AN ADAPTIVE MEAN SHIFT TRACKING METHOD USING MULTISCALE IMAGES

ZHUO-LIN JIANG[1], SHAO-FA LI[1], DONG-FA GAO[2]

[1] School of Computer Science & Engineering, South China University of Technology, Guangzhou, 510641, China
[2] School of Informatics, Guangdong University of Foreign Studies, Guangzhou, 510420, China
E-MAIL: zhljiang@scut.edu.cn, csshfli@scut.edu.cn, gaodf@hubu.edu.cn

Abstract

An adaptive mean shift tracking method for object tracking using multiscale images is presented in this paper. A bandwidth matrix and a Gaussian kernel are used to extend the definition of target model. The method can exactly estimate the position of the tracked object using multiscale images from Gaussian pyramid. The tracking method determines the parameters of kernel bandwidth by maximizing the lower bound of a log-likelihood function, which is derived from a kernel density estimate with the bandwidth matrix and the modified weight function. The experimental results show that it can averagely converge in 2.55 iterations per frame.

Keywords: Mean shift; object tracking; Gaussian pyramid; automatic bandwidth selection; multiscale

1. Introduction

Real-time and autonomous object tracking remains a challenging task for various vision applications [1,2,3], such as visual surveillance, driver assistance, human action understanding, etc. Mean shift is a nonparametric and iterative procedure to search a local maximum of a density function, which was proposed by Fukunaga and Hostetler in 1975 [4]. Cheng [5] applied the mean shift algorithm to computer vision domain and demonstrated its perfect potential in clustering and multistart global optimization. Comaniciu et al have adopted mean shift for image filtering and image segmentation [2,6]. Due to its simplicity and robustness, the mean shift algorithm has become popular in object tracking in recent years [1,3,7,8,9,10,11].

The performance of the mean shift algorithm strongly depends on the kernel bandwidth parameters [12], which determine the support region of interest for mean shift searches. A problematic issue is how to choose the bandwidth parameters, which is still a leading issue in kernel-based object tracking [1,12,13]. Comaniciu et al [9] determine the bandwidth by repeating the mean shift iterative procedure, using the size of plus or minus 10 percent of the current bandwidth. The bandwidth yielding the largest Bhattacharyya coefficient is chosen as the optimal bandwidth. This method cannot exactly track an object with variable scales and the object will be lost at last. Two solutions of bandwidth selection are presented in [13] for mean shift kernel, but they are time-consuming for real-time applications. Collins [12] uses the DOG kernels and the Epanechnikov kernel in scale space to obtain the spatial location and scale. However, the Epanechnikov shadow in the scale dimension will be a flat kernel, yielding the computed bandwidth that is equal to a weighted average of scales in scale space. To our knowledge, the result is not satisfying for the practical needs.

In this paper we present an extension of Comaniciu's [9] idea to achieve object tracking both spatially and in scale. Our method not only finds the position of the object, but also chooses the optimal kernel bandwidth that describes the scale of the object. It extends the tracking method in [9] in three ways. First, bandwidth matrix is allowed to define the target model. Second, mean shift is used in multiscale images from the Gaussian pyramid to get the accurate spatial location. Third, optimal bandwidth parameters are automatically selected, by maximizing the lower bound of the log-likelihood derived from a kernel density estimate with the bandwidth matrix and the modified weight function. Experimental results show that the method is feasible.

2. Mean shift and target model

2.1. Mean shift procedure

Mean shift only employs one bandwidth parameter to determine the kernel bandwidth [6]. We extended it to use a bandwidth matrix H for the better characterization and localization of a tracked object.

Given n data points $\{x_i\}_{i=1\ldots n}$ in the d-dimensional space R^d, the multivariate kernel density estimator $\hat{f}(x)$ with kernel K(x) and a d×d bandwidth matrix H is introduced in [6].

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} |H|^{-1/2} K[H^{-1/2}(x - x_i)] \tag{1}$$

where K(x) is a radial symmetric kernel

1-4244-1066-5/07/$25.00 ©2007 IEEE

satisfying $K(x)=c_{k,d}k(\|x\|^2)$. C$_{k,d}$ is a normalization constant and the function k(x) denotes the profile of the kernel K(x).

