Hindawi Publishing Corporation
Advances in Artificial Intelligence
Volume 2010, Article ID 175603, 15 pages
doi:10.1155/2010/175603

Research Article

Multibandwidth Kernel-Based Object Tracking

Aras Dargazany, Ali Soleimani, and Alireza Ahmadyfard

Department of Electrical and Robotics Engineering, Shahrood University of Technology, Shahrood, Iran

Correspondence should be addressed to Aras Dargazany, aras.dargazany@gmail.com

Received 1 August 2009; Revised 2 December 2009; Accepted 20 April 2010

Academic Editor: Ce Zhu

Object tracking using Mean Shift (MS) has been attracting considerable attention recently. In this paper, we try to deal with one of its shortcoming. Mean shift is designed to find local maxima for tracking objects. Therefore, in large target movement between two consecutive frames, the local and global modes are not the same as previous frames so that Mean Shift tracker may fail in tracking the desired object via localizing the global mode. To overcome this problem, a multibandwidth procedure is proposed to help conventional MS tracker reach the global mode of the density function using any staring points. This gradually smoothening procedure is called Multi Bandwidth Mean Shift (MBMS) which in fact smoothens the Kernel Function through a multiple kernel-based sampling procedure automatically. Since it is important for us to have less computational complexity for real-time applications, we try to decrease the number of iterations to reach the global mode. Based on our results, this proposed version of MS enables us to track an object with the same initial point much faster than conventional MS tracker.

1. Introduction

Using a kernel function as a density estimator are methods in image processing which drew much attention. Object tracking [1–7] with MS is a nonparametric technique, introduced in [8–10]. Actually, it is a mode detector algorithm in the density distribution space. This method assigns weights to the pixels within a window. It is proposed that MS iteration step size can be adapted [9]. It also said MS and Newton algorithm are connected [11]. MS is utilized in object tracking as an alternative for particle filtering tracking [2, 12, 13]. The Kernel scale can also be updated [3]. To avoid changes in target representation, multiple kernel functions can be used in order to maintain the pixel location values [5, 7, 14]. Another way toward using a kernel function is similarity function [15]. The main problems with MS: The first is failure caused in rapid movement of an object in two consecutive frames; the second is that MS is a local optimizing technique, that is, we should not expect to localize the global and optimal mode; the third is slow tracking. In fact, MS searching procedure is initialized by the tracked object in the last frame. Flattening the optimization surface (created by windowing around the initial point) is used to handle sampling method through finding the global maximum [16–18]. The multi-bandwidth kernel function can flatten the cost function (i.e., Likelihood Surface) to avoid local modes rather than global one. Through increasing the bandwidth of the kernel, all modes will be unified into one main mode whose peak location is the nearest point to the global optimal mode as we desired. Then, by using the convergence point of this mode of unified likelihood surface, the first step will be easily modified through MS iterations. The bandwidth determines the degree of flatness of the surface. In this way, the number and the position of modes are being evolved and changed slowly. Lower computational cost is more desirable in real-time tracking. Using nearest neighbor of a sample in MS iteration and kd-tree to reduce the number of nearest neighbors decrease the cost [1, 19]. The density distribution is described through clustering the feature space also to lessen the cost [20]. Quasi-Newton is also then put forward to linearize the rate of convergence [11, 21]. Many attempts have been made to enhance the efficient bound optimization algorithm [22–24]. Our proposed method can detect an object globally. This technique can prevail over the recent methods and also extend the last technical efforts [25]. This paper is summarized as follows: an overview and reviewing analysis over MS is given in Section 2. The proposed Method will be elaborated along with enhancement of MS iterations in detecting the global mode of cost function in Section 3. Implementation of the proposed method and how faster this proposed method can function are revealed in Section 4. Using color as feature, automatic initialization, how to cope with background problem, and being a self-localizer are all stated with experimental results in Section 5. At the end, the conclusion comes in Section 6.

2. Mean Shift

The conventional and original version is a nonparametric kernel density estimator

$$\widehat{f}_K(x) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(x - x_i). \tag{1}$$

According to (1), $K_H(x) = |\mathbf{H}|^{-(1/2)} K(\mathbf{H}^{-(1/2)}x)$ where K is a kernel function (i.e., here Gaussian function) with 2D bandwidth as $\mathbf{H} \in R^{d\times d}$, which is equal to $\mathbf{H} = h^2\mathbf{I}$ (i.e., identity matrix) [9, 10]. In this way, we can have kernel estimator as

$$\widehat{f}_K(x) = \frac{c_k}{nh^d}\sum_{i=1}^{n} k\left(\left\|\frac{x - x_i}{h}\right\|^2\right), \tag{2}$$

formulated through the following equation:

$$\begin{gathered}
\widehat{\nabla} f_K(x) \equiv \nabla \widehat{f}_K(x) = \frac{2c_k}{h^2 c_g}\widehat{f}_G(x)\cdot m_G(x) = 0, \\
\widehat{f}_G(x) = \frac{c_g}{nh^d}\sum_{i=1}^{n} g\left(\left\|\frac{x - x_i}{h}\right\|^2\right), \\
m_G(x) = \frac{\sum_{i=1}^{n} x_i g\left(\|(x - x_i)/h\|^2\right)}{\sum_{i=1}^{n} g\left(\|(x - x_i)/h\|^2\right)} - x,
\end{gathered} \tag{3}$$

$K(x)$ is kernel function and its gradient is $g(x)$. Here $m_G(x)$ is a vector created by MS procedure. These equations result in:

$$x \longleftarrow \frac{\sum_{i=1}^{n} x_i g\left(\|(x - x_i)/h\|^2\right)}{\sum_{i=1}^{n} g\left(\|(x - x_i)/h\|^2\right)}. \tag{4}$$

In this paper, it is proposed to use multiple bandwidths in a conventional mean shift tracker (i.e., a broad bandwidth tracking a larger motion). A broad bandwidth played the central role to help tracking a larger motion. Due to the smoothness incorporated by the large bandwidth, the fixed point iteration can track due to converging faster. It is argued that the bandwidths can be automatically obtained. However, it can be seen below the overall algorithm for choosing effect of the algorithm in detail. The algorithm is also shown and explained below. Since we are dealing with an automatic bandwidth selection, the optimal bandwidth is the main and final goal after smoothening likelihood surface (i.e., cost function). The optimal bandwidth is the bandwidth in which global mode and other local modes are clear enough to seek (i.e., initial state of likelihood surface) and in the same bandwidth seeking about the global mode will take place by MBMS (Multi-Bandwidth Mean Shift).

In fact, there are a variety of ways to select the optimal bandwidth for an automatic optimal bandwidth selecting procedure but in this automatic procedure, first we calculate the optimal bandwidth according to [26, 27] in order to have a minimum AMISE (i.e., Asymptotic Mean Integrated Square Error) as introduced in [26, 27], AMISE is a an estimate of distance between two different densities for evaluating the performance of a kernel density estimator [28]. Having found the optimal bandwidth, we mean to find a large bandwidth sufficiently enough through the proposed MB (i.e., Multi-Bandwidth) procedure. By sufficiently this large bandwidth, we desire a bandwidth which is much larger than the optimal bandwidth with the minimum AMISE.

