A moving body recognition method based on self-adaption weighting meanshift algorithm

Shoujia WANG[a], *Wenhui LI[b], Tianshu YOU, Mingyu SUN and Yongjian Liu

College of Computer Science and Technology, Jilin University, Changchun, China 130012

[a]wangsj035@163.com, *[b]liwh@jlu.edu.cn

Keywords: Meanshift; Kalman filter; body tracking; perturbation insensitivity; self-adaption

Abstract. Classic meanshift algorithm is widely used in pattern recognition, but its accuracy decrease large when there is perturbation in background. In this paper, to solve this problem, we put forward an improved meanshift algorithm which is based on self-adaption weighting meanshift algorithm. At first, we catch moving object area. Secondly, we use capture to set new weighting coefficient. Then we track target model by the new coefficient to decrease effect of background. The experiment result of moving body in supervisory control video shows perturbation insensitivity, robustness and stability.

Introduction

In study area of computer vision, tracking and recognition of moving object is important. It is used widely in human society. Meanshift algorithm is a classic algorithm in this area because of its small iteration times and computation complexity [1].This concept is given by a paper about probability density gradient estimation of Fukunaga firstly. It is an estimate method based on density gradient with no parameter. To be more precise, this method means an iteration process. In this process, we compute shift of mean value by current point, then we move this point to its shift value and use it as new start point. The iteration ends with a given condition.

Since Fukunaga put forward it in 1975, meanshift algorithm has been large improved. Cheng improves classic meanshift algorithm for two directions in 1995 [2]. He defines a class of kernel functions. These function recomputed contribution of samples by distance. In other words, the contribution of mean shift vector is different when distance of sample to shift point is different. Then he set a weighting coefficient to discriminate different importance of sample points. In year 2002, Comaniciu use meanshift algorithm to analysis of character space [3]. Nowadays, this algorithm is widely used in image smooth and segmentation.

In this paper, we improve meanshift algorithm. We revise weighting value of background to eliminate static area. Then we track and predict moving body by combined Kalman filter. This new method improves robustness and accuracy of moving body tracking. Experiment shows accuracy when there are some perturbation or masking in background.

Tracking algorithm

Meanshift algorithm

The basic meanshift algorithm is shown in formula (1) [4].

$$M_h(x) \equiv \frac{1}{k} \sum_{x_i \in S_h} (x_i - x) \tag{1}$$

In formula (1), S_h is a high dimensions sphere area with radius h. It is a set of all point y in formula (2). Parameter k denotes there are k points of all x_i mapping into S_h

$$S_h(x) \equiv \left\{ y : (y-x)^T (y-x) \le h^2 \right\} \tag{2}$$

In classic meanshift algorithm, we find that contributions of sample points into S_h are same from formula (1), regardless of distance to x. But as we known, a point is more effective of statistic characteristics estimation nearby x when it is nearer to x. So when we use distance into computation of $M_h(x)$, moreover, give weighting coefficient for each sample, classic meanshift is improved to formula (3).

$$M(x) \equiv \frac{\sum_{i=1}^{n} G_H(x_i - x)w(x_i)(x_i - x)}{\sum_{i=1}^{n} G_H(x_i - x)w(x_i)}$$

(3)

Improved meanshift algorithm

Firstly, we lead into kernel function in definition 1[5][6].

Definition 1. Let X=d dimensions Euclidean space, vector x=a point of X, x's module $\|x\|^2 = x^T x$, R=real number field. A function K(x) is called a kernel function when there is a profile function k:[0,∞]→R of function K: X→R, in other words, $K(x) = k\left(\|x\|^2\right)$, and k is not negative, not increasing (if a<b, then k(a)≥k(b)), piecewise continuous and $\int_0^\infty k(r)dr < \infty$.