In this paper, we use two-dimensional space R^2 where $H=diag[h_x^2,h_y^2]$ and $x_i=\{x_{i,x},x_{i,y}\}^t$. Epanechnikov kernel is a frequently used kernel in mean shift [5]. As there is a great improvement in the results employing a Gaussian kernel [6], the kernel K(x) used in this paper is a Gaussian kernel, the profile of which is

$$k(x)=\begin{cases} e^{-x} & \|x\|\le\lambda \\ 0 & \|x\|>\lambda \end{cases} \qquad (2)$$

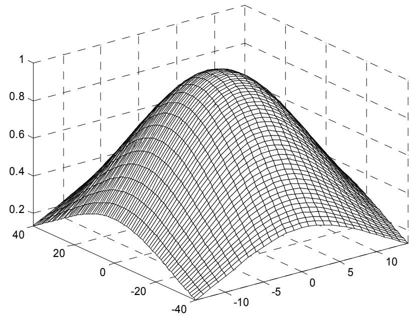


Figure 1.Bandwidth of Gaussian kernel.

Figure 1 shows the bandwidth of Gaussian kernel, the effective coordinates range of which is $x\in[-14,14]$ and $y\in[-40,40]$.As a result, the parameters of bandwidth matrix H are (h$_x$, h$_y$) = (14, 40).Assuming the derivative of the kernel profile k(x) exists for $x\in[0,\infty)$, the function g(x) is $g(x)=-k^{'}(x)$. g(x) is the profile of the kernel $G(x)=c_{g,d}g(\|x\|^2)$. The mean shift vector is defined as [6]

$$m_H(x)=\frac{\sum_{i=1}^{n}x_i g\left(\left\|H^{-1/2}(x-x_i)\right\|^2\right)}{\sum_{i=1}^{n}g\left(\left\|H^{-1/2}(x-x_i)\right\|^2\right)}-x \qquad (3)$$

The sequence $\{y_j\}_{j=1,2...}$ of successive locations of the kernel G is given by [6]

$$y_{j+1}=\frac{\sum_{i=1}^{n}x_i g\left(\left\|H^{-1/2}(y_j-x_i)\right\|^2\right)}{\sum_{i=1}^{n}g\left(\left\|H^{-1/2}(y_j-x_i)\right\|^2\right)} \quad \text{j=1,2...} \qquad (4)$$

The mean shift procedure moves toward the direction of maximum increase in the underlying density. It can converge at a local maximum point y_c that satisfies $\nabla f(\mathrm{y_c})=0$, through the successive computation of mean shift vector m$_H$(x) followed by the translation of kernel G(x).

2.2. Target model with bandwidth matrix

The rectangle used to entirely include the tracked object is called target rectangle or tracking window in this paper, which is shown in Figure 2.The horizontal size and the vertical size of tracking window can be denoted as h$_H$ and h$_V$ respectively. Once the bandwidth matrix parameters h$_x$ and h$_y$ are determined, h$_H$ and h$_V$ can be set as $h_H=2h_x$ and $h_V=2h_y$ .The center of the target rectangle is (x_c,y_c) .

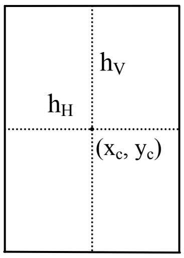


Figure 2.Target rectangle and tracking window.

We employ the bandwidth matrix H to extend the definition of target model, which is different from [9]. The function $b(x):R^2\to\{1...m\}$ associates the pixel at location x$_i$ with the color index b(x$_i$) in the kernel histogram. Given n two-dimensional pixel locations $\{x_i\}_{i=1,..,n}$ in the target rectangle centered at x$_c$, according to [6,9], the probability of the feature u=1…m in the target model with bandwidth matrix H is

$$\hat{q}_u=C_{H,q}\sum_{i=1}^{n}|H|^{-1/2}k\left(\left\|H^{-1/2}(\mathrm{x_i}-x_c)\right\|^2\right)\delta[b(x_i)-u] \quad (5)$$