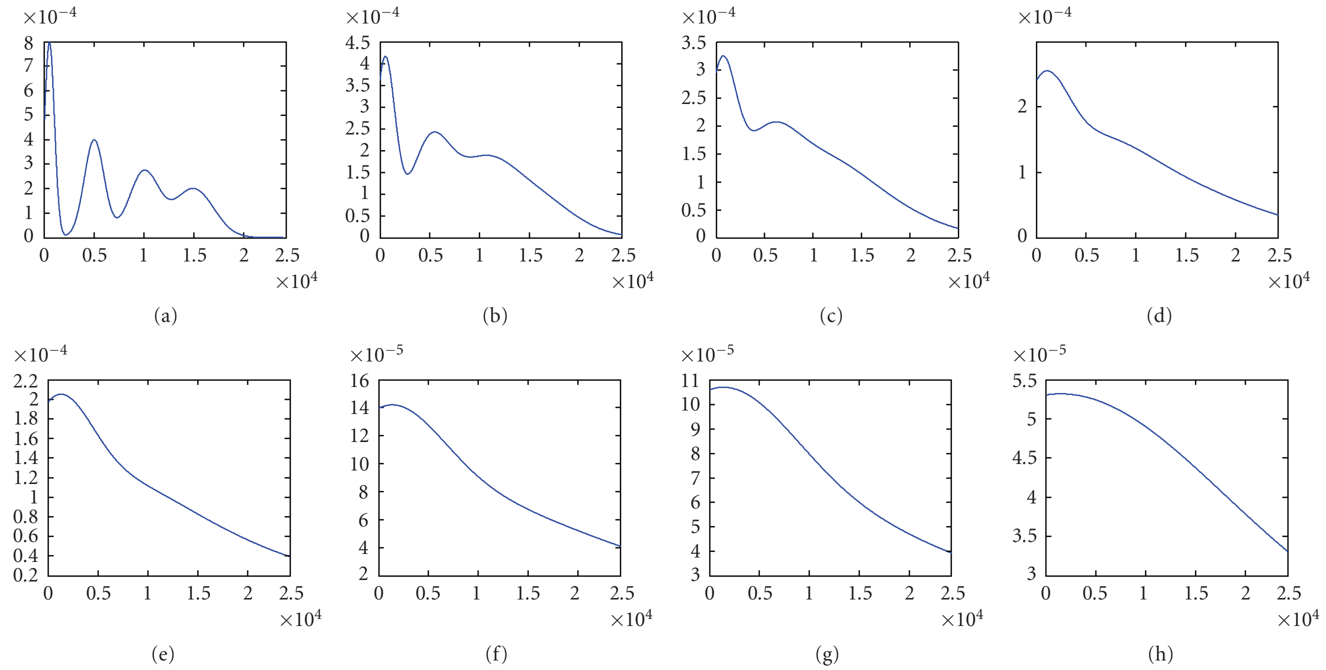
This large bandwidth is meant to create a unimodal likelihood surface which is consisting of a single mode. This mode will be used by Mean Shift procedure in the first run. It does not matter whether this mode is global or not. Having found the final location of this mode using Mean Shift helps us achieve finally the global mode via the MB procedure in the optimal bandwidth (i.e., very last selected bandwidth) using MS iterations.

Having observed the above equations, therefore we can have now the optimal bandwidth at the last stage of MBMS, in which all modes especially global mode are clear. If we use three or four times as much as the optimal bandwidth as you can see in 1D and 2D likelihood surfaces (i.e., Cost Function) Figures 1 and 2, we will have a unimodal surface then. By detecting this lonely single mode, the closest point in the best basin of attraction, we can reach the real global mode in more stable way without facing any other local mode.

3. The Proposed Method

Monotonic decrease of bandwidth ends to h0 as the experimental optimal bandwidth. Figure 1 shows a one-dimensional Gaussian mixture with some modes which are all going to be unified and to change into a unimodal surface via the bandwidth increasing trend [29].With a multi-bandwidth process, it will lead to the evolution of modes as illustrated in Figure 1. In this way, seeking the global mode would not be trapped in the local modes. The algorithm and the procedure are very simple as you see in the Figure 1. The great advantage of this method is that the starting point location is not important at all. This paper is based specifically on Gaussian kernel due to its monotonicity [30]. The optimal bandwidth is h0. The Gaussian kernel can also be transformed to reduce the cost [29, 31, 32].

There is also a method to find the minima instead of maxima [33]. In order to find an important mode among others, variable selection of bandwidth is utilized [34]. In our work, we can guarantee the tracking with the proper and manual bandwidth selection but it also can be extended through automatic estimation of bandwidth using some features like the target region variance [4]. At the largest



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 1: One-dimensional Gaussian mixture surface changes with the bandwidth: 500, 1000, 1500, 2500, 3500, 5500, 7500, and 15500.

bandwidth, we have a unimodal surface. Via MS iteration the convergence at the largest bandwidth (unimodal) will be easily achieved as it illustrated in Figure 2. It is mainly argued that there are multiple bandwidths in the proposed method. However, it is shown how these multiple bandwidths collaborate during bandwidth selection procedure in MBMS (i.e., Multi-Bandwidth Mean Shift). Here, the strategy of the use of these multiple bandwidth selection is given below.

MBMS algorithm:

(1) selecting the sequence of bandwidth h_m as $m = M, \dots, 0$ (i.e., Multi-bandwidth smoothening procedure),

(2) a starting location for first MB (multi-bandwidth) procedure and converge using h_m in $\widehat{f}_{h_M,K}(\cdot)$ which is $\widehat{x}^{(M)}$ using MS (Mean Shift),

(3) we run MS for each h_m (i.e., indicated in 1) to get the convergence position $\widehat{x}^{(m)}$ with the initial position $\widehat{x}^{(m+1)}$, this means that the convergence position achieved in previous bandwidth. Finally, reaching $\widehat{x}^{(0)}$ means that we have located the global mode on the likelihood surface (i.e., Cost Function).

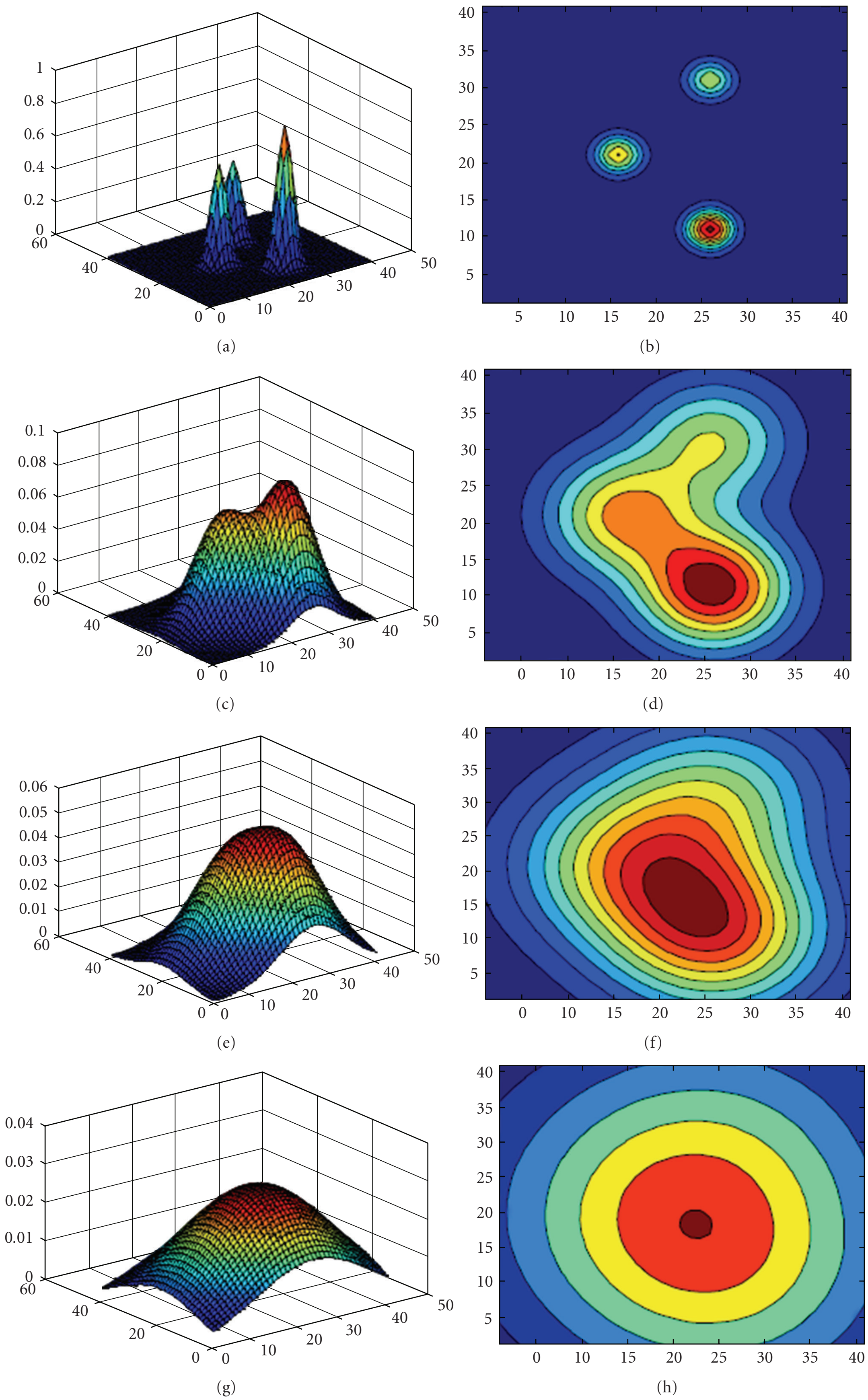
4. The Improved MS

Multi-bandwidth selecting and utilizing cost us some delay and computational burden. MS is proved to be a quadratic bound optimization according to [6, 9, 11]. This algorithm is an ascending approach with adaptable iteration size. In MS, no default parameter is needed to be set at first. It has been reasoned that a Gaussian mixture model can be optimized by fixed-point bound optimization method [35]. This method is also applicable to the surface created by other kernel functions [24]. Because of its speed, the bound optimization algorithms has a slow convergence [22–24]. Suppose that the similarity function for two frames is defined with $\rho(x, x^{(\kappa)})$ so that the adaptable step size can be determined as

$$x^{(\kappa+1)} = x^{(\kappa)} + \beta\left[\arg\max_{x} \rho\left(x, x^{(\kappa)}\right) - x^{(\kappa)}\right]. \tag{5}$$

β is a learning rate as you see above. If we want a more reliable convergence, it should be $0 < \beta < 2$ as proved and generalized in [22, 24]. In our work, it is set $\beta = 1$, then it begins initializing and running MS iteratively. Gaussian kernel function is helping us because its gradient equals to the original function so that their application in weight assignment and flattening the surface with the same bandwidth is quite similar [28]. This analysis is validated by our experimental results.

