2006 IEEE International Conference on
Systems, Man, and Cybernetics
October 8-11, 2006, Taipei, Taiwan

Robust Mean Shift Tracking Based on Multi-Cue Integration

Hong Liu, Ze Yu and Hongbin Zha

Abstract—**Color-based Mean Shift has been addressed as an effective and fast algorithm for tracking color blobs. This deterministic searching method suffers from low saturation color object, color clutter in backgrounds and complete occlusion for several frames. This paper proposes a direct motion-color integration method to solve the low saturation color problem and the color background clutter problem. Based on the direct cue integration, an occlusion handler that is able to deal with long term full occlusion is proposed to solve the complete occlusion problem as well. Moreover, motivated by the idea of tuning weight of each cue according to its performance, a method of adaptive multi-cue integration based Mean Shift is proposed. Weights of each cue are adjusted according to a quality function, which is used to evaluate the performance of each cue in the adaptive integration scheme. Extensive experiments show that this method can adapt the weight of individual cue efficiently, and increase the robustness of tracking in various conditions.**

I. Introduction

Tracking objects in complex environments is a quite challenging task in smart surveillance fields. An ideal tracking algorithm should deal with many problems, such as varying illuminations, background clutter, occlusion, etc. There are two trends in the computer vision tracking community. One is to develop more inherently robust algorithms and the other is to employ multiple cues to enhance tracking robustness at present. To increase the robustness and generality of tracking, it is of our interest to employ multiple cues under a robust tracking framework.

Tracking algorithms fall into two categories. The first category is probabilistic methods. These methods view the tracking algorithm as a state solving problem under the Bayesian framework, model uncertainty and propagate the conditional densities through the tracking process. The representative methods are Kalman Filter and its derivatives, CONDENSATION[1], Particle Filter[2], Monte Carlo tracking[3], etc. The second category is Deterministic methods. These methods compare a model with current frame and find out the most probable region. Mean Shift[4][5] and Trust Region[6] fall into this category. The deterministic methods can hardly handle complete occlusion very well, as tracking is initialized to the previous tracking results. However, they are usually more accurate than probabilistic multi-hypothesis tracking, like Particle Filter.

Mean Shift is a non-parametric method for climbing the density gradient to find the peak of a distribution, which belongs to the deterministic methods category. Mean Shift is fast convergent and robust to small distractors in distributions. Mean Shift is first applied to color tracking as Continuous Adaptive Mean Shift (CAMSHIFT) by Bradski [4] and Commaniciu et. al.[5] respectively, using color histogram to model the object's color. The two algorithms differ in the calculations of color distribution and kernel scale. Recent literatures [6][7][8] extended the Mean Shift algorithm, focusing on solving the window scale problem. How to solve the background color clutter and the complete occlusion problem has not been addressed in these literatures.

To develop a more general vision tracking system, using a single cue is not sufficient because of changing environments in real world. Various complementary features can be combined to get more robust tracking results. Some researchers are focusing on establishing a multi-cue integration mechanism under the probabilistic framework, including Dynamic Bayesian Network [9], Monte Carlo method [10], Particle Filter [11], etc.

Up to now, most of the Mean Shift related literatures employ only single color probability distribution. This made tracking results vulnerable to complex conditions, such as, similarly colored backgrounds, tracking low saturation object, changing appearance caused by object rotation, etc. There are four reasons to integrate motion cue into Mean Shift. First, motion cue can enhance the robustness of the original color-base Mean Shift methods in various conditions. Second, motion detection results are regarded as a motion distribution map and can be integrated with color distribution naturally. Third, as Mean Shift is robust to small distractors, we can employ preliminary motion detection algorithm, which saves computation. Last, Mean Shift is a fast mode seeking algorithm, and it saves computational resources for cue-integration methods.

