International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Improved Mean Shift for Efficient Visual Tracking

Hui Yin, Yongfeng Cao[1], Hong Sun, Wen Yang

Lab of Signal Processing, School of Electronic Information, Wuhan University, Wuhan 430079, China

Abstract: Two new methods based on mean shift iterations for real-time tracking of non-rigid objects are proposed. The first method called MSIⅠ (Mean shift improved methodⅠ) adds a perturbation step in the conventional procedure. In this way, the mean shift more probably avoids stopping at the local plateau. The second method called MSI Ⅱextends into four mean shift procedures to prevent tracker from converging at the local maximum point. Experiments on face tracking show the advantages and limitations of the new approaches.

Keywords: Mean shift; Bhattacharyya coefficient; visual tracking.

1. Introduction

Mean shift procedure was proposed in 1975 by Fukunaga and Hostetler [1] and developed by Cheng's paper [2]. It has been introduced recently in several computer vision papers for tracking and segmentation applications [3-6]. It is a simple and fast adaptive tracking procedure that climbs density gradients to find the peak of probability distributions where the target appears. One of the biggest problems in motion-based tracking using mean shift method is to lose the object due to rapid movements. If the rapid moving target is out of the tracker's searching region, the mean shift tracker can converge to a local maximum on the background that has a similar color distribution as the target model. In this situation the tracker has no chance to recover.

This paper applies two new approaches to solve this problem. The first one gives the convergent location a random perturbation and reapplies the mean shift procedure until it converges again. This improved method has been used to clustering [5] and we employ it to prevent the tracker from stopping in the local plateau. The second one extends the convergent location of current frame into four new candidates. They locate at the four directions such as up, down, left and right respectively. We run the mean shift procedure at the new positions and regard the one which has the biggest similarity as the terminal target position. Experiments show that the two methods are robust to rapid movement, lighting variations, partial occlusions, and rotation.

The paper is organized as follows. Section 2 presents the mean shift property. Section 3 introduces the conventional mean shift tracking algorithm. In section 4 we will develop our improved new methods. Experiments and comparisons are given in section 5 and conclusion will be drawn in section 6.

2. Mean Shift analysis

Given a set $\{x_i\}_{i=1\ldots n}$ of n points in the d dimensional space R^d, the multivariate kernel density estimate with kernel $K(x)$ and window radius(bandwidth) h, computed in the point x is given by

$$\hat{f}(x)=\frac{1}{nh^d}\sum_{i=1}^{n}K\left(\frac{x-x_i}{h}\right) \qquad (1)$$

The profile of a kernel K is defined as a function $k:[0,\infty)\rightarrow R$ such that $K(x)=k(\|x\|^2)$. Employing the profile notation we can write the density estimate (1) as

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

We denote

$$g(x) = -k'(x) \quad (3)$$

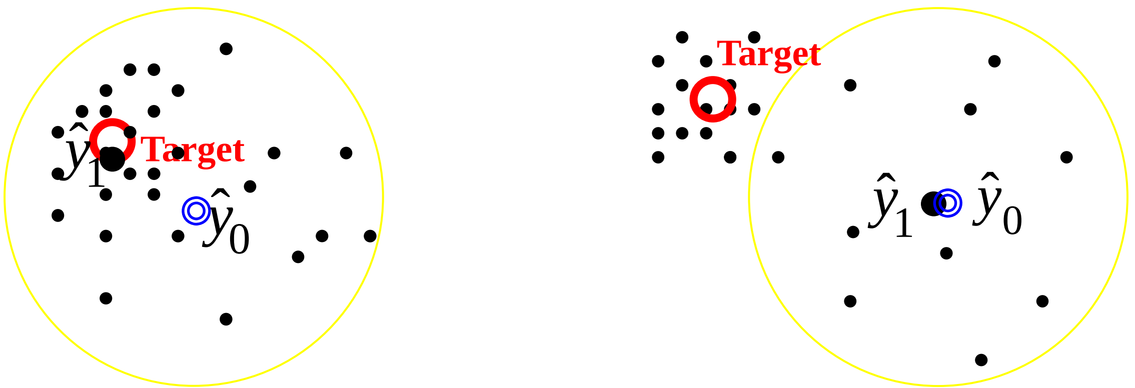
as the profile of kernel G. Then, by taking the estimate of the density gradient as the gradient of the density estimate we have the sample mean shift vector (details see [8])

$$M_{h,G}(x) = \frac{\sum_{i=1}^{n} x_i g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} - x \quad (4)$$