where C$_{H,q}$ is the normalization constant that guarantees the summation of $\hat{q}_u$ for u=1…m is equal to one

$$C_{H,q}=\frac{1}{\sum_{i=1}^{n}|H|^{-1/2}k\left(\left\|H^{-1/2}(x_i-x_c)\right\|^2\right)} \qquad (6)$$

and δ is the Kronecker delta function. Similarly, the probability of the feature u=1…m in the target candidate for the current image is computed as [9]

$$\hat{p}_u(y)=C_{H,p}\sum_{i=1}^{n_k}|H|^{-1/2}k\left(\left\|H^{-1/2}(x_i-y)\right\|^2\right)\delta[b(x_i)-u] \quad (7)$$

where C$_{H,p}$ is a normalization constant. The similarity function defines the distance between the target model and the target candidate. It can be defined as [9]

$$d(y)=\sqrt{1-\hat{\rho}(y)} \qquad (8)$$

where $\hat{\rho}(y)\equiv\rho[\hat{p}(y),\hat{q}]=\sum_{u=1}^{m}\sqrt{\hat{p}_u(y)\hat{q}_u}$ is the Bhattacharyya coefficient. It is computed using Taylor expansion around $\hat{p}_u(\hat{y}_0)$ [3,9]

$$\rho[\hat{p}(y),\hat{q}]\approx\frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{y}_0)\hat{q}_u}+\frac{1}{2}\sum_{u=1}^{m}\hat{p}_u(y)\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{y}_0)}} \qquad (9)$$

Recalling (7) yields

$$\rho[\hat{p}(y),\hat{q}] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{y}_0)\hat{q}_u} + \frac{C_{H,p}}{2}\sum_{i=1}^{n_k}\left|H\right|^{-1/2}k\left(\left\|H^{-1/2}(x_i-y)\right\|^2\right)w_i \qquad (10)$$

where

$$w_i = \sum_{u=1}^{m}\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{y}_o)}}\,\delta[b(x_i)-u] \qquad (11)$$

To find the location of the tracked object in the current frame, the distance expression (8) should be minimized. It is equivalent to maximizing the Bhattacharyya coefficient. We employ the mean shift procedure to find the maximum of the second term in (10) when the first term is a y-independent value. The second term represents a kernel density estimate computed at y with weight function w_i in the current image. The kernel iteratively moves from y_0 to the new location y_1 for the Gaussian kernel according to following formula [9].

$$y_1 = \frac{\sum_{i=1}^{n_k} x_i w_i \exp\left(-\left\|H^{-1/2}(x_i-y_0)\right\|^2\right)}{\sum_{i=1}^{n_k} w_i \exp\left(-\left\|H^{-1/2}(x_i-y_0)\right\|^2\right)} \qquad (12)$$

where w_i

$$w_i = \sum_{u=1}^{m}\left(\frac{\hat{q}_u}{\hat{p}_u(\hat{y}_0)}\right)^2 \delta[b(x_i)-u] \qquad (13)$$

is a weight function, which is different to expression (11) defined in [9]. From our many experiments, we modify the index of the proportion factor in w_i (11) to be quadratic, enabling our method to track moving objects in fast motion.

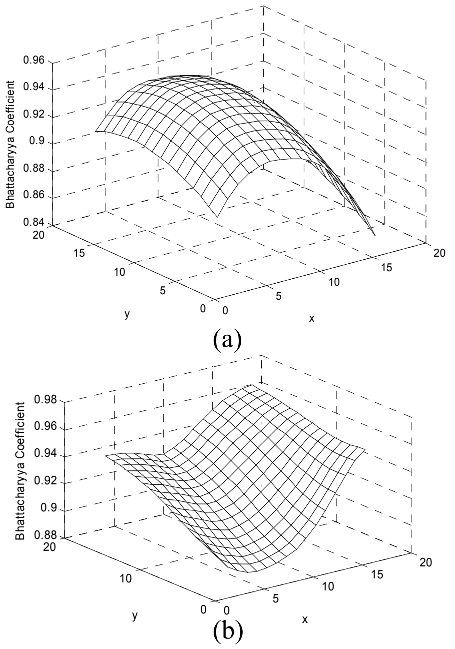
3. Target localization and bandwidth selection

Kernel bandwidth is very important for the performance of the mean shift algorithm. When kernel bandwidth is too large, it will contain many background pixels which greatly influence the histogram distribution of the target candidate. As a result, the tracking algorithm may be easily confused in the clutter background.