4.1. Experimental Result. We run the two algorithms for the same data sets and insert the results in Table 1 and Figures 4 and 5. The starting point is initialized in different location. As you see the improved MS achieves better results than original MS and we can claim that it functions even faster than PF [30, 36]. The proposed algorithm outperforms MS in number and step-size of iterations. It can be also compared with the Quasi-Newton method implemented in [21, 37–39]. This developed MS is implemented in MATLAB programming space and is run in an Asus notebook inside Vista windows Pentium 2 GHz CPU. The proposed MS is more different than conventional MS. In comparison with other linear bound fixed-point optimization algorithms such as Quasi-Newton [21], our proposed method is more accurate and



(a)

(b)

(c)

(d)

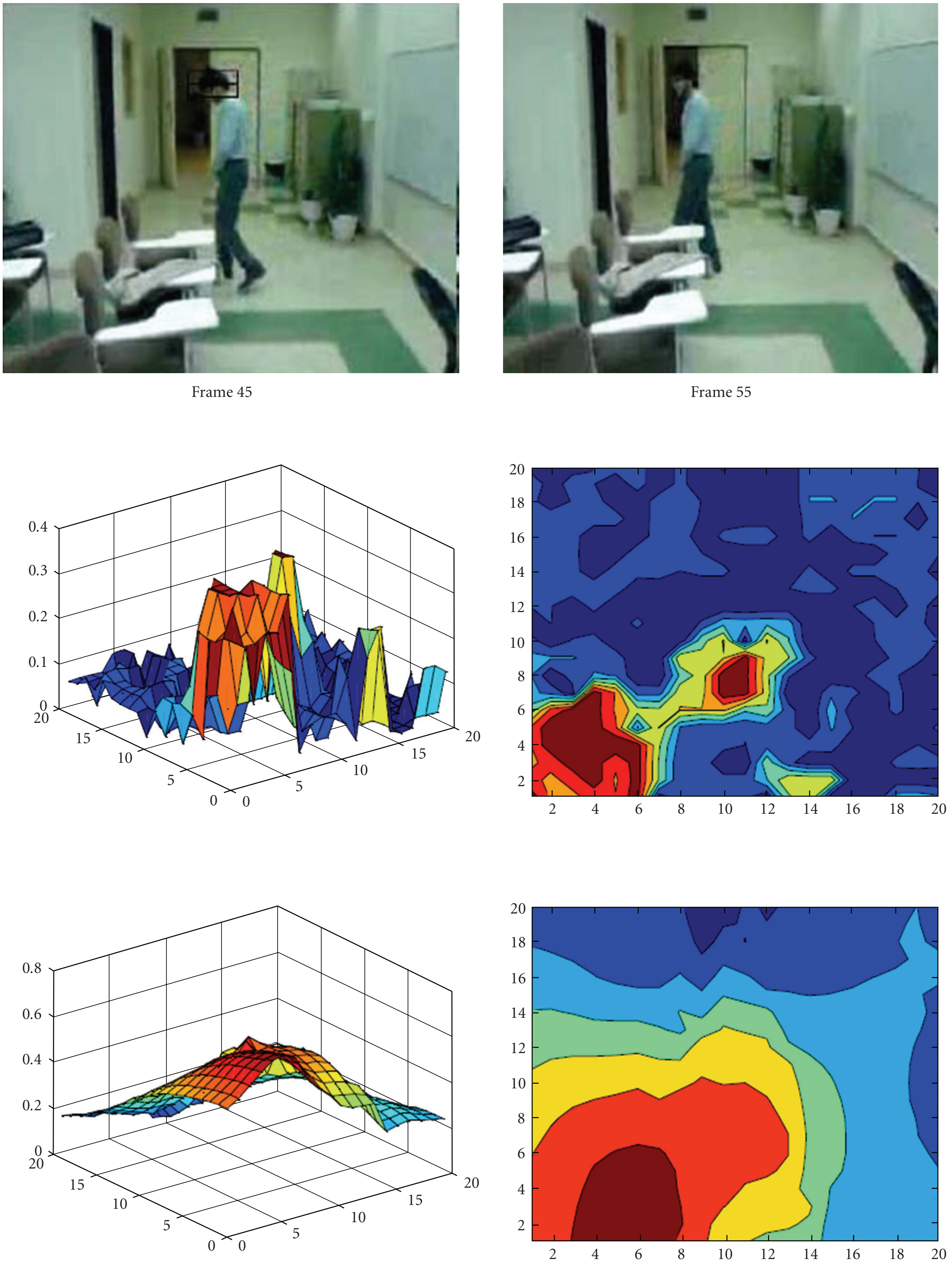
(e)

(f)

(g)

(h)

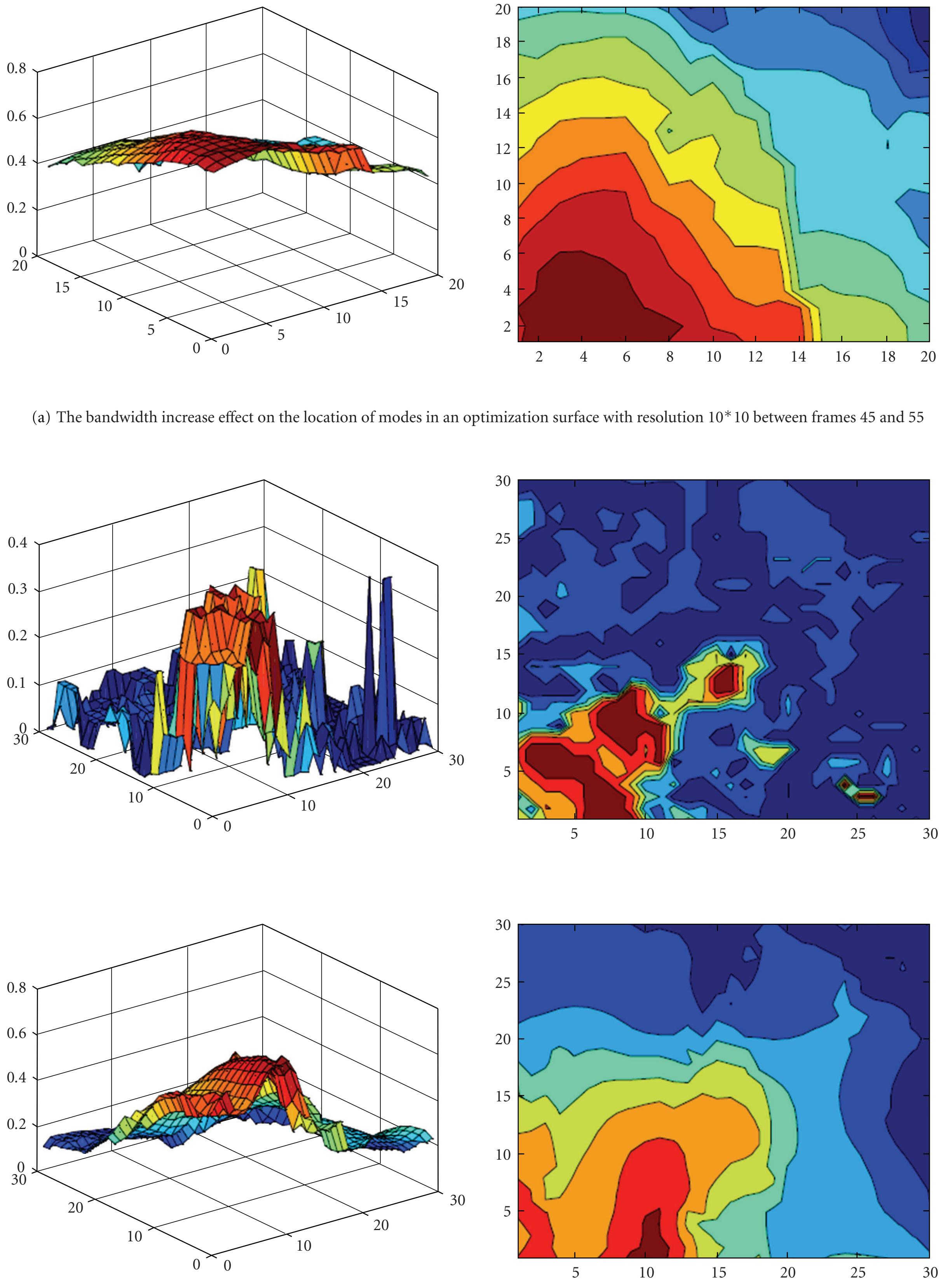
Figure 2: Two-dimensional Gaussian mixture surface changes with the bandwidth: [0.13 0;0 0.13], [1.3 0;0 1.3], [2.6 0;0 2.6], and [6.5 0;0 6.5].