It follows that

$$M_{h,G}(x) = \frac{h^2}{2/C}\frac{\hat{\nabla} f_K(x)}{\hat{f}_G(x)} \quad (5)$$

where $\hat{\nabla} f_K(x)$ is the estimate of the density gradient and $\hat{f}_G(x)$ is the density estimate at x computed with kernel G. Expression (5)shows that the sample mean shift vector obtained with kernel G is an estimate of the normalized density gradient obtained with kernel K. The mean shift procedure is defined recursively by computing the mean shift vector $M_{h,G}(x)$ and translating the center of kernel G by $M_{h,G}(x)$.



(a) (b)

Figure.1 Conventional mean shift tracker: the red bold circle illustrates the target position which is the density center of the small black points in the whole space; the blue homocentric circle represents $\hat{y}_0$ which is the tracker's position in previous frame; the black solid ball stands for $\hat{y}_1$ which is the tracker's position in current frame (a)when target does not move rapidly between frames (b)when target does move rapidly enough between frames.

3. Tracking Algorithm

3.1 Target Model

Let $\{x_i^*\}_{i=1...n}$ be the pixel locations of the target model, centered at 0. We define a function $b: R^2 \to \{1...m\}$ which associates to the pixel at location x_i^*. The index $b(x_i^*)$ of the histogram bin corresponds to the color of that pixel. The probability of the color u in the target model is derived by employing a convex and monotonic decreasing kernel profile k which assigns a smaller weight to the locations that are farther from the center of the target. The weighting increases the robustness of the estimation since the peripheral pixels are the least reliable, being often affected by occlusion or background. The radius of the kernel profile is taken equal to one, by assuming that the generic coordinates x and y are normalized with h_x and h_y, respectively. Hence, we can write

$$\hat{q}_u = C\sum_{i=1}^{n} k\left(\left\|x_i^*\right\|^2\right)\delta[b(x_i^*)-u] \quad (6)$$

where δ is the Kronecker delta function. The normalization constant C makes $\sum_{u=1}^{m}\hat{q}_u = 1$.

3.2 Target Candidates

Let $\{x_i\}_{i=1..n_h}$ be the pixel locations of the target candidate, centered at y in the current frame. Using the same kernel profile k, but with radius h, the probability of the color u in the target candidate is given by

$$\hat{p}_u(y) = C_h k\left(\left\|\frac{y-x_i}{h}\right\|^2\right)\delta[b(x_i)-u] \quad (7)$$

where C_h is the normalization constant which makes $\sum_{u=1}^{m}\hat{p}_u = 1$. The radius of the kernel profile determines the number of pixels (i.e. the scale) of the target candidate.

3.3 Distance Minimization

We take the Bhattacharyya coefficient as the similarity measurement. The larger the Bhattacharyya coefficient is, the more similar the distributions are. Considering discrete densities such as our color histograms $\hat{p}(y)$ and $\hat{q}$, the coefficient is defined as

$$\hat{\rho}[y] \equiv \rho[\hat{p}(y),\hat{q}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(y)\hat{q}_u} \qquad (8)$$

Using (8) the distance between two distributions can be defined as

$$d(y) = \sqrt{1-\rho[\hat{p}(y),\hat{q}]} \qquad (9)$$

The most probable location y of the target in the current frame is obtained by minimizing the distance (9), which is equivalent to maximizing the Bhattacharyya coefficient $\hat{\rho}[y]$. The search for the new target location in the current frame starts at the estimated location $\hat{y}_0$ of the target in the previous frame. Thus, the color probabilities $\{\hat{p}_u(\hat{y}_0)\}_{u=1\ldots m}$ of the target candidate at location $\hat{y}_0$ in the current frame have to be computed first. Using Taylor expansion around the value $\hat{p}_u(\hat{y}_0)$, the Bhattacharyya coefficient (8) is approximated as (after some manipulations)

$$\rho[\hat{p}(y),\hat{q}] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{y}_0)\hat{q}_u} + \frac{1}{2}\sum_{u=1}^{m}\hat{p}_u(y)\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{y}_0)}} \qquad (10)$$