In this paper, probability distributions from motion and color cues are integrated under the Mean Shift framework. Motion cue can enhance the robustness of the original color-base Mean Shift methods in various conditions like tracking object with low saturation color and background color clutter. Based on this, an occlusion handler is proposed to solve the full occlusion problem in the deterministic Mean Shift as well. Experiments show that it can handle long time occlusion which can not be handled by Particle Filter.

Motivated by the idea of tuning weight of each cue according to its reliability, a method of adaptive multi-cue integration based Mean Shift is proposed as well. The

Manuscript received on March 13, 2006. This work is supported by National Natural Science Foundation of China (NSFC, Project No. 60175025).

Hong Liu is with the National Key Lab on Machine perception, Peking University, Beijing, 100871 China. (e-mail: liuhong@cis.pku.edu.cn).

Ze Yu is with the National Key Lab on Machine perception, Peking University, Beijing, 100871 China. (e-mail: angeloyz@gmail.com).

Hongbin Zha is with the National Key Lab on Machine perception, Peking University, Beijing, 100871 China. (e-mail: zha@cis.pku.edu.cn).

1-4244-0100-3/06/$20.00 ©2006 IEEE

difference between [11] and our work is that we employ the adaptive integration mechanism under the framework of Mean Shift, with a new quality function suitable for blob tracking.

The rest of paper is organized as follows: section II brings out the direct color-motion integration method and the occlusion handler. Section III illustrates the strategy of adaptive multi-cue integration. Experiment results and conclusions are presented in Section IV and Section V, respectively.

II. Integrating Multiple Cues

A. Deficiency with Color-based Mean Shift

To use Mean Shift iterations, a probabilistic distribution map indicating the tracked object should be calculated first. A color probabilistic map is calculated by histogram back projection: first, the color histogram of the object's color is calculated and stored in a look-up table. When each new frame comes in, the table is looked up for each pixel's color, and a probability value is assigned to each pixel. Hence, a probabilistic distribution map is obtained. Mean Shift procedure can be employed to find the nearby dominant distribution peak afterwards.

In color-based tracking, to gain robustness against illumination variation, RGB video images are first converted into HSV format. Suppose that the color model is a 1D color histogram created according to the hue channel. Though it improves the robustness against lighting changes when taking the hue channel to create the color model, it brings in a new problem. When pixels' color has a low saturation near zero, which means the RGB channels have similar values, hue is not defined or inaccurate.

In these cases, the hue value can not represent the original RGB color reliably, which results in inaccuracy and noise in the back projection image. This is the first deficiency of using single color cue.

The second deficiency is similarly colored backgrounds. Mean Shift is robust to small distractors, but if the distractor is larger than the object color area, the object may be lost when it moves near the similar distractor.

B. Direct Motion Cue Integration

It should be noticed that the distrators are all from the background. When the camera is static, the background is assumed not moving, which can be used as a priori to eliminate those noisy areas in the back-projection image. Therefore, it is believed that motion information can be employed to solve the deficiencies.

First, adaptive background subtraction technique[12] is employed to get the entering object's silhouette. Suppose $I(X,t)$ is the incoming frame, $B(X,t)$ is the updated background image, th is an appropriate threshold, then:

$$D(X,t)=\begin{cases}1, & |I(X,t)-B(X,t)|>th \\ 0, & else\end{cases} \tag{1}$$

The motion image $D(X, t)$ obtained from motion detection can be regarded as representing the pixels' probabilistic distribution of motion, so it is convenient to be integrated into other 2D discrete distributions. What's more, as background subtraction results always have distractors, combining with other complementary cue, e.g. color, may be helpful to eliminate distractors and noise. For each image I, denote $p_m(X,t)$ to be the motion probability of the pixel at X at time t. Let

$$p_m(X,t)=D(X,t) \tag{2}$$

Here, $p_m(X,t)$ is a binarized distribution that represents the probability of motion for each pixel. Then, $p_m(X,t)$ is integrated into the original color probability distribution $p_c(X,t)$.