Frame 45

Frame 55

(a)

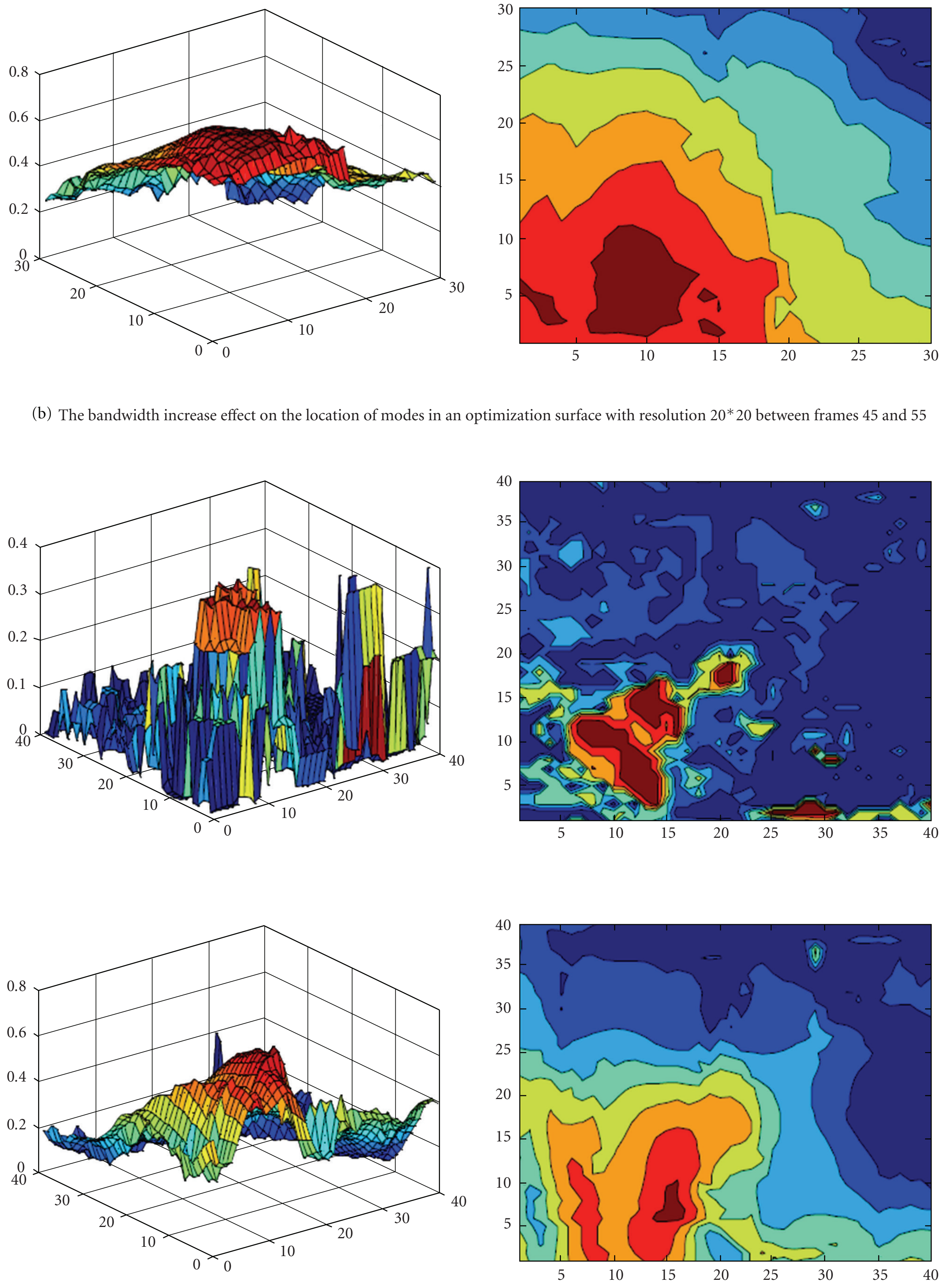
Figure 3: Continued.



(a) The bandwidth increase effect on the location of modes in an optimization surface with resolution 10*10 between frames 45 and 55

(b)

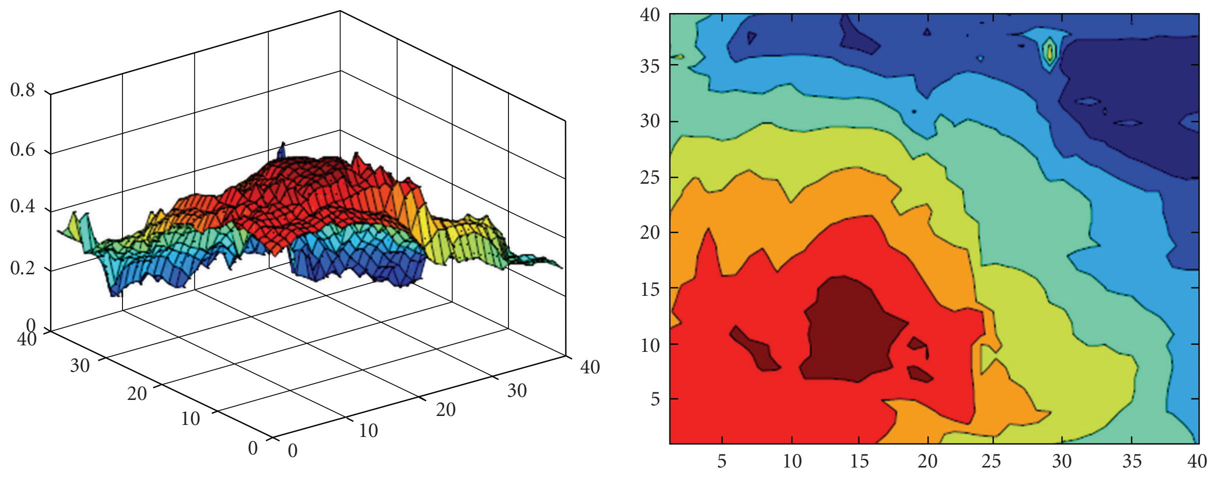
Figure 3: Continued.



(b) The bandwidth increase effect on the location of modes in an optimization surface with resolution 20*20 between frames 45 and 55

(c)

Figure 3: Continued.



(c) The bandwidth increase effect on the location of modes in an optimization surface with resolution 30*30 between frames 45 and 55

Figure 3

robust. Because of the possibility of overshooting due to sufficiently large step-size occurrence this method can also be utilized and adopted in clustering through segmentation, since it is potential enough to gather patterns into a cluster as a global and local mode.