Introducing (7) in (10) we obtain

$$\rho[\hat{p}(y),\hat{q}] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{y}_0)\hat{q}_u} + \frac{C_h}{2}\sum_{i=1}^{n_h}\omega_i k\left(\left\|\frac{y-x_i}{h}\right\|^2\right) \qquad (11)$$

where

$$\omega_i = \sum_{u=1}^{m}\delta[b(x_i)-u]\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{y}_0)}} \qquad (12)$$

Thus, to minimize the distance (9), the second term in equation (11) has to be maximized, the first term being independent of y. The second term represents the density estimate computed with kernel profile k at y in the current frame, with the data being weighted by ω_i (12). The maximization can be efficiently achieved based on the mean shift iterations, using the following algorithm.

3.4 Conventional mean shift algorithm

Given the distribution $\{\hat{q}_u\}_{u=1\ldots m}$ of the target model and the estimated location $\hat{y}_0$ of the target in the previous frame:

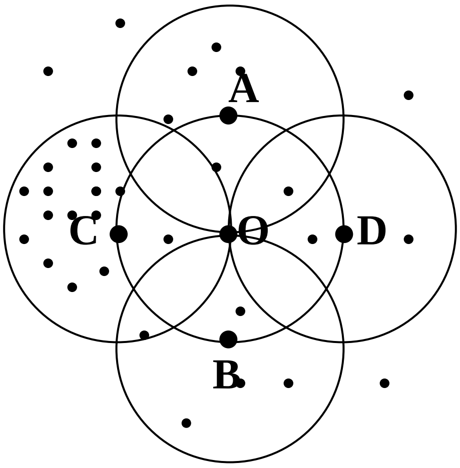


Figure.2 MSIⅡ: Extend the convergent point O into four new points locate at A, B, C and D

1. Initialize the location of the target in the current frame with $\hat{y}_0$, compute the distribution $\{\hat{p}_u(\hat{y}_0)\}_{u=1\ldots m}$, and evaluate $\rho[\hat{p}(\hat{y}_0),\hat{q}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{y}_0)\hat{q}_u}$.

2. Derive the weights $\{\omega_i\}_{i=1\ldots n_h}$ according to (12).

3. Based on the mean shift vector, derive the new location of the target

$$\hat{y}_1 = \frac{\sum_{i=1}^{n_h} x_i\omega_i g\left(\left\|\frac{\hat{y}_0-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h}\omega_i g\left(\left\|\frac{\hat{y}_0-x_i}{h}\right\|^2\right)} \qquad (13).$$

Update $\{\hat{p}_u(\hat{y}_1)\}_{u=1\ldots m}$, and evaluate $\rho[\hat{p}(\hat{y}_1),\hat{q}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{y}_1)\hat{q}_u}$.

4. *While* $\rho[\hat{p}(\hat{y}_1),\hat{q}] < \rho[\hat{p}(\hat{y}_0),\hat{q}]$

   Do $\hat{y}_1 \leftarrow \frac{1}{2}(\hat{y}_0 + \hat{y}_1)$.

5. *If* $\|\hat{y}_1 - \hat{y}_0\| < \varepsilon$ *Stop. Otherwise Set* $\hat{y}_0 \leftarrow \hat{y}_1$, and go to Step 1.[6]

4. Two improved methods

The conventional mean shift algorithm has been used successfully in [6]. But one of the biggest problems in motion-based tracking is to lose the object due to rapid movements [7].Experiments show that when the target moves rapidly, the mean shift tracker possibly misses it. We can explain this situation as: If the old and new regions centered at the detected points do not overlap, the mean shift tracker can converge to a local maximum on the background that has a similar color distribution as the target model. In this situation, the tracker has no chance to recover. See Figure.1. Figure.1 (a) shows the situation when the target does not move rapidly and the old and new regions overlap. Therefore, the tracker can achieve the best location. Figure.1 (b) shows the target moves rapidly enough and it has moved out of the search region. In this case the tracker can only converge at the local mode in the background which is similar to the model histogram.

We apply two new methods to handle this problem.