How to combine both maps is of our interest. An intuitive direct cue-integration method is to use pixel-wise *logical and* operator, and yields a probability distribution of moving color pixels,

$$p(X,t)=\bigcap_i p_i(X,t) \tag{3}$$

The value of probability $p_i(X, t)$ may not be a binary value, as in the color probabilistic distribution map, $p_c(X, t)$ can be any value in [0, 1]. Therefore, the pixel-wise *logical and* operator is adjusted a little. Suppose $a \geq 0, b \geq 0$, we define and operation as:

$$a \cap b = \min(a,b) \tag{4}$$

If either a or b is 0, the result will be 0. After direct cue integration, Mean Shift can be employed to find the mode of the distribution.

The color-motion integration based Mean Shift algorithm includes following steps:

(1) Calculate the color probabilistic distribution map $p_c(X, t)$ through back projection.
(2) Calculate the motion distribution map $p_m(X, t)$ through motion detection. Integrate both maps using formula (3).
(3) Choose a search window scale s_0 and initial location P_0 on the combined distribution map $p(X, t)$.
(4) Compute the mean location $\hat{P}$ and zeroth moment M_{00} of the pixels in the search window (P, s).
(5) Set the new window parameters as $P=\hat{P}$, $s=k\sqrt{M_{00}}$. (k is a constant).
(6) Repeat steps (4) and (5) until convergence.

The motion and the color cues are employed explicitly. Motion continuity has been implicitly used, as it is initialized according to the tracking result of the last frame. Note that the integration scheme in formula (3) is open and more cues can be integrated in.

C. Occlusion Handling

Mean Shift algorithm is vulnerable to full occlusion for a few frames because present iteration is initialized according to the previous. If the object is totally occluded for a couple of frames, the tracking window will drift away and the algorithm has no mechanism to continue tracking. Based on the direct color-motion integration mechanism, an occlusion handler can also be advanced, which can be used to detect the full occlusion cases and to reinitialize tracking automatically when the object reappears.

Based on the direct-cue integration, a distribution map with little background noise is obtained, which makes it possible to search large non-zero region on the distribution map to find the reappearing object. Without the direct-cue integration, the background noise may cause the occlusion handler to fail. Fig.1 shows the flow chart of the direct cue integration based occlusion handler.

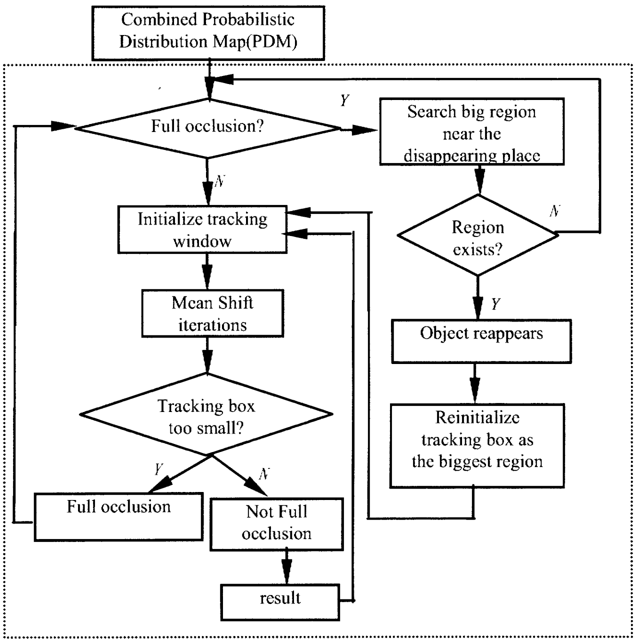


Fig.1 Direct color-motion integration based Mean Shift occlusion handler (emphasized by the dotted box).

If the object color is fully occluded by some other objects, the tracking window will shrink. When the window area or the density of non-zero pixels in the window is smaller than respective thresholds, it is regarded as the full occlusion case. In such case, large non-zero regions are searched on the object's probabilistic distribution map (PDM) near the place where the object disappeared. If some large regions are found, the largest region is used to initialize the tracking window.