If the bandwidth is chosen too small, the histogram of the target candidate only includes partial pixels of the tracked object. Since the tracking window may possibly roam around the mode of underlying density, it will result in a poor localization for the tracked object.

The center of the minimal target rectangle (see Figure 2) is defined as the centroid of the tracked object in this paper. Figure 3 shows the Bhattacharyya coefficients of points in a region with a size of 16×16 pixels centered at the object centroid. When the size of the tracking window is larger than that of the tracked object, the Bhattacharyya coefficient is a monotonic decreasing function as it is shown in Figure 3(a). The function highly depends on the distance from the object centroid, where we can get the maximum of Bhattacharyya coefficient. You can find a few extrema on the curved surface when the size of tracking window is smaller than object size in Figure 3(b), which is one of the main reasons for the poor object localization.

From the theory of mean shift in [4,5,14], we can find the centroid of the tracked object if the object moves in the tracking window, because the center of the tracking window can usually overlap with the centroid of the tracked object. It can be proved experimentally that the mean shift algorithm can locate the object if the tracking window contains the majority of object pixels [14].



(a)

(b)

Figure 3.Bhattacharyya coefficient of particular region around object centriod. (a) The size of tracking window is larger than object size. (b) The size of tracking window is smaller than object size.

3.1. Target localization with multiscale images

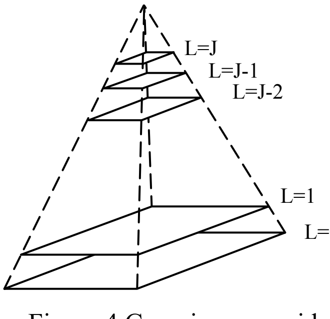


Figure 4.Gaussian pyramid.

We obtain multiscale images using the Gaussian pyramid [15] of each frame. Lowpass filtering followed by subsampling for the images in previous levels can generate the Gaussian pyramid shown in Figure 4. Each pixel value for the image at level L is computed as a weighted average of pixels in a 5×5 neighborhood at level L-1. Given the initial image $f_0(i,j)$ which has a size of M×N pixels, the level-to-level weighted average process is implemented by [15]

$$f_L(i,j)=\sum_{m=-2}^{2}\sum_{n=-2}^{2}r(m,n)f_{L-1}(2i+m,2j+n) \quad (14)$$

where r(m,n) is a separable 5×5 Gaussian low pass filter given by [15]

$$\begin{aligned}&r(m,n)=r(m)r(n)\\&r(0)=a,\quad r(1)=r(-1)=1/4,\quad r(2)=r(-2)=1/4-a/2\end{aligned} \quad (15)$$

The parameter a can be chosen from 0.3 to 0.6. The separability of r(m,n) can reduce the computational complexity in generating multiscale images. The size of the tracked object in the subsampling image becomes smaller and smaller from level to level while the kernel bandwidth remains the same as it is in the initial image $f_0(i,j)$. Once the tracking window includes most of the object pixels at level L, it can find the location of object in image $f_L(i,j)$ using the mean shift algorithm. This is the reason for us using Gaussian pyramid.

If the size of the initial image is M×N and it satisfies $M=p\times 2^J$ and $N=q\times 2^J$, J is the maximal level of image pyramid and p×q is the image size in level J. The size of subsampling image at level k is $M2^{-k}\times N2^{-k}$ where k=0,…, J. Given the current kernel bandwidth parameters h_x, h_y and J, the maximal level k of Gaussian pyramid for the current image in this paper can be obtained by the following equation.

$$\begin{aligned}&ml=\min\left(\left\lfloor\log_2^{M/h_y}\right\rfloor\left\lfloor\log_2^{N/h_x}\right\rfloor\right)\\&k=\begin{cases}ml & J\geq ml\\ J & J<ml\end{cases}\end{aligned} \quad (16)$$

where '⌊ ⌋'is the down rounded operation.