4.1 MSI Ⅰ

To handle the problem of stopping at the local plateau, we employ a random perturbation to the convergent point and call it MSIⅠ(Mean shift improved method Ⅰ). In fact, this idea is self-contained in the mean shift theory [8]. And it has been used in the application of clustering [5]. But it has not been included in the convention mean shift tracking theory. This is the first time we formally employ it as an important iteration step in the procedure of tracking. After perturbing, we reapply the mean shift procedure in the new position, and get a new Bhattacharyya similarity coefficient. Choose the bigger one to get a better tracking result. The detail steps of MSIⅠare as follows:

1-5 steps are the same as section 3.4.

6. $\hat{y}_2{}' = \hat{y}_1 + v_y$, where v_y is a random perturbed vector.

$\hat{y}_0 \leftarrow \hat{y}_2{}'$, go to step1- step5 and get $\hat{y}_2$.

7. $\hat{y}_1 = \arg_y \max(\rho[\hat{p}(\hat{y}_1),\hat{q}],\rho[\hat{p}(\hat{y}_2),\hat{q}])$.

4.2 MSIⅡ

To avoid missing the target farther, we present the following method called the MSIⅡ. Firstly, we run the mean shift algorithm till it converges at y_1 . Secondly, we extend the location to four new initial points which are located at $0^\circ, 90^\circ, 180^\circ, 270^\circ$ directions and a radium away from y_1 respectively. Reapply the mean shift procedure at each new point till it converges. Finally, choose the one which has the biggest Bhattacharyya coefficient as the location of the target in the current frame. Details see Figure.2.

The procedure is as follows:

Step 1-5 are the same as section 3.4.

```
6.  $\hat{y}_2(\mathbf{1}) = \hat{y}_1 + \Delta_x , \hat{y}_2(\mathbf{2}) = \hat{y}_1 - \Delta_x$
    $\hat{y}_2(\mathbf{3}) = \hat{y}_1 + \Delta_y , \hat{y}_2(\mathbf{4}) = \hat{y}_1 - \Delta_y$
    for  $i = 1:4$
    $\hat{y}_2(i) = \hat{y}_1 + \Delta_y , \hat{y}_2(\mathbf{4}) = \hat{y}_1 - \Delta_y$ go to step 1-5
        get the convergent point  $\hat{y}_2(i)$
        and  $\rho[p(\hat{y}_2(\mathbf{i})),\hat{q}]$;
    end
7.  $\hat{y}_1 = \arg_y \max(\rho[p(\hat{y}_1),\hat{q}\ ],\rho[p(\hat{y}_2(i)),\hat{q}], i = 1...4);$
```

5. Experiments

The improved and proposed methods have been applied to the task of tracking face in three different sequences. The Epanechnikov profile has been used in all the experiments in this paper. The target color histogram model has been derived by taking 1D histogram from the H (hue) channel in HSV space with 361 bins. The first sequence has 100 frames of 240×352pixels each and the initial size of the target model is 73×59.

The tracking results are presented in Figure.3 The conventional mean shift algorithm result is signed by white circle, MSIⅠgreen circle and MSI Ⅱred circle. The improved two trackers are proved to be robust to rapid movement. For example, in frame 63 and 67, the conventional mean shift misses the target due to the person’s rapid jump while the two improved trackers still keep tracking well. And MSI Ⅱis more accurate than MSIⅠ. Furthermore, at the end of the sequence the light is turned off so the room gets darker, but as we use the H channel which is robust to lighting, this does not influence the performance at all.

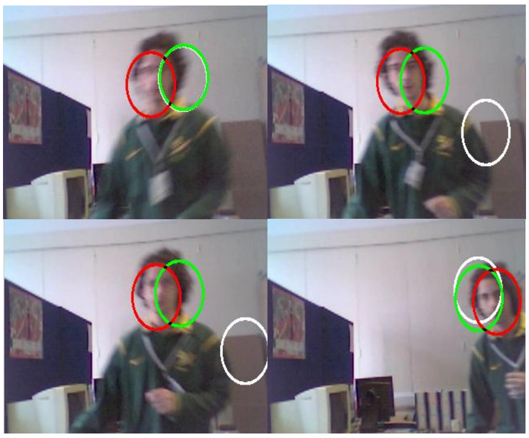


Figure.3 Rapid movement sequence: The frames 58, 63, 67 and 87are shown (left-right, top-down). The white, green and red circles show the tracking result of conventional mean shift algorithm, MSIⅠ, and MSI Ⅱrespectively.