With both the color-motion integration method and the occlusion handler, we can deal with color background clutter and full occlusion over a few frames, which are said to be the deficiency of deterministic methods, mentioned in [3]. What's more, the occlusion handler can handle long time complete occlusion or the object's departure from the FOV of the camera for a couple of frames, which can not be handled by multi-hypothesis based probabilistic tracking methods, like Particle Filter.

III. Adaptive Multi-Cue Integration

Though direct multi-cue integration can enhance tracking performance of color-base Mean Shift algorithm, it will erode the color probabilistic image because of the inevitable holes in the motion detection results. This can be a disadvantage of direct integration, when an object's color has sufficiently high saturation component and its color probabilistic map alone is good enough for tracking. What's more, the direct multi-cue integration method assumes that the contribution of each cue is the same, regardless of their reliability. Hence, we employ an adaptive multi-cue integration technique. Our work differs from [14] in the point that we employ the adaptive integration mechanism under the framework of Mean Shift, with a new quality function suitable for blob tracking.

Suppose $p_i(X, t)$ is the probability distribution map of cue i, $p(X, t)$ is the combined probability distribution map, the cues are integrated as a weighted sum of probability distribution,

$$p(X,t) = \sum_i \omega_i(t) \times p_i(X,t) \tag{5}$$

$$\sum_i \omega_i(t) = 1 \tag{6}$$

The adaptive integration method adapts each cue's weight according to the reliability of each cue in previous frame. Suppose the performance of individual cue i can be evaluated using a quality function $q_i(t)$. The normalized quality of cue i:

$$\bar{q}_i(t) = \frac{q_i(t)}{\sum_i q_i(t)} \tag{7}$$

The relation between the quality and weight of cue i can be defined as following,

$$\tau \dot{\omega}_i(t) = \bar{q}_i(t) - \omega_i(t) \tag{8}$$

Formula (8) can be used to update individual weight of each cue. As the quality function $q_i(t)$ is normalized, the weight $\omega_i(t)$ is normalized as well. τ is a time constant controlling the speed of update. If $\bar{q}_i(t) > \omega_i(t)$, $\omega_i(t)$ tends to increase. Here $q_i(t)$ represents the feedback of tracking results, so equation (8) can be regarded as a running average, and $\omega_i(t)$ is adapted according to $q_i(t)$, which brings in the information about the performance of cue i in the last frame. The problem is how to define an appropriate quality function $q_i(t)$.

Quality functions $q_i(t)$ can be viewed as feedback of the tracking result $\hat{X}(t) = (P, s)$. Here P and s are the estimated center and scale of the track window, respectively. Each cue's weight is adjusted according to the quality of the cue in last frame. In this paper, the quality function can be defined as the ratio between the numbers of non-zero pixels inside and outside the track window on individual probability distribution map. Denote I to be the whole image area, $q_i(t)$ is defined as:

$$q_i(t) = \frac{nonzero\_pixels(p_i(x,t), \hat{X}(t))}{nonzero\_pixels(p_i(x,t), I - \hat{X}(t))} \tag{9}$$

This quality function is easy to implement. Each cue's performance is evaluated by this quality function and the weights are adapted accordingly. When object with low saturation color is trracked, the quality function value of color cue will be much lower than that of motion cue and the weight of motion cue dominates. When object changes its color appearance, the color-based algorithm would fail because the tracked color is invisible, but the adaptive multi-cue based Mean Shift is able to continue because of cues other than color.

The adaptive weighted sum integration is different from the direct integration method. In the direct integration method, if a pixel value is zero in the motion probabilistic distribution map, its probability value is set to 0 no matter what its value in the color probabilistic distribution map is. However, in the weighted sum integration the combined probability value is always decided by both the color probability and motion probability. Considering possible detection holes in the cue extraction process, the adaptive weighted sum integration is more robust to detection holes than the direct integration method.