The coarse to fine approach allows to recover the location information of target from a image sequence. For the low computational cost, we only use the images at level 0 and level k. Let Y_k be the spatial location of the tracked object at level k and Y_0 be the corresponding location at level 0, they satisfy

$$Y_k=Y_0/2^k \quad (17)$$

$$Y_0=Y_k\times 2^k \quad (18)$$

We first use the mean shift algorithm to locate the object position $T_{c,k}^{'}$ in the subsampling image at level k, and then the corresponding location of $T_{c,k}^{'}$ is used as the initial location y_0 in the original image at level 0. Finally the exact position $T_{c,k}$, which we are interested to find, can be located using the mean shift algorithm.

3.2. Bandwidth selection

In order to find the location of target in the current frame, the second term in (10) which represents the kernel density estimate with the bandwidth matrix H and the modified weight function w_i (13) should be maximized

$$p(y,H)=\frac{C_{H,p}}{2}\sum_{i=1}^{n_k}|H|^{-1/2}k\left(\left\|H^{-1/2}(x_i-y)\right\|^2\right)w_i \quad (19)$$

Since it has the form of a weighted sum of component densities, it can be viewed as a mixture of parametric models [16]. Introducing Gaussian kernel into (19) yields

$$p(y,H)=\frac{C_{H,p}}{2}\sum_{i=1}^{n_k}|H|^{-1/2}\exp\left(-\left\|H^{-1/2}(x_i-y)\right\|^2\right)w_i \quad (20)$$

We know that kernel bandwidth is critical for the efficiency of the mean shift algorithm from above analysis. A preferable method of bandwidth selection is to not only choose the optimal bandwidth that satisfies the practical needs, but also increase the kernel density estimate (20).

Due to the monotonicity of logarithmic function and the use of Gaussian kernel, we define a log-likelihood function for (20) as

$$L(y,H)=\ln\sum_{i=1}^{n_k}|H|^{-1/2}\exp\left(-\left\|H^{-1/2}(x_i-y)\right\|^2\right)\frac{w_i}{C_w} \quad (21)$$

where the normalization constant $C_w=\sum_i w_i$ is derived by satisfying $\sum_i\frac{w_i}{C_w}=1$. We use the method in [16] to compute a local lower bound approximation to the log-likelihood. Usually the computed lower bound is analytically tractable. The bandwidth that maximizes the lower bound can increase both the log-likelihood and the kernel density estimate (20). Since the log function is concave, we obtain the lower bound for L(y,H) according to the Jensen inequality [16,17]

$$L(y,H)\geq\sum_{i=1}^{n_k}\frac{w_i}{C_w}\ln\left(|H|^{-1/2}\exp\left(-\left\|H^{-1/2}(x_i-y)\right\|^2\right)\right) \quad (22)$$

Let mlp(y, H) denote the lower bound of L(y, H). We take the partial derivative of mlp(y, H) regarding H^{-1} and equate it to zero

$$\frac{\partial mlp(y,H)}{\partial H^{-1}}=\sum_{i=1}^{n_k}\frac{w_i}{C_w}\left(\frac{1}{2}H-(x_i-y)(x_i-y)^T\right)=0 \quad (23)$$

It means that the kernel bandwidth can be determined by maximizing the lower bound of L(y, H). Given sample points $\{x_i=(x_{i,x},x_{i,y})\}_{i=1,\ldots,n_k}$ and the current centroid $y=(y_x,y_y)$ of target, the adaptive bandwidth matrix $H=diag[h_x^2,h_y^2]$ for the current frame is calculated by the following equations.

$$h_x=\sqrt{\frac{2\sum_{i=1}^{n_k}w_i(x_{i,x}-y_x)^2}{\sum_{i=1}^{n_k}w_i}} \quad (24)$$

$$h_y = \sqrt{\frac{2\sum_{i=1}^{n_k} w_i (x_{i,y} - y_y)^2}{\sum_{i=1}^{n_k} w_i}} \qquad (25)$$