Figure.4 shows the Bhattacharyya distance of the three trackers in rapid movement sequence. From it we can see the improved methods outperform the conventional method. It means that MSIⅠ, and MSIⅡcan find a higher mode of the density function. Figure.5 shows the number of iterations for the rapid movement sequence. Obviously, the improved methods need more iteration times so they calculate longer. MSIⅡspends the most response time of all to get the best tracking performance.

The tracking result of another sequence of rapid rotation is shown in Figure.6. It has 121 frames. From the sequence we can see MSIⅡis more robust to rapid rotation than the other two trackers. Even when the target disappears from the camera for a short moment, MSIⅡcan recover quickly as has been shown in frame 101 and 102.

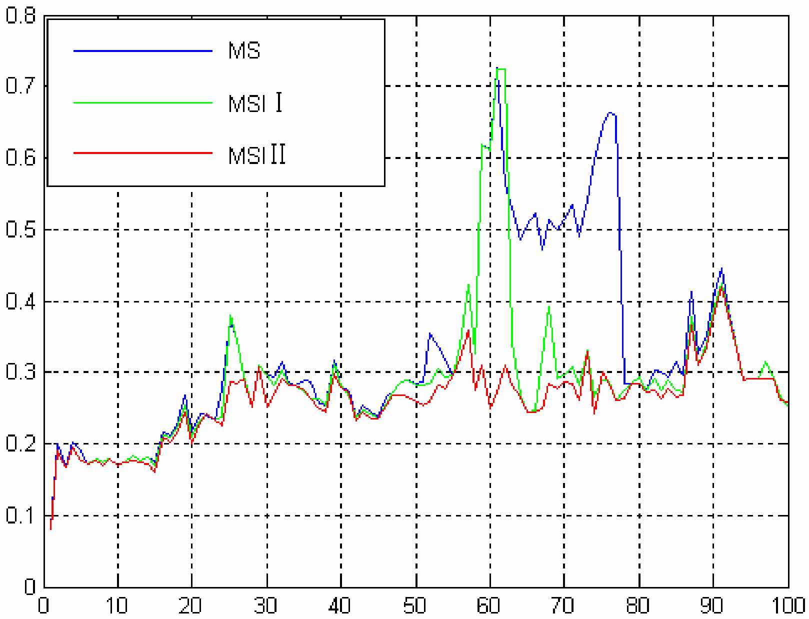


Figure.4 The Bhattacharyya distance for rapid movement sequence. The blue, green and red lines correspond to the conventional mean shift, MSIⅠ, and MSIⅡrespectively.

Figure.7 corresponds to the Bhattacharyya distance of the rapid rotation sequence. From Figure.7 we can conclude that MSIⅡalways gets the smallest distance in the whole procedure, while MSIⅠimproves only a little.

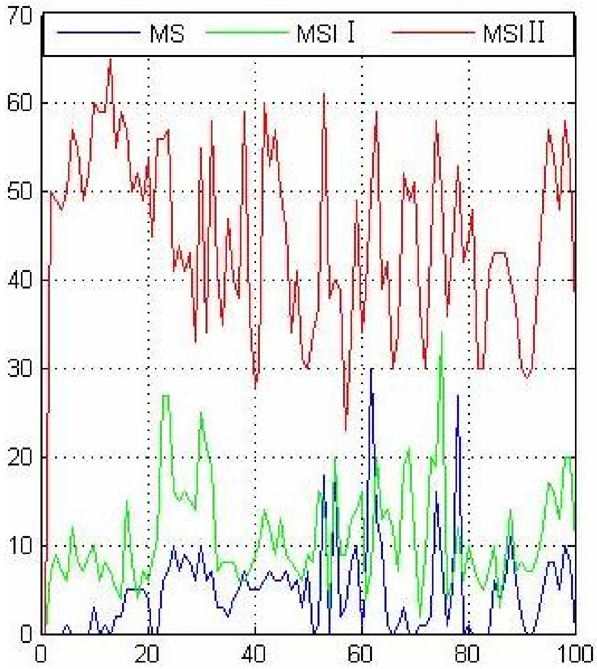


Figure.5 The number of iterations for rapid movement sequence. The blue, green and red lines correspond to the conventional mean shift, MSIⅠ, and MSIⅡrespectively.