We use gray or color distribute to describe an object. To assume center is x_0, we use formula (4) to describe the object.

$$\hat{q}_u = C \sum_{i=1}^{n} k\left(\left\|\frac{x_i^s - x_0}{h}\right\|^2\right) \delta\left[b\left(x_i^s\right) - u\right]$$

(4)

Another object with position y can be described with formula (5).

$$\hat{p}_u(y) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{x_i^s - y}{h}\right\|^2\right) \delta\left[b\left(x_i^s\right) - u\right]$$

(5)

So object tracking can be simplify to find a y makes $\hat{p}_u(y)$ is most similar to $\hat{q}_u$. It can be measured by Bhattacharrya coefficient $\hat{\rho}(y)$. It is formula (6) and formula (7) is a computation method.

$$\hat{\rho}(y) \equiv \rho\left[p(y), q\right] = \sum_{u=1}^{m} \sqrt{p_u(y)\hat{q}_u}$$

(6)

$$\rho\left[p(y), q\right] \approx \frac{1}{2} \sum_{u=1}^{m} \sqrt{p(y_0)q_u} + \frac{1}{2} \sum_{u=1}^{m} p_u(y)\sqrt{\frac{q_u}{p_u(y_0)}}$$

(7)

When we distribute weighting of each x_i, a usual method is to use it into computation of object. But there are computation complexity and error when background is not simple. So we put forward a self-adaption weighting method based on foreground capture result. It is shown in formula (8).

$$w_i = \begin{cases} \sum_{u=1}^{m} \delta[b(x_i) - u]\sqrt{\frac{q_u}{p_u(y_0)}} & \text{if I=1} \\ 0 & \text{if I=0} \end{cases}$$

(8)

Then, $\hat{\rho}(y)$ can be rewritten to formula (9).

$$\rho\left[p(y),q\right]\begin{cases} \dfrac{1}{2}\displaystyle\sum_{u=1}^{m}\sqrt{p(y_0)q_u} + \dfrac{C_h}{2}\displaystyle\sum_{i=1}^{n} w_i k\left(\left\|\dfrac{y-x_i}{h}\right\|^2\right) & \text{if I=1} \\[2em] \dfrac{1}{2}\displaystyle\sum_{u=1}^{m}\sqrt{p(y_0)q_u} & \text{if I=0} \end{cases}$$

$$(9)$$

In new method, we don't change weighting of moving area. It means that importance of every sample points doesn't be affected. We change static areas in tracking to zero because they can be treated same to background. So tracking area decreases to moving area. It means all areas of background are avoided. So accuracy and robustness are higher and computational time is less.

Kalman filter

Kalman filter used in predict correctional estimate [7]. Its effect is well when to minimize covariance error and used widely in tracking system of image sequence.

Standard Kalman filter divide a discrete dynamics system to two processes. They are dynamics system and observation system. State equation is corresponding to dynamic system and Observation system is corresponding to observation system. Sate equation produces prediction state vector of time k by use input vector, dynamics system before time k and dynamics noise vector. Observation equation produces observation vector of time k by use state vector of time k, observation system before time k and observation noise vector. Then we solve state replace equation to find state revised value of time k by use both prediction state vector and observation vector. By iteration, we can use Kalman filter to predict system state and revise state value.

Experiment

Steps of algorithm are shown below.

Step 1. Catch continuous images frame and set them to I_n, I_{n+1}, …, I_{n+m};
Step 2. Gets foreground moving area by use object detection method;
Step 3. Use Kalman filter to predict next position of moving object;
Step 4. Use improved meanshift iteration algorithm to track moving area.
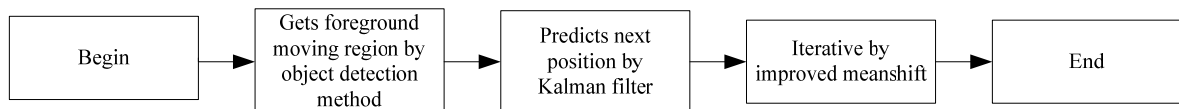Flow chart of algorithm is shown in figure 1.