After the combined probability distribution map is obtained, a region detection algorithm should operate on the combined map and find the object. Spengler et. al.[11] uses projection of the combined distribution $p(X, t)$ to the coordinate axes to find the estimated position, in this paper more robust Mean Shift algorithm is used, with a new quality function.

The motion cue is inherently very suitable to be integrated into the Mean Shift framework, using above adaptive integration method. Background subtraction results usually have holes and patches of noise. The holes may be remedied by other cues by the weighted sum integration. Moreover, as Mean Shift is robust to small distractors, noise from background subtraction are no longer a problem.

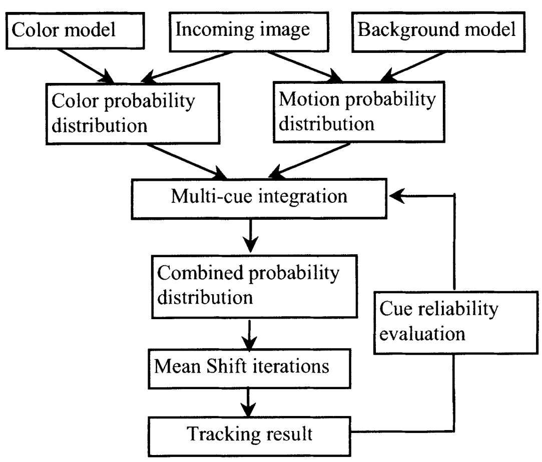


Fig.2 flow chart of adaptive color-motion integration based Mean Shift tracking algorithm

The flow chart of adaptive multi-cue integration is illustrated in Fig.2. Note that cue performance evaluation forms a feedback loop, which is the most prominent difference from direct multi-cue integration. Each cue's performance is evaluated in the cue evaluation phase through the new quality function.

IV. Experiments

To evaluate the effectiveness of the above methods, an experiment system is designed. Experiments were carried out on a PC with a 1.8 GHz P4 CPU and 523MB memory. Pixels that have saturation lower than 30 and have brightness lower than 10 are discarded. First we test the direct color-motion integration method. The occlusion handler based on the direct cue integration is tested as well. At last, the advantage of adaptive cue integration is demonstrated. S1 and S2 are the sequences with low saturation objects. S3 and S4 are sequences in which good color cue is tracked with similar background color clutter. S5 and S6 are multi-people sequences in which good color cue is tracked but without occlusion. S7 to S12 are video sequences with one person occluded by the other. All the results are real-time.

A. Direct Motion Cue Integration

Algorithm is tested with video sequences. A 1D hue-histogram with 16 bins is used. In cases with only color cue, object was lost after initialization later.

Table 1 shows the tracking results in four video sequences, from S1 to S4. Original color-based Mean Shift algorithm, CAMSHIFT, failed in all sequences. Tracking window converged to a wrong region or is much larger than the object color area is defined as tracking failure.

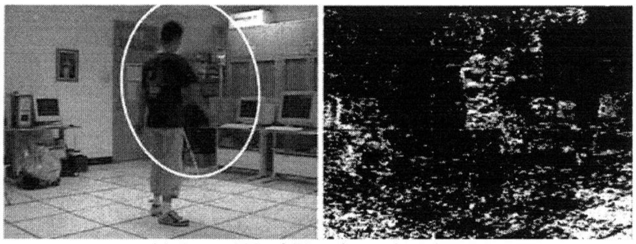
TABLE I
COMPARISONS OF TRACKING RESULTS OF INTEGRATING AND NOT INTEGRATING MOTION INFORMATION

Video Sequence	Integrate Motion Info.	Model Initialization (n-*th* frame)	Fail (n-*th* frame)	Successful rate
S1	N	60	175	82%
(200 frames)	Y		Success	100%
S2	N	30	112	48%
(200 frames)	Y		Success	100%
S3	N	30	151	71%
(200 frames)	Y		Success	100%
S4	N	40	83	27%
(200 frames)	Y		Success	100%