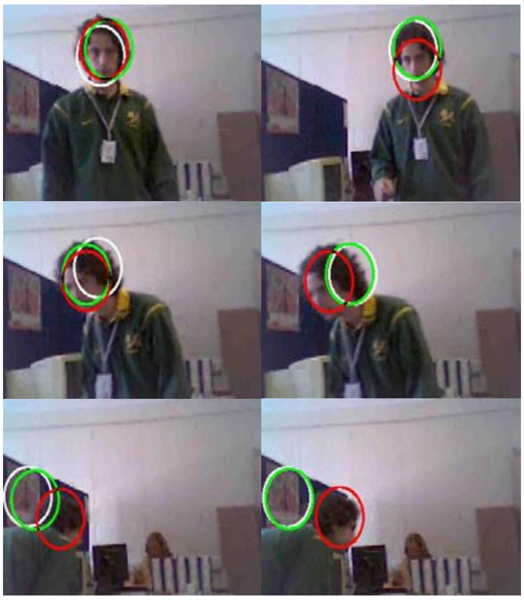


Figure.6 Rapid rotation sequence: The frames 62,68,87,88,101 and 102are shown (left-right, top-down). The white, green and red circles show the tracking result of conventional mean shift algorithm, MSIⅠ, and MSI Ⅱrespectively.

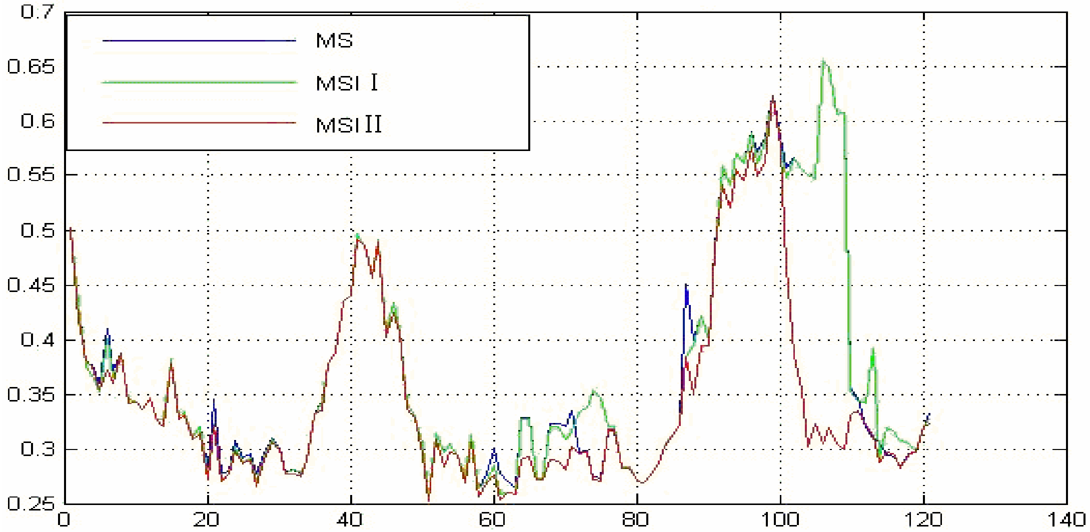


Figure.7 The Bhattacharyya distance for rapid rotation sequence. The blue, green and red lines correspond to the conventional mean shift, MSI Ⅰ, and MSI Ⅱ respectively.

Figure.8 shows the tracking result when occlusion happens. The two new trackers are more robust to occlusion by the similar color like hand.

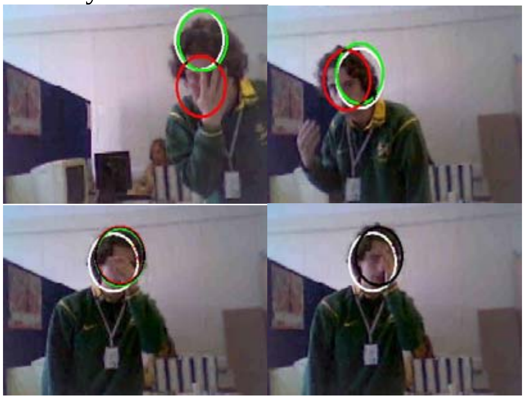


Figure.8 Occlusion sequence: The frames 38,103,121 and 123are shown (left-right, top-down). The white, green and red circles show the tracking result of conventional mean shift algorithm, MSI Ⅰ, and MSI Ⅱ respectively.