5. Tracking Application and Implementation

Implementing the algorithm and the experimental result on an object is represented in a normalized rectangle region as the target region. We choose color as feature in target model p and target candidate q region. As it is clear, we should measure the similarity between the two selected regions in two consecutive frames through similarity measurement criteria $\hat{\rho}(x) = \rho(\hat{q}, \hat{p}(x))$ such as Bhattacharyya, Kullback-Leibler or Matusita, respectively in [2, 4, 5, 40, 41]. The pixel of rectangle region $\{y_i\}_{i=1}^{n}$ are being assigned weight according to their distance from the center of normalized circle surrounded in the rectangle, less weight and value they have through Gaussian function distribution. Via the weighted pixels, their colors in RGB scale will be extracted. Then the color would be sorted into m Bin histogram with their weight, consequently their value would be summed up in their R, G and B color bins of the histogram. This process is followed in both target model and candidate regions

$$\hat{p}_u(x) = \frac{1}{c}\sum_{i=1}^{n} k\left(\|y_i\|^2\right)\delta\left(\left(b(y_i) - u\right)\right), \tag{6}$$

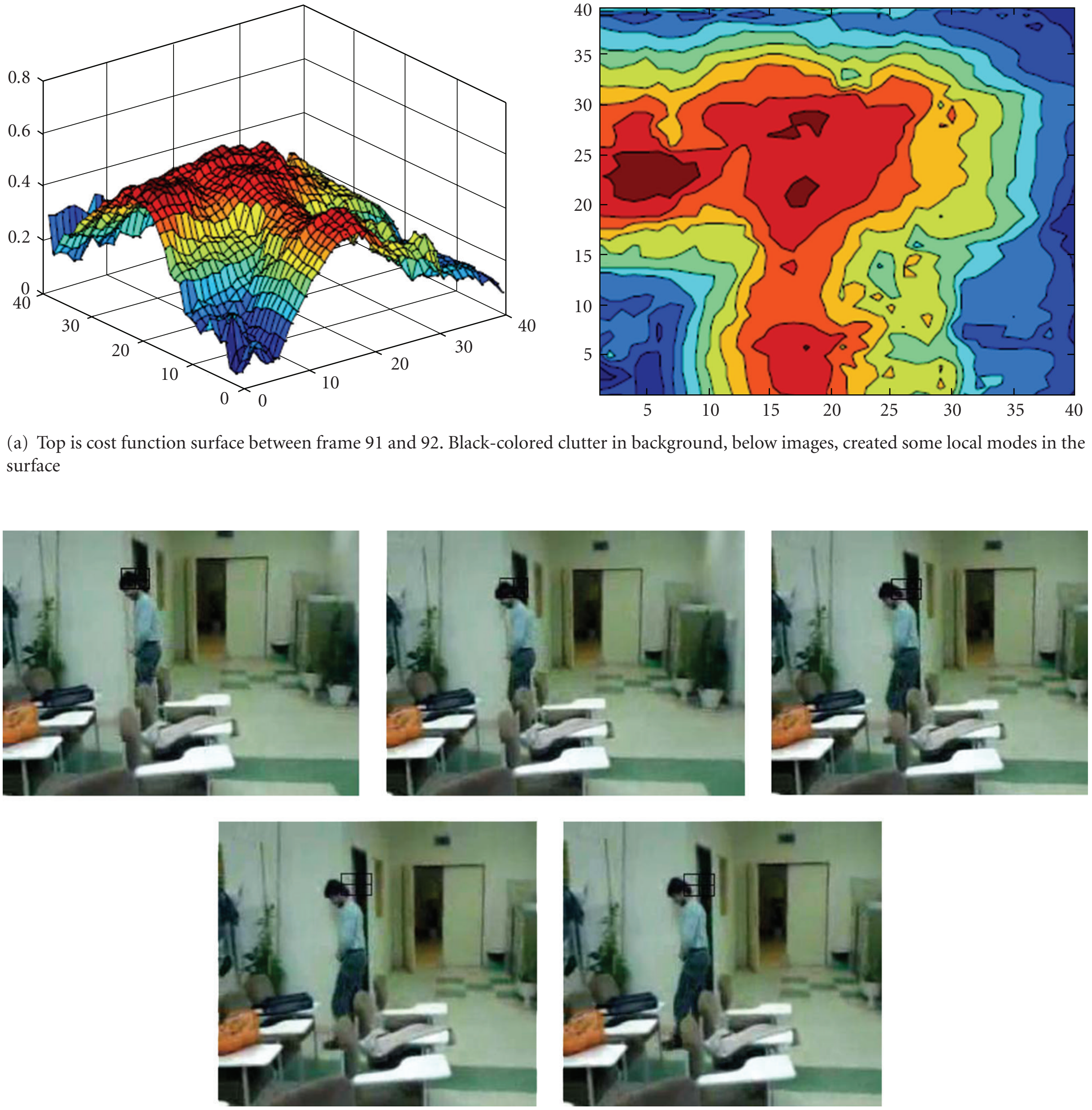
p and q are the normalized density value of the target model and candidate. Each pixel value $\{y_i\}_{i=1}^{n}$ is converted into histogram bin via its color accompanied by its weight. The similar process is performed upon the both target model and candidate. With the initial point x the tracking procedure started through approximating the distance between the place and location of the object in the two consecutive frames Δx. Then the new location of object would be found and object is tracked

$$\Delta x^* = \arg\max_{\Delta x} \rho\left(\hat{q}, \hat{p}\left(x^* + \Delta x\right)\right). \tag{7}$$

Using the Tailor expansion, the linear first-order extension helps us solve the optimization problem efficiently via MS iteration at the initial point x through extending similarity [2, 5].

5.1. Location Approximation. Conventional MS cannot seek a global mode in presence of local mode due to the fixed-bandwidth which is created by rapid motion, illumination changes, clutter and occlusion as shown in Figure 3. In our experiment, it does not matter for improved MS the initial point to start. It can easily locate the target fast. In the examples, there may be different number of bandwidth but entirely they are 4 or 3 for all tracking sequences as clarified in Figure 3. We are using color as feature but of course it is possible to be taking advantage of other features as well like motion and orientation. If we do not use the multi-bandwidth procedure, most of the searches will end to local modes unless they are initially located at the closest basin points to the global mode in Figure 3. Using the broad bandwidth kernel function at first MS run will let us find this closest basin point near to the global mode.

5.2. Object Location and Track. Entirely all of the methods in object tracking, ever written and proposed could have covered some of the weak points in this field, but there are some common problems in all of them: All have problem in finding the object location in large distance movement between two successive frames. The starting-point for them



(a) Top is cost function surface between frame 91 and 92. Black-colored clutter in background, below images, created some local modes in the surface

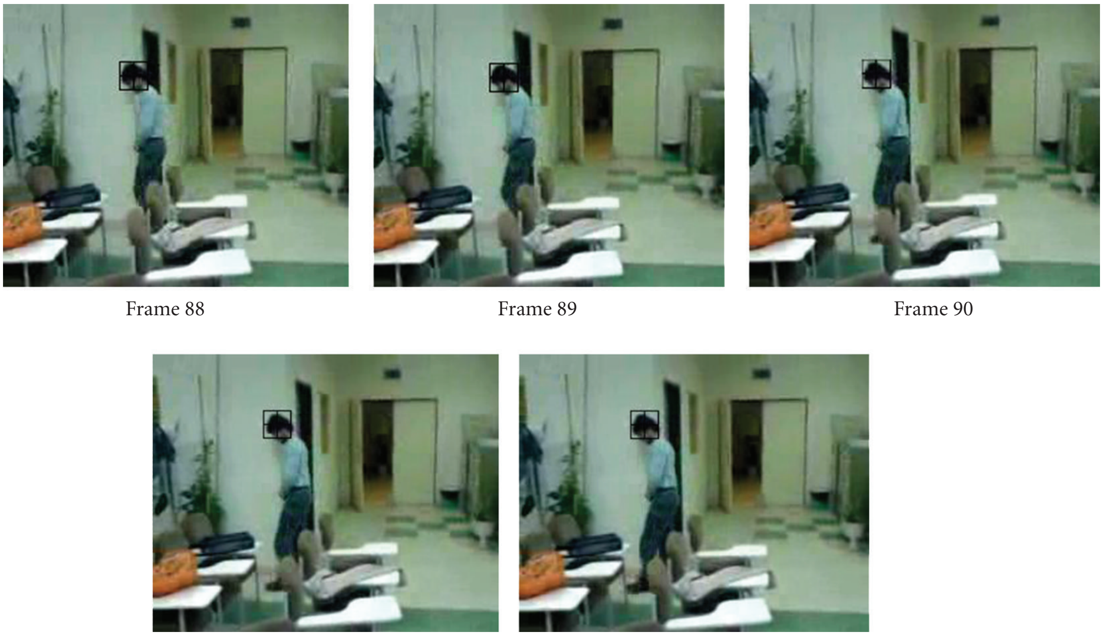
(b) Frames 88, 89, 90, 91, and 92 show us a tracking failure due to the black-colored clutter in background

FIGURE 4: Top is cost function surface between frame 91 and 92.

all is very important to track the object correctly. Background problem such as clutter, occlusion, and illumination changes can completely influence the tracking path and cause failure exactly as illustrated in Figure 4. They are not really capable to self-recovering from their failure due to the same local modes existing in Figure 4(a). It failed in tracking as shown in Figure 4(b).