From many experiments, we further improve the equations (24) and (25). They should add a constant to include more image pixels for tracking objects. The constants for the bandwidth parameters h_x and h_y can be defined as λ_x and λ_y respectively. The last bandwidth parameters h_x and h_y can be computed according to

$$h_x = \sqrt{\frac{2\sum_{i=1}^{n_k} w_i (x_{i,x} - y_x)^2}{\sum_{i=1}^{n_k} w_i}} + \lambda_x \qquad (26)$$

$$h_y = \sqrt{\frac{2\sum_{i=1}^{n_k} w_i (x_{i,y} - y_y)^2}{\sum_{i=1}^{n_k} w_i}} + \lambda_y \qquad (27)$$

It shows the good results in Figure 5 and 6 when $\lambda_x = 4$ and $\lambda_y = 7$.

3.3. Practical algorithm

Given the target model $\{\hat{q}_u\}_{u=1...m}$ and the object location $Y_{c,k-1}$ of last frame k-1, repeating the following steps can find the new location in current frame k and choose the optimal bandwidth parameters..

1. Compute the multiscale images using (14), (15) and (16).

2. Let the corresponding position of $Y_{c,k-1}$ be the initial position at level k according to (17), and use the mean shift algorithm to find the object position $T'_{c,k}$ at level k.

3. Let H=λH, where λ>1.

4. Employ the corresponding position of $T'_{c,k}$ as the initial location at level 0 according to (18), and use the mean shift algorithm again to get the object location $T_{c,k}$, do $Y_{c,k}$= $T_{c,k}$.

5. Calculate the bandwidth parameters h_x and h_y according to (26) and (27).

We modify bandwidth matrix H to H=λH (λ>1), which can involve more pixels in the tracking window. $Y_{c,k}$ is the location of the tracked object in the current frame, and h_x and h_y are the parameters of the bandwidth matrix H for the current frame.

4. Experiments and discussions

4.1. Experimental environments

A single color CCD camera WA-2480 was installed on a support in the laboratory, which was used to monitor the motion of objects. The camera has functions such as an automatic electronic shutter and back-light compensation. The video capture card is 100moons SDK-2000. The interval between consecutive current frames was approximately 100 ms. The experimental RGB color images have a size of 320×240 pixels. They were changed into gray images and the kernel histogram was quantized into 16 bins. We used Gaussian kernel to compute kernel histogram and perform mean shift iterations. The initial bandwidth parameters were (h_x, h_y) = (14, 40). This algorithm was implemented using Visual C++ 6.0 on a windows platform.

4.2. Experimental results

Figure 5 shows tracking examples in the case of a human being squatting down. Figure 5(a) can track the human through the entire sequence. It can obtain the location of object when the size of tracking window is larger than the object size, but it is poor in kernel bandwidth selection, which directly reflects the size of the tracked object. The poor estimation of the position and the bandwidth using the method in [9] is shown in Figure 5(b). Using the method presented in this paper can track the object both in spatial location and in scale, the results are presented in Figure 5(c).

Figure 6 shows some resultant images of human tracking in a sequence of a person approaching the camera. Figure 6(a) and 6(b) are tracking results using the fixed bandwidth mean shift algorithm and the tracking method in [9] respectively. Both of them cannot get the correct spatial location through the image sequence. Object loss even occurs in Figure 6(b). Our method can track the person spatially and in scale, the resultant images of which are shown in Figure 6(c).

Figure 7 presents the tracking trajectories using different tracking methods. These trajectories are from the successive 40 frames in the sequence of a person approaching the camera. The ground truth trajectory is used to assess the tracking error of different tracking methods. Apparently, it is shown that the plus or minus 10 percent of the bandwidth selection method [9] cannot track the person through the sequence. The trajectory of fixed bandwidth mean shift deviates significantly from the ground truth trajectory. Our method can get sufficiently close to the ground truth trajectory in the whole process of tracking. Figure 8 shows the number of iterations of our method for the successive 40 frames in the sequence of a person approaching the camera. The mean number of iterations is 2.55 per frame, which is even smaller than the typical 4 mean shift iterations in [9].