Figure.9 shows the Bhattacharyya distance of the whole procedure. And MSI Ⅱ still performs the best of all.

6. Discussion

By employing perturbation step, we get MSI Ⅰ. This is an existing method and we just reuse it. By extending one point into four points to perform the mean shift procedure, we propose MSI Ⅱ. Various test sequences show the superior tracking performance, obtained by MSI Ⅱ. However, this depends on sacrificing the computational time. MSI Ⅰ keeps its efficiency but its improvement is not as obvious as MSI Ⅱ. Therefore, if the efficiency is not required strictly, we use MSI Ⅱ to get better

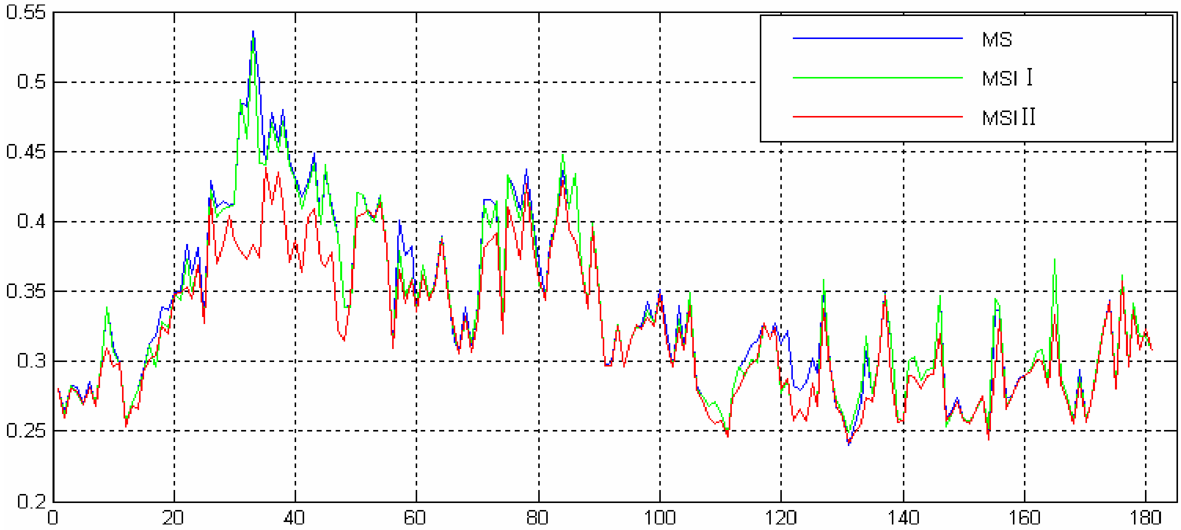


Figure.9 The Bhattacharyya distance for occlusion sequence. The blue, green and red lines correspond to the conventional mean shift, MSI Ⅰ, and MSI Ⅱ respectively.

performance; if efficiency is the dominate part, we employ MSIⅠ.

All the methods find the target model with the similarity being expressed by a metric based on the Bhattacharyya coefficient. MSIⅡcan find a higher mode so it is more sensitive to color distribution than MSIⅠ. Meanwhile, it has to ignore more space information among frames than MSIⅠ. In fact, the space information is the prior acknowledge preventing tracker from going wrong. For example, in Figure.10 the target's face is occluded by another person's face with similar color distribution for a while. As a result , at the end of the sequence only MSIⅠtracks the right target while the other two miss. However, the distance curves in Figure.11 show that MSIⅠhas the biggest distance of all, although it tracks right. This experiment tells us that we should employ more target feature like texture and shape together with color distribution to track accurately in complexity environments. And our research interests will focus on it in the future.