The conventional MS has all of these problems described above as well. In our work, we have enabled the tracker to be robust enough in different initial points in an image by considering the efficiency and efficacy of bandwidth variation of kernel function through adaptive step-size iteration. We are actually utilizing an object detector incorporated in localizing procedure to recover from any failure when occurred. It was also previously proposed to use a detector for Particle filtering tracking-based [42]. By choosing a broad bandwidth we try to pass other local modes to reach the basin of attraction for the desired global mode. Some implementation problems:

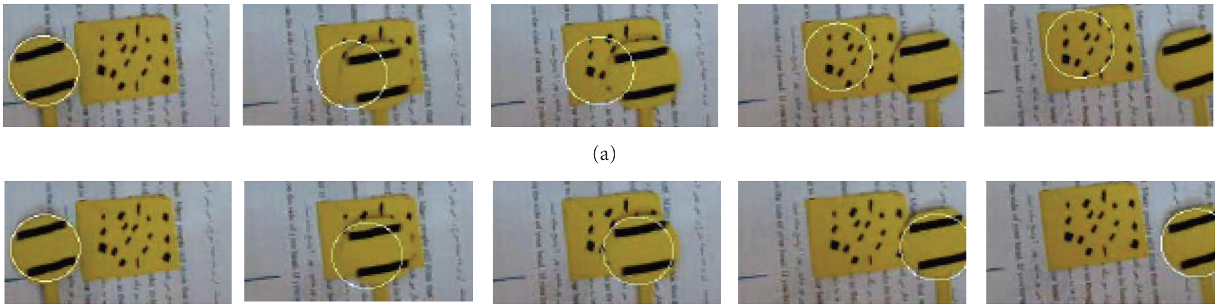
The multi-bandwidth tracker starts in a 3 or 4 bandwidth shifting iteration in an MS procedure. It is worth to say that through using color as feature so that there may be some unwanted modes created just because of the difference between two points color values [43] which may cause a local



Frame 88 Frame 89 Frame 90

Frame 91 Frame 92

Figure 5: Multi-bandwidth tracker did not fail to track in this cluttered sequence (multimodal surface).



(a)

Frame 1 Frame 14 Frame 17 Frame 26 Frame 33

(b)

Figure 6: Top row: MS, Bottom Row: MBMS.

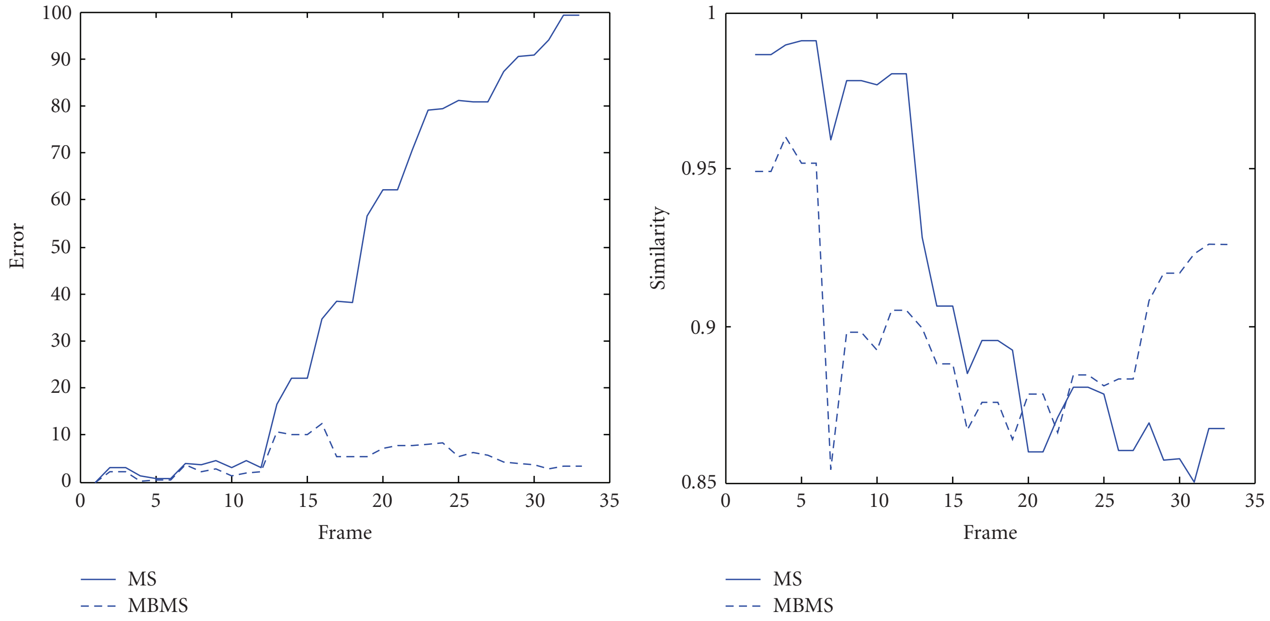
mode that cause the MS seeker to be trapped in its basin of attraction. Through a modification in color histogram value, we can increase the step-size adaptively. If we want to have faster mode seeking so in a trade off balance we may lose some accuracy as well. Recently it has been proposed that we can also add some accuracy with a little more time of computation [44]. We have to increase our computational cost due to larger bandwidth kernel windowing in a frame. As explained, this algorithm was applied to application of an object tracking with faster mode seeking results [36] as shown in Table 1(a) and 1(b).

The most important problem of the proposed method is that the series of bandwidth selection is manual, but we can be looking for some issues to be proposing an automatic selector of bandwidth using some features, but in this paper, we are using manually multi-bandwidth series to track correctly as illustrated in Figure 5.

Figure 6. In this circle tracking scenario, MS fails to track the circle due to getting distracted by the dotted square (i.e., a local mode) but in the same Frame(17) while the MBMS (i.e., the proposed method) is successfully tracking the circle in spite of square distraction in that frame. As can be seen in (Frame 26), MS tracker is unable to track successfully as the circle is moved in front of the square.

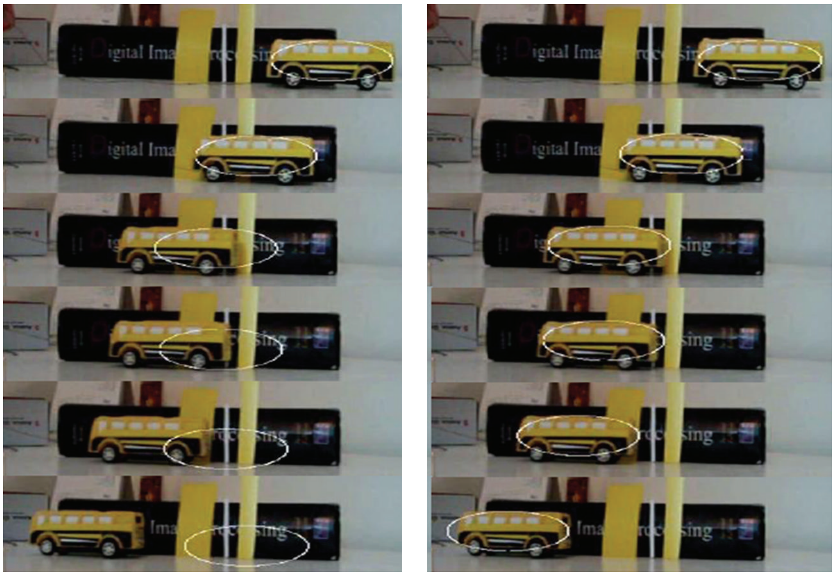
Figure 8. In this Bus tracking scenario, MS fails to track the Bus due to getting distracted by the similar color clutters (i.e., two local modes) while the MBMS (i.e., the proposed method) is successfully tracking the bus in spite of clutter distraction in these frames. As can be seen in MS tracker is unable to track successfully as the bus is moved in front of the same color clutters.

Figure 10. In this hand tracking scenario, MS (a) fails to track the hand due to getting distracted by the face (i.e., a local mode) but in the same frame (140) while the MBMS (i.e., the proposed method) (b) is successfully tracking the



(a) (b)

Figure 7: Left plot: Error from the GT (i.e., Ground Truth is zero state), Right plot: Similarity with Target (GT has 100% similarity). Compared to the real GT (i.e., Ground Truth), we can observe that MBMS successfully performs circle tracking through the entire sequence.