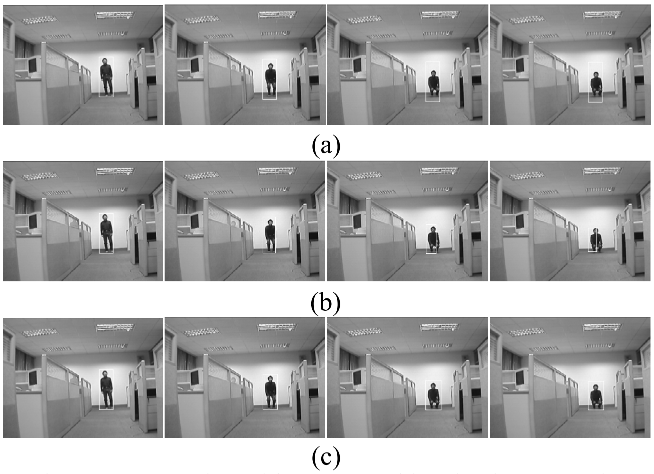


Figure 5.Examples of human tracking in the case of a person squatting down. (a) Using fixed bandwidth mean shift algorithm. (b) Using the plus or minus 10 percent of the bandwidth selection method. (c) Using the method presented in this paper. The frames 50, 62, 69, 75 are shown.

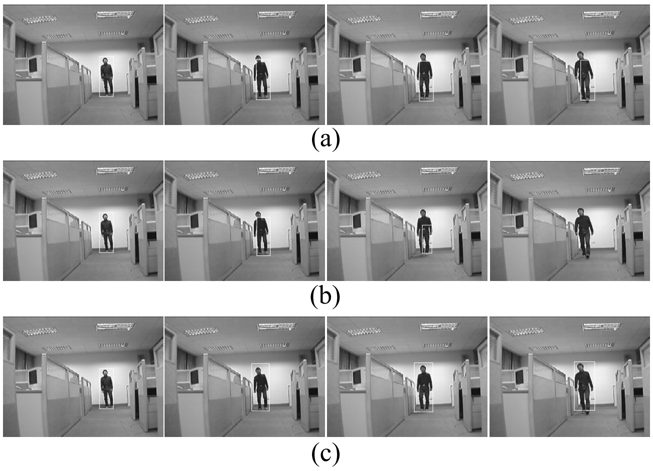
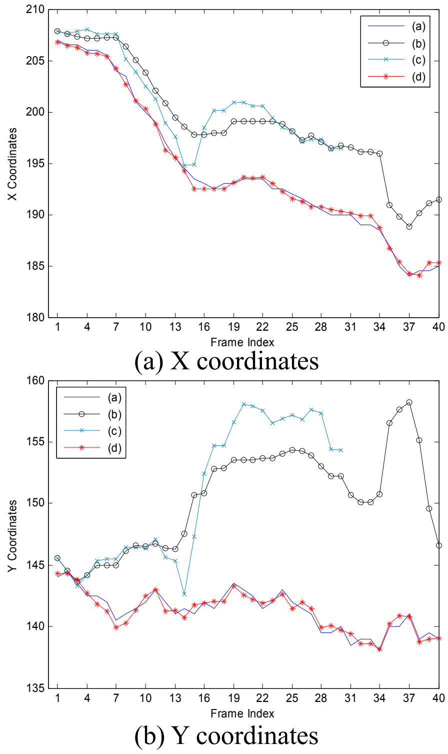


Figure 6.Examples of human tracking in the case of a person approaching the camera. (a) Using fixed bandwidth mean shift algorithm. (b) Using the plus or minus 10 percent of the bandwidth selection method. (c) Using the method presented in this paper. The frames 50, 66, 72, 82 are shown.

4.3. Discussions

We experimented with different image sequences in our laboratory to show the effectiveness and the robustness of this algorithm. The algorithm is convergent in 2.55 iterations per frame averagely. It shows that the algorithm has potential in real-time applications. The automatic bandwidth selection is favorable for the autonomous object tracking. Although this algorithm does not take complex cases such as occlusion into consideration, it can meet the needs to a great degree.



(a) X coordinates

(b) Y coordinates

Figure 7.Comparison of tracking trajectories using different methods. (a) Ground truth trajectory. (b) Tracking by fixed bandwidth mean shift. (c) Using the plus or minus 10 percent of the bandwidth selection method. (d) Tracking by our method.