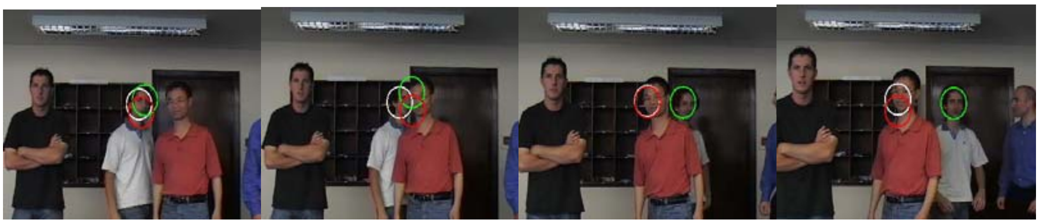


Figure.10 Multi-face sequence: The frames 65, 70, 84 and 96 are shown (left-right). The white, green and red circles show the tracking result of conventional mean shift algorithm, MSIⅠ, and MSI Ⅱrespectively.

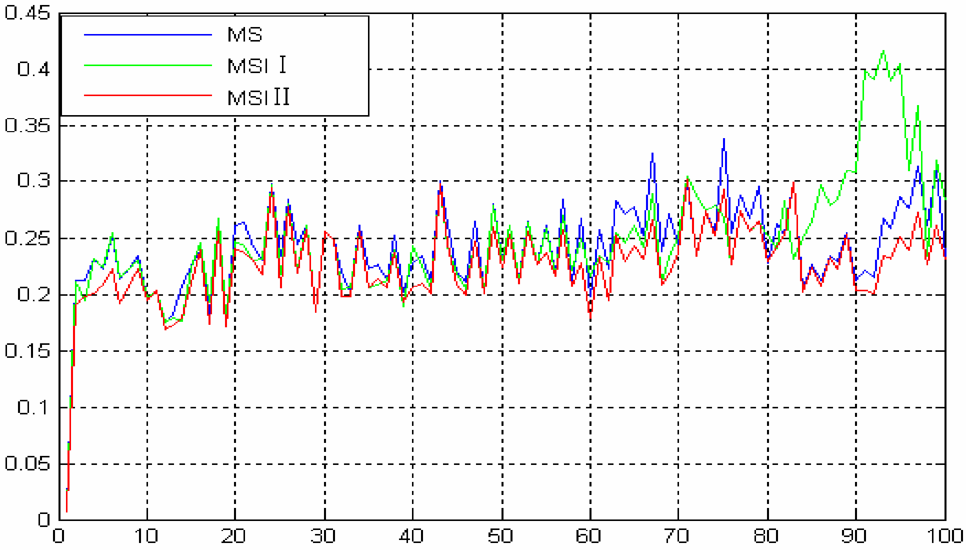


Figure.11 The Bhattacharyya distance for multi-face sequence. The blue, green and red lines correspond to the conventional mean shift, MSIⅠ, and MSIⅡrespectively.

Acknowledgement

The original video sequences are downloaded from http://www.elec.qmul.ac.uk/staffinfo/andrea/PHD-MT.html. The authors would like to thank the research institute for sharing their database.

We also would like to thank for the support from Open Fund of State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing of China and The National High Technology Research and Development Program of China（No.2007AA12Z155）.

References

[1]K.Fukunaga and L.D.Hostetler, "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition," *IEEE Trans. on Information Theory*, Vol.21, pp.32-40, 1975.

[2]Y. Cheng, "Mean Shift, Mode Seeking, and Clustering," *IEEE Trans .on Pattern Analysis and Machine Intelligence*, Vol.17, No.8, pp.790-799, Aug.1995.

[3]Dorin Comaniciu and Peter Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.24, No.5, May 2002.

[4]Dorin Comaniciu and Visvanathan Ramesh, "Kernel-Based Object Tracking", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.25,No.5,May 2003.

[5]Dorin Comaniciu and Peter Meer, "Distribution Free

Decomposition of Multivariate Data", *Pattern Analysis and ApplicationVol.*,2,pp.22–30,1999.

[6]Dorin Comaniciu and Visvanathan Ramesh, "Real-Time Tracking of Non-Rigid Objects using Mean Shift", *IEEE CVPR 2000*.

[7]Katja Nummiaro and Esther Koller-Meier, "A Color-based Particle Filter", *International Workshop on Generative-Model-Based. Vision GMBV'02*, in conjunction with ECCV'02 2002, pp.53~60.

[8]Yaron Ukrainitz and Bernard Sarel, http://www.wisdom.weizmann.ac.il/~deniss/vision_spring04/files/mean_shift/mean_shift.ppt