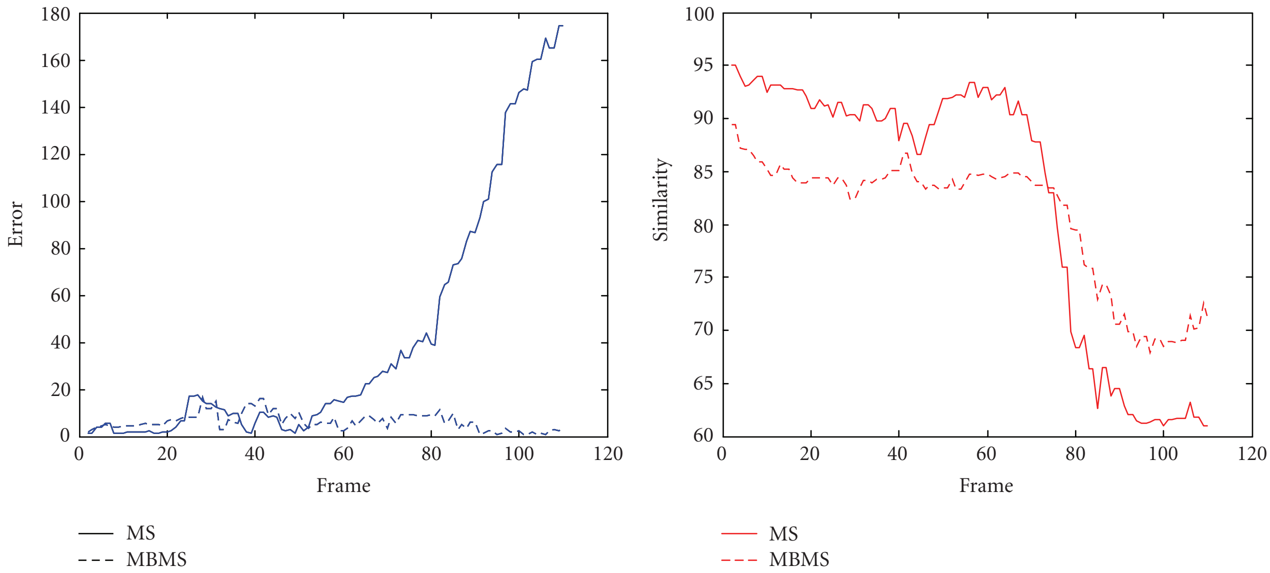
(a) (b)

Figure 8: Frames: 1, 52,79,82,90,110, left column: MS, right column: MBMS.

hand in spite of face distraction in that frame. As can be seen in ((a)-frame 260) MS tracker is unable to track successfully as the hand is moved in front of the face. Compared to the real GT (i.e., Ground Truth) in (c), we can observe that MBMS successfully tracks the hand through the entire sequence.

Definitely, we can observe that MBMS successfully performs better hand tracking through the entire sequence with lower number of iterations than MS.

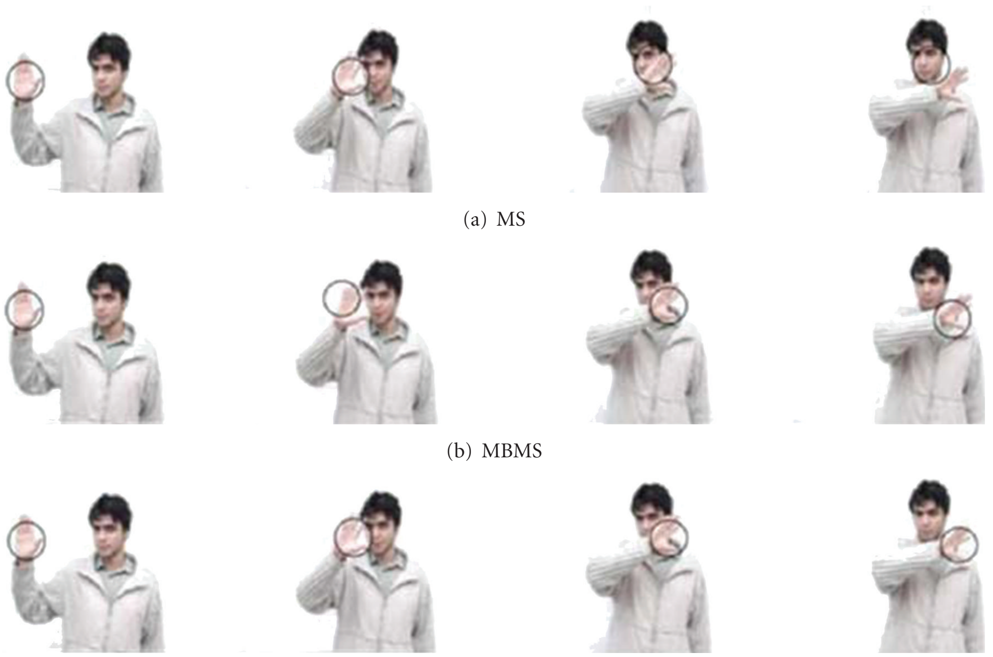
At Table 2-Comparison of Number of Iterations for Convergence for 1D and 2D data set. The Initial Location For Each Run Is Shown in the Second Column.



(a)

(b)

FIGURE 9: Left plot: Error from the GT (i.e., Ground Truth is zero state), Right plot: Similarity with Target (GT has 100% similarity). Compared to the real GT (i.e., Ground Truth), we can observe that MBMS successfully performs bus tracking through the entire sequence.



(a) MS

(b) MBMS

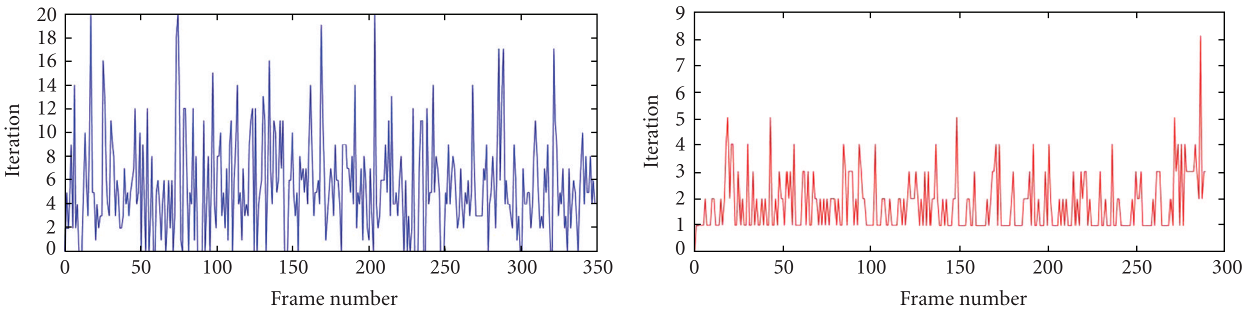
(c) GT

FIGURE 10: Frames: 1, 140, 260, 290.

(1) Data set no. 1 (1D synthetic data). A total of 1000 data points are drown with equal probability from four normals: $N(3,1)$, $N(1,1)$, $N(0,1)$, and $N(-2,1)$

(2) Data set no. 2 (2D synthetic data). A total of 1050 bivariate data points are drown with equal probability from three normals

$$N\left(\begin{bmatrix}-7\\10\end{bmatrix},\begin{bmatrix}5.5 & -4.5\\-4.5 & 5.5\end{bmatrix}\right), \qquad N\left(\begin{bmatrix}0\\12\end{bmatrix},\begin{bmatrix}8.5 & 6.5\\6.5 & 8.5\end{bmatrix}\right),$$
$$N\left(\begin{bmatrix}12\\15\end{bmatrix},\begin{bmatrix}14.4 & -4.5\\-4.5 & 3.6\end{bmatrix}\right). \tag{8}$$



(a) MS

(b) MBMS

Figure 11: Top plot: Iteration number for MS while tracking the hand in Figure 10(a), Bottom plot: Iteration number for MBMS while tracking hand in Figure 10(b).

Table 1

(a) The step size iterations between frame 88 and 89.

Algorithm	Step-size x	Step-size y	Center in Frame 88	Center in Frame 89
Standard MS	0.0040	−0.0006	106	106
Multi-Bandwidth	0.0027	−0.007	58	58

(b) It shows the step size iterations between frame 89 and 90 in seven iteration but different step sizes for standard method and proposed one.

Algorithm	Step-size		Center position of square 90	
	x	y	x	y
Standard	−0.1353	0.0006	104.5116	58.0067
Multi-bandwidth	−0.0889	−0.0192	105.0216	57.7888
Standard	−0.0881	−0.0002	103.5420	58.0048
Multi-bandwidth	−0.0640	−0.0199	104.3180	57.5700
Standard	−0.0515	0.0200	102.9758	58.2248
Multi-bandwidth	−0.0449	−0.0037	103.8240	57.5298
Standard	−0.0754	0.0185	102.1470	58.4286
Multi-bandwidth	−0.0613	−0.0138	103.1495	57.3778
Standard	−0.0082	0.0169	102.0563	58.6143
Multi-bandwidth	−0.0278	−0.0121	102.8439	57.2442
Standard	−0.0197	0.0023	101.8392	58.6395
Multi-bandwidth	−0.0265	−0.0193	102.5524	57.0324
Standard	0.0124	0.0181	101.8392	58.6395
Multi-bandwidth	−0.0101	−0.0107	102.5524	57.0324

Table 2: Comparison of Number of Iterations for Convergence for 1D and 2D data set.