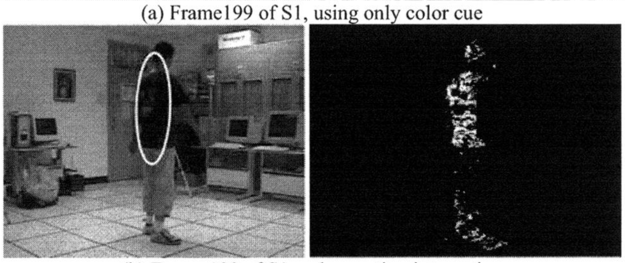
S1 and S2 are the sequences with low saturation objects. S3 and S4 are sequences with similar background color clutter.

Fig.3 shows comparison between integrating and not integrating motion cue algorithms in frame 199 of Sequence S1. It can be seen from the results that Integration of motion information helps color-based Mean Shift algorithm overcome the difficulty of tracking object with low saturation color.

Even when the saturation of the object color is not low, background color clutter may also cause tracking failure. Integrating motion cue enables color-based Mean Shift algorithm to track objects in this case. Fig.4 shows the results of a real-time video sequence in frame 90 of sequence S3, the lighting condition in this sequence is dark, which makes the background pixels' hue values not accurate and forms a large area of distractor. Motion cue helps to eliminate the distractor at the door, and enables the color-based Mean Shift algorithm to converge to the right color area.

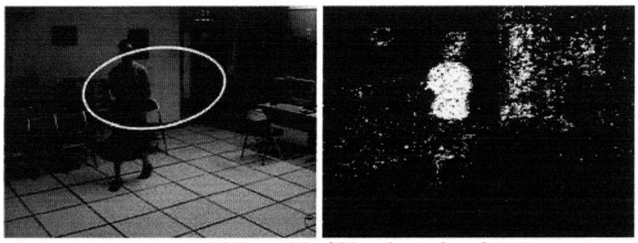


(a) Frame199 of S1, using only color cue

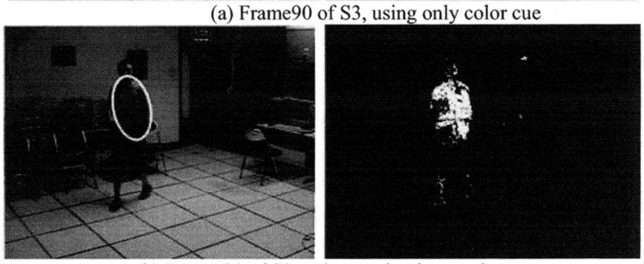


(b) Frame 199 of S1, color-motion integration

Fig.3. Tracking results in frame 199 of video sequence S1: tracking object with low saturation color. (a) shows the tracking results and probabilistic distribution maps using only color cue, (b) show the results after color-motion cue integration, compared with (a).



(a) Frame90 of S3, using only color cue



(b) Frame 90 of S3, color-motion integration

Fig.4. Tracking results in frame 90 of video sequence S3: tracking human under similarly colored backgrounds. (a) show the tracking result and corresponding probabilistic distribution map using only color cue, (b) show the results after color-motion cue integration, compared with (a).

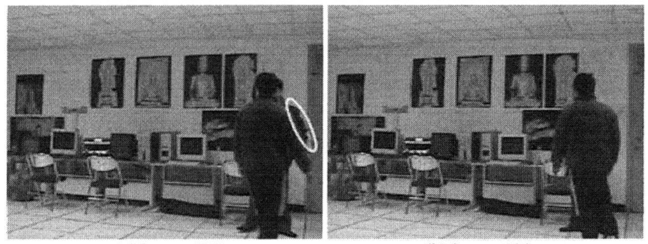
B. Occlusion Handling

The occlusion handler is tested through multiple-human video sequences. In these sequences, the occluded person is tracked. When the occluded person reappears from occlusion, the occlusion can reinitialize tracking and recover from object lost. In sequences S11 and S12, the tracked person was lost for over 20 frames, however, the occlusion handler can still reinitialize tracking after it reappears.

