by using background subtraction. The way to updating background is as follow: if the area detected in the current frame belongs to the foreground, then, the background template of this area remains the same, which means no updating. If it does not or the background of last frame becomes the background of current frame, then, the background of the current frame will be used as a new background template. The mathematical model of this process is:

$$M(x,y) = \begin{cases} 0, \left| f_t(x,y) - b_t(x,y) \right| < Th \\ 1, \left| f_t(x,y) - b_t(x,y) \right| \geq Th \end{cases} \tag{20}$$

where $M(x,y)=0$ denotes the background area and $M(x,y)=1$ denotes the foreground area. $f_t(x,y)$ denotes the pixel in current frame and $b_t(x,y)$ denotes the pixel in the background template of current moment. $T(\bullet)$ denotes the affine transformation, and Th is the threshold. Define

$$b_t^{'}(x,y)=T(b_{t-1}^{'}(x,y)) \tag{21}$$

Then, background update as follow formula.

$$b_t(x,y)=\begin{cases} f_t(x,y) \ \ M(x,y)=0 \ or \ b^{'}(x,y)=0 \\ b_t(x,y) \ \ M(x,y)=1 \end{cases} \tag{22}$$

Since the background updating related to the affine transformation matrix, whereas the affine transformation matrix is obtained by the correct matching feature points, so the accuracy of the affine transformation matrix subject to the corner detection and matching algorithms, and thus, the more corners, the background model is more accurate.

6.2 Detection results

The motion compensation basically eliminates the influence caused by movement of image sensor. Then, the moving object can be detected by the background subtraction method. However, global motion compensation does not fully offset the motion of background due to the changes in illumination, the distortion of the image sensor, the accuracy of compensation algorithm and other reasons. Therefore, there will be some small holes, isolated dots, small block-like noise and thin lines. Mathematical morphological filtering is a good method to eliminate the noise. The left side images in Fig.5 denote the result of frame difference algorithm without compensation and the right side images in Fig.5 denote the background subtraction algorithm after compensation. It can be seen from Fig.5 that the moving object is detected very well with the method proposed in this paper.

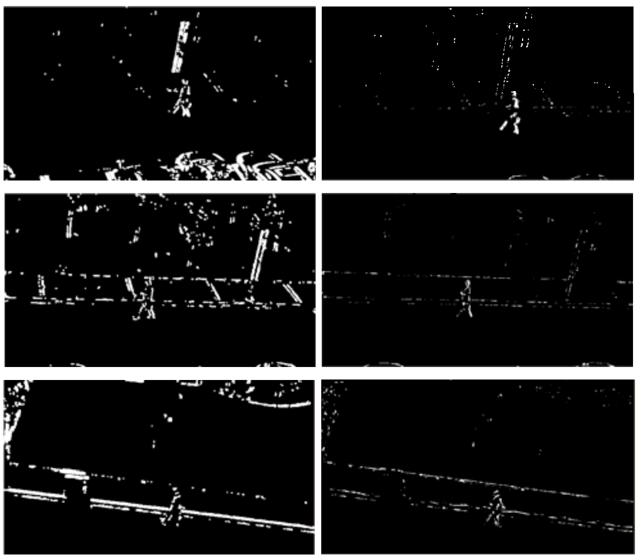

Fig. 5: The Comparison results

7 Conclusion

In real life, most application of moving target detection requires to consider the change of background, namely dynamic background. An adaptive threshold Harris algorithm is proposed in this paper and experimental result show that this method is more effective in detecting corners, and combining with SIFT algorithm makes this algorithm more efficient. Dynamic background updating guarantees the effectiveness of background subtraction algorithm. Experimental result showed that the method used in this paper can better obtain the motion parameters, and ultimately the detection result of moving target under dynamic background is good.

References

[1] Dong Zhang, Motion detection for rapidly moving cameras in fully 3D scenes, in *Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, 2010: 444–449.

[2] Y. Sheikh, O. Javcd, and T. Kanade, Background subtraction for freely moving cameras, in *International Conference on Computer Vision (ICCV)*,2009:1219-1225.

[3] Xiaqiong Yu, Xiangning Chen, and Mengnan Gao, Motion Detection in Dynamic Scenes Based on Fuzzy C-means Clustering, in *International Conference on Communication Systems and Network Technologies*, 2012:306-310.

[4] Qiang Zhu, Shai Avidan, Kwang-Ting Cheng, Learning a sparse corner-based representation for time-varying background modeling, in *International Conference on Computer Vision (ICCV)*, 2005:1-8.

[5] Jiang Zhe, Ding Wenrui, Aerial video image object detection and tracing based on motion vector compensation and statistic analysis, in *Asia Pacific Conference on Microelectronics & Electronics,* 2009:302-305.

[6] J. Kim, J. Shin, D. Kim, FH-adaptive motion compensation for moving object detection in an active camera, in *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 2009:532-537.

[7] Zhang Shun, Su Xiuqin, Global motion compensation for image sequences and motion object detection, in *International Conference on Computer Application and System Modeling (ICCASM)*, 2010:406-409.

[8] H.P. Moravec, Visual mapping by a robot rover, in *Proceedings of the 6th International joint Conference on Artificial intelligence*, 1979: 598–600.

[9] C. Harris, M. Stephens, A combined corner and edge detector, in *4th Alvey Vision Conference*, 1988: 147–151.

[10] Gao Bo, Jian Zhong Cao, A robust image registration algorithm used for panoramic image mosaic, in *International Conference on Image Analysis and Signal Processing (IASP)*, 2012: 1–4.

[11] Shi-zhe Shen, Improved Harris corner detection algorithm based on auto-adaptive threshold and pre-selection, in *Journal of Data Acquisition &Processing*, 26(2):207-213, 2011.

[12] Yan-ming Mao, An improved corner detection method based on Harris, in *COMPUTER TECHNOLOGY AND DEVELOPMENT*, 19(5):130-133, 2009.

[13] D.G. Lowe, Object recognition from local scale-invariant features, in *International Conference on Computer Vision*, 1999: 1150-1157.

[14] D.G. Lowe, Distinctive image features from scale-invariant keypoints, in *International Journal of Computer Vision*, 60(2): 91-110, 2004.

[15] D.I. Barnea, H.F. Silverman, A class of algorithms for fast digital image registration, in *IEEE Transactions on Computing*, 21(2):179-186, 1972.

[16] P. Anandan, Measuring visual motion from image sequences, *Doctoral dissertation*, Computer Science Department, University of Massachusetts, May 1987.

[17] Z. Zhang, R. Deriche, O. Faugeras, A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry, in *Artificial Intelligence* , 78(1-2):87-119, 1995.