Data set	Initial	Number of iterations	
		MBMS	MS
data set #1	0	33	51
	1	36	77
	3	21	33
data set #2	(−5,20)	22	34
	(−10,16)	21	29
	(20,10)	23	35

6. Conclusion

A new kernel-based object tracking framework is proposed. The contribution is mainly the use of a prior large bandwidth for a priori tracking followed by the estimated tracking. This framework is robust to noise and clutters so that it can escape from many local maxima. This tracking algorithm (i.e., MBMS) can converge faster than does the conventional kernel-based object tracking (i.e., MS). However, there are still some problems, and some weaknesses which are to be later clarified and reparaphrased. Many results can be analyzed theoretically. This paper as a reference can be much helpful for later extension of this work. The experimental results above must have illustrated this approach performance. As shown in above database, it can also be concluded that in rapid motion of an object, large displacement between two adjacent frames occurs which will lead MS to a failure in tracking an object. By means of multi-bandwidth proposal, we can be improving MS in recovering from the failure by incorporating a detector in localization process called multi-bandwidth kernel functionality. In comparison with conventional MS and other techniques like [1] and fast transforming the Gaussian mixture [7, 29], the speed of convergence has increased and the number of iterations has decreased in contrast with an enhancement in each step-size iteration. Object tracking is an important issue in Artificial Intelligence. Its worldwide usages in robotic engineering, Machine Intelligence, Computer Vision, and Human-Computer Interface (HCI) are well-known throughout the world of engineering sciences. In the future, this method can be extended to the more automatic bandwidth selector equipped with several other features to track objects in many varieties of applications inside the industry.

References

[1] B. Georgescu, I. Shimshoni, and P. Meer, "Mean shift based clustering in high dimensions: a texture classification example," in *Proceedings of the 9th IEEE International Conference on Computer Vision*, pp. 456–463, Nice, France, October 2003.

[2] 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.

[3] R. T. Collins, "Mean-shift blob tracking through scale space," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, pp. 234–240, Madison, Wis, USA, June 2003.

[4] A. Elgammal, R. Duraiswami, and L. S. Davis, "Probabilistic tracking in joint feature-spatial spaces," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, pp. 781–788, Madison, Wis, USA, June 2003.

[5] G. D. Hager, M. Dewan, and C. V. Stewart, "Multiple kernel tracking with SSD," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, pp. 790–797, Washington, DC, USA, July 2004.

[6] H. Wang and D. Suter, "MDPE: a very robust estimator for model fitting and range image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 139–166, 2004.

[7] C. Yang, R. Duraiswami, and L. Davis, "Efficient mean-shift tracking via a new similarity measure," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 176–183, San Diego, Calif, USA, June 2005.

[8] K. Fukunaga and L. D. Hostetler, "The estimation of the gradient of a density function, with application in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, no. 1, pp. 32–40, 1975.

[9] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, 1995.

[10] 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.

[11] M. Fashing and C. Tomasi, "Mean shift is a bound optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 471–474, 2005.

[12] S. K. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Transactions on Image Processing*, vol. 13, no. 11, pp. 1491–1506, 2004.

[13] É. Arnaud, É. Mémin, and B. Cernuschi-Frías, "Conditional filters for image sequence-based tracking—application to point tracking," *IEEE Transactions on Image Processing*, vol. 14, no. 1, pp. 63–79, 2005.

[14] Z. Zivkovic and B. Kröse, "An EM-like algorithm for color-histogram-based object tracking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, pp. 798–803, Washington, DC, USA, July 2004.

[15] Z. Fan, Y. Wu, and M. Yang, "Multiple collaborative kernel tracking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 502–509, San Diego, Calif, USA, June 2005.

[16] P. van Laarhoven and E. Aarts, *Simulated Annealing: Theory and Applications*, Springer, New York, NY, USA, 1987.

[17] R. M. Neal, "Annealed importance sampling," *Statistics and Computing*, vol. 11, no. 2, pp. 125–139, 2001.

[18] J. Deutscher and I. Reid, "Articulated body motion capture by stochastic search," *International Journal of Computer Vision*, vol. 61, no. 2, pp. 185–205, 2005.

[19] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.

[20] K. Zhang, M. Tang, and J. T. Kwok, "Applying neighborhood consistency for fast clustering and kernel density estimation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 1001–1007, San Diego, Calif, USA, June 2005.

[21] C. Yang, R. Duraiswami, D. DeMenthon, and L. Davis, "Mean-shift analysis using quasi-newton methods," in *Proceedings of the International Conference on Image Processing (ICIP '03)*, pp. 447–450, Barcelona, Spain, September 2003.

[22] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixtures," *Neural Computation*, vol. 8, no. 1, pp. 129–151, 1996.

[23] L. E. Ortiz and L. P. Kaelbling, "Accelerating EM: an empirical study," in *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence*, pp. 512–521, Stockholm, Sweden, 1999.

[24] R. Salakhutdinov and S. Roweis, "Adaptive overrelaxed bound optimization methods," in *Proceedings of the 20th International Conference on Machine Learning*, pp. 664–671, Washington, DC, USA, August 2003.

[25] C. Shen, M. J. Brooks, and A. van den Hengel, "Fast global kernel density mode seeking with application to localisation and tracking," in *Proceedings IEEE International Conference on Computer Vision*, vol. 2, pp. 1516–1523, Beijing, China, October 2005.

[26] D. Comaniciu, "An algorithm for data-driven bandwidth selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 281–288, 2003.

[27] M. C. Jones, J. S. Marron, and S. J. Sheather, "A brief survey of bandwidth selection for density estimation," *Journal of the*

American Statistical Association, vol. 91, no. 433, pp. 401–407, 1996.

[28] W. Härdle, M. Müller, S. Sperlich, and A. Werwatz, *Nonparametric and Semiparametric Models*, Springer Series in Statistics, Springer, New York, NY, USA, 2004.

[29] C. Yang, R. Duraiswami, N. A. Gumerov, and L. Davis, "Improved fast gauss transform and efficient kernel density estimation," in *Proceedings of the 9th IEEE International Conference on Computer Vision*, pp. 464–471, Nice, France, October 2003.

[30] P. Hall, M. C. Minnotte, and C. Zhang, "Bump hunting with non-Gaussian kernels," *Annals of Statistics*, vol. 32, no. 5, pp. 2124–2141, 2004.

[31] P. Chaudhuri and J. S. Marron, "Scale space view of curve estimation," *Annals of Statistics*, vol. 28, no. 2, pp. 408–428, 2000.

[32] T. Lindeberg, *Scale-Space Theory in Computer Vision*, Kluwer, Norwell, Mass, USA, 1994.

[33] A. Blake and A. Zisserman, *Visual Reconstruction*, MIT Press, Cambridge, Mass, USA, 1987.

[34] D. Comaniciu, "Nonparametric information fusion for motion estimation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, vol. 1, pp. 59–66, Madison, Wis, USA, June 2003.

[35] M. Carreira-Perpiñán and C. Williams, "On the number of modes of a Gaussian mixture," Tech. Rep. EDI-INF-RR-0159, School Informatics, University of Edinburgh, Edinburgh, UK, February 2003, http://www.inf.ed.ac.uk/publications/report/0159.html.

[36] C. Shen and M. J. Brooks, "Adaptive over-relaxed mean shift," in *Proceedings of the 8th International Symposium on Signal Processing and Its Applications*, Sydney, Australia, August 2005.

[37] S. Hettich and S. D. Bay, "The UCI KDD archive," Tech. Rep., Department of Information and Computer Science, University of California, Irvine, Calif, USA, 1999, http://kdd.ics.uci.edu/.

[38] T.-L. Liu and H.-T. Chen, "Real-time tracking using trust-region methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 3, pp. 397–402, 2004.

[39] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, no. 1–3, pp. 503–528, 1989.

[40] T. Kailath, "The divergence and Bhattacharyya distance measures in signal selection," *IEEE Transactions on Communication Technology*, vol. 15, no. 1, pp. 52–60, 1967.

[41] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[42] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little, and D. G. Lowe, "A boosted particle filter: multitarget detection and tracking," in *Proceedings of the 8th European Conference on Computer Vision*, vol. 1, pp. 28–39, Prague, Czech Republic, May 2004.

[43] H. Wang and D. Suter, "False-peaks-avoiding mean shift method for unsupervised peak-valley sliding image segmentation," in *Proceedings of the 7th International Conference on Digital Image Computing: Techniques and Applications (DICTA '03)*, pp. 581–590, Sydney, Australia, December 2003.

[44] A. T. Kalai and S. Vempala, "Simulated annealing for convex optimization," *Mathematics of Operations Research*, vol. 31, no. 2, pp. 253–266, 2006.