Fig.1 Flow chart of algorithm

The experiment result is shown in figure 2 and 3. Figure 2 shows result of classic meanshift and tracking effect of our algorithm is shown in figure 3.
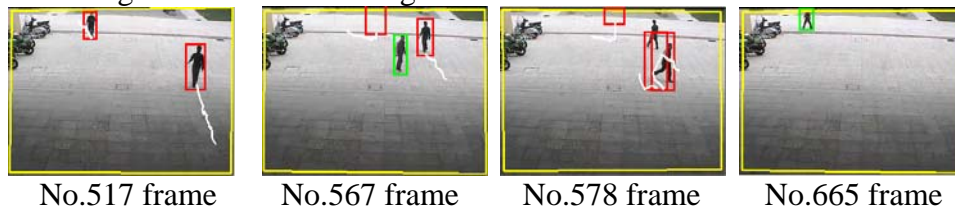

| No.517 frame | No.567 frame | No.578 frame | No.665 frame |

Fig.2 Tracking effect of classic meanshift algorithm

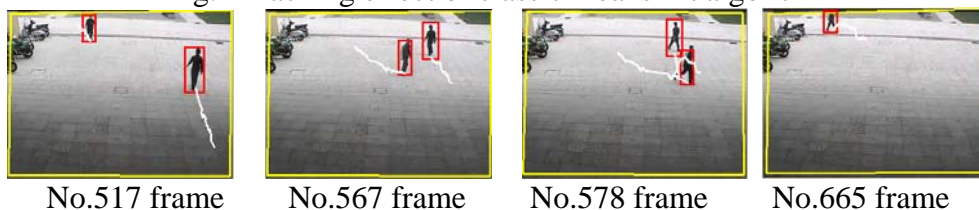| No.517 frame | No.567 frame | No.578 frame | No.665 frame |

Fig.3 Tracking effect of improved meanshift algorithm

In figure 2 and 3, green frames are new moving object, red frames are tracking object. It is to say that green frames mean there are some objects missed.

In experiment results, we find that No.517 frame shows normal when a similar color object goes into tracking window. Then a tracking object missed from No.576 frame and changed to a new tracking object. There is a center of tracking window diverged till to object missed in No.578 frame. Then in No.665 frame, tracking misses and changed to a new tracking object because of partly hide of tracking object. In figure 3, since there is normal in No.517 frame, all tracking objects are not missed though there is overlapping or hide in No.567, No.578 and No.665 frame.

Conclusion

In this paper, we put forward a self-adaption weighting meanshift method to recognize and track moving body. We reset weighting coefficient automatically by detect tracking area is foreground or background. Then it can make better tracking. To compare with exist method, the new method decreases computation amount of background, and changed tracking area and predict to moving body. So tracking effect is better than others. Experiment result validates accuracy, effectiveness and robustness. In next step, we will find a method to solve tracking miss with object whole hidden.

Acknowledgments

References

[1] D Comaniciu, V Ramesh, P Meer, in: Real-time tracking of non-rigid objects using mean shift, Computer Vision and Pattern Recognition, (2000).

[2] Y Cheng: Mean shift, mode seeking, and clustering. Pattern Analysis and Machine Intelligence. Vol. 17-8 (1995), p. 790-799.

[3] D Comaniciu, P Meer: Mean shift: a robust approach toward feature space analysis. Pattern Analysis and Machine Intelligence. Vol. 24-5 (2002), p. 603-619.

[4] ZHU Shengli，ZHU Shanan: An algorithm of mean shift template update based a group of kalman filters. Journal of Image and Graphics. Vol. 12-3 (2007), p. 460-465.

[5] RT Collins, in: Mean-Shift blob tracking through scale space, Computer Vision and Pattern Recognition, (2003).

[6] E Magglo, A Cavallaro, in: Multi-part target representation for color tracking, International Conference on Image Processing(ICIP), (2005).

[7] SJ Julier, JK Uhlmann, in: A new extension of the Kalman filter to nonlinear systems, Signal processing, sensor fusion, and target recognition VI, (1997).