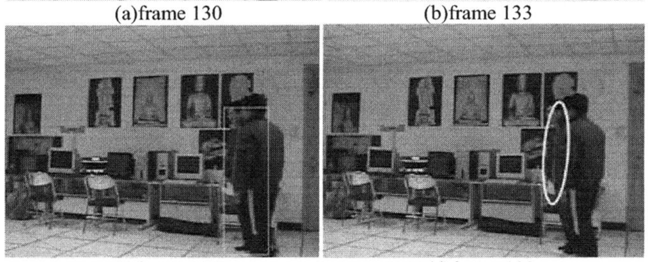
TABLE II
RESULTS OF OCCLUSION HANDLING

Sequence	Full Occlusion time(Frames)	Recovered from FO
S7	3	Y
S8	2	Y
S9	7	Y
S10	5	Y
S11	21	Y
S12	45	Y

Fig.5 shows a full occlusion case in sequence S9. Color model is initialized on the boy's red coat before occlusion. In frame 130, full occlusion occurs. In frame 137, the occlusion handler can recover from full occlusion and reinitialize tracking.



(a)frame 130 (b)frame 133



(c)frame 137 (d)frame 138

Fig.5. Full occlusion case from video sequence S9. (a),(b),(c) and (d) are tracking results from S9.

C. Adaptive Color- Motion Integration Based Mean Shift

In the adaptive multi-cue integration experiments, 2D histogram is used, with the hue component discretized into 16 bins and the saturation into 10 bins, other experimental conditions unchanged.

Adaptive multi-cue integration strategy is tested in sequences S1 to S6, the object can be tracked during the whole sequences. Fig.6 shows a representative result of tracking low saturation color in video sequence S1. Color model is selected according to the boy's shirt. As color cue has a low value of quality function, its weight is diminished and the weight of motion cue (the lighter curve) increases. It can be seen from Fig.7 that the motion cue dominates.

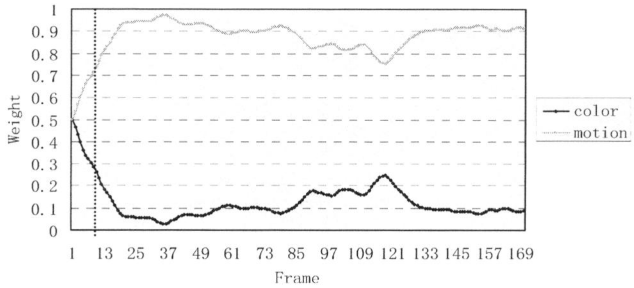
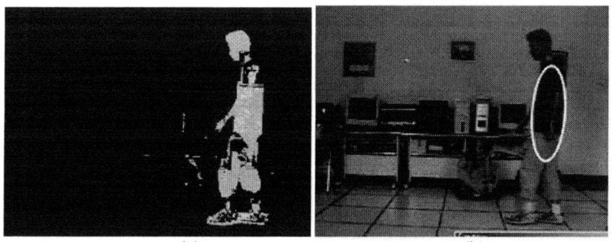


Fig. 6. Adaptive color-motion integration for tracking low saturation color. The lighter curve is the weight of motion cue, and darker is the weight of color cue. The vertical dotted line indicates the time when Fig.7 is taken.



(a) (b)

Fig. 7. Adaptive color-motion integration: track low saturation color. (a) is combined probability density maps. (b) is the corresponding tracking result. Motion cue is dominating. Refer to Fig.6.