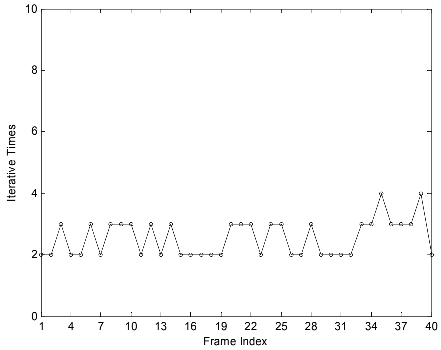


Figure 8.The number of iterations.

5. Conclusion

The experimental results show that the method can correctly track the object of interest spatially and in scale. We use the coarse to fine approach to get the correct position of object using the Gaussian pyramid representation of each frame at particular resolutions. We derive the selection formula of the kernel bandwidth from many experiments, which can describe the size of object directly. We experimented with different types of image sequences to better satisfy the needs of object localization. More importantly, the experimental results show that our algorithm is effective. Our future work includes the following: (1) methods of object tracking spatially and in scale in complex scenarios, including occlusion and irregular changes in illumination; (2) effective representation of target model encoding spatial configurations of features, which can reject the confusion among objects having identical feature distribution but different spatial configurations of features [1].

Acknowledgements

This work was supported by the key industrial research project (No.2004B10101032) in Guangdong Province, China.

References

[1] V. Parameswaran, V. Ramesh, and I. Zoghlami, “Tunable kernels for tracking”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 2179-2186, 2006.

[2] D. Comaniciu and P. Meer, “Mean shift analysis and applications”, Proceedings of the Seventh IEEE International Conference on Computer Vision, Vol. 2, pp. 1197-1203, 1999.

[3] D. Comaniciu, V. Ramesh, and P. Meer, “Real time tracking of non-rigid objects using mean shift”, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 142-149, 2000.

[4] K. Fukunaga and L. D. Hostetler, “The estimation of the gradient of a density function, with applications in pattern recognition”, IEEE Transactions on Information Theory, Vol. 21, No. 1, pp. 32-40, 1975.

[5] Y. Z. Cheng, “Mean shift, mode seeking, and clustering”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, No. 8, pp. 790-799, 1995.

[6] D. Comaniciu and P. Meer, “Mean shift: a robust approach toward feature space analysis”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 5, pp. 603-619, 2002.

[7] C. J. Yang, R. Duraiswami, and L. Davis, “Efficient mean-shift tracking via a new similarity measure”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. 176-183, 2005.

[8] F. Porikli and O. Tuzel, “Multi-kernel object tracking”, IEEE International Conference on Multimedia and Expo, pp. 1234-1237, 2005.

[9] D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-based object tracking”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 5, pp. 564-577, 2003.

[10] C. Beleznai, B. Fruhstuck, and H. Bischof, “Human tracking by mode seeking”, Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, pp. 1-6, 2005.

[11] Z. Zivkovic and B. Krose, “An EM-like algorithm for color-histogram-based object tracking”, Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. 798-803, 2004.

[12] R. T. Collins, “Mean-shift blob tracking through scale space”, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 234-240, 2003.

[13] D. Comaniciu, V. Ramesh, and P. Meer, “The variable bandwidth mean shift and data-driven scale selection”, Proceedings of Eighth IEEE International Conference on Computer Vision, Vol. 1, pp. 438-445, 2001.

[14] N. S. Peng, J. Yang, Z. Liu, and F. C. Zhang, “Automatic selection of kernel-bandwidth for mean-shift object tracking”, Journal of Software, Vol. 16, No. 9, pp. 1542-1550, 2005.

[15] P. J. Burt and E. H. Adelson, “The Laplacian Pyramid as a Compact Image Code”, IEEE Transactions on Communications, Vol. 31, No. 4, pp. 532-540, 1983.

[16] R. Harpaz and R. Haralick, “The EM algorithm as a lower bound optimization technique”, CUNY Ph.D. Program in Computer Science Technical Reports, pp. 1-14, 2006.

[17] S. L. Xu and C. H. Xue, Mathematical analysis, Tsinghua University Press, 2005.