When the color cue is reliable, i.e. there are little distractors on the color probability map, the adaptive integration mechanism should not depress the color cue, but should give it a higher weight. Cases when color is reliable are tested, too. Sequence S3 to S6 are sequences where good color features are tracked, Table 3 shows the tracking results. In all sequences, color cue has higher average weights than motion cue. Fig.8 shows a typical result from video sequence S4. The color model is initialized on the boy's orange T-shirt. The color cue becomes dominant after initialization. It can also be seen from Fig.9 that both cues have impact on the combined probability distribution map and that the color cue dominates.

TABLE III
RESULT OF TRACKING RELIABLE COLOR CUE USING ADAPTIVE CUE INTEGRATION

Video Sequence	Model Initialization (n-*th* frame)	Average weight (color)	Average weight (motion)
S3 (200 frames)	67	0.91	0.09
S4 (200 frames)	47	0.63	0.37
S5 (384 frames)	80	0.69	0.30
S6 (384 frames)	84	0.52	0.48

Notice that in all sequences, the color cue has a higher average weight.

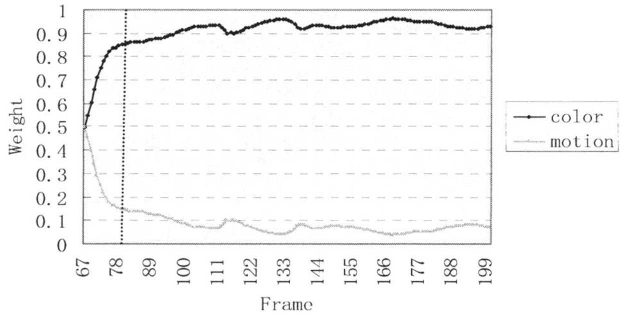
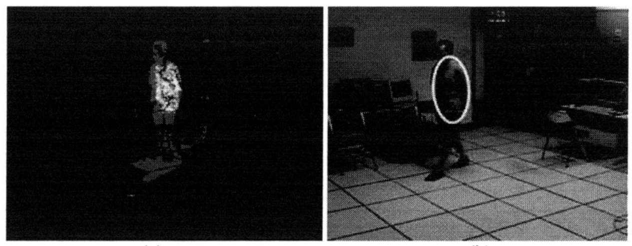


Fig. 8. Adaptive color-motion integration: track reliable color. The lighter curve is the weight of motion cue, and darker is the weight of color cue. The vertical dotted line indicates the time when Fig.9 is taken.



(a) (b)

Fig. 9. Adaptive color-motion integration: track reliable color. (a) is combined probability density maps. (b) is the corresponding tracking result. Color cue is dominating. Refer to Fig.8.

With adaptive cue integration, the problem of changing appearance caused by human rotation can be handled as well, which is demonstrated in Fig.10 and Fig.11. Color model is initialized with the bright blue pattern on the boy's T-shirt when he is facing to the camera. For the first few frames, the weight of color cue has a tendency to increase because of its high quality. At frame 110, the boy begins to turn left and walk to the right of the image. The blue pattern becomes invisible, in which case the color-based tracking algorithm would fail. However, the object can still be tracked because the weight of motion cue begins to increase and becomes dominant in the combined map. The failed color cue is compensated by the motion cue.

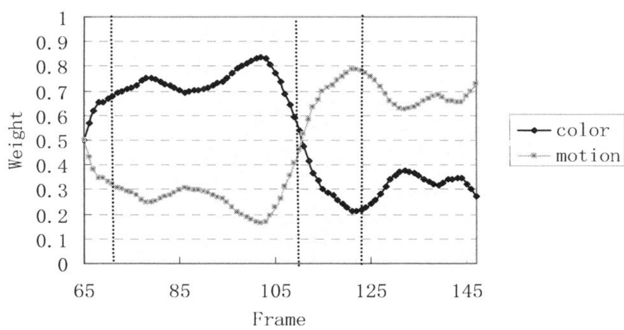


Fig. 10. Adaptive color-motion integration: object changes appearance. The lighter curve is the weight of motion cue, and darker is the weight of color cue. The vertical dotted line indicates the time when Fig.11 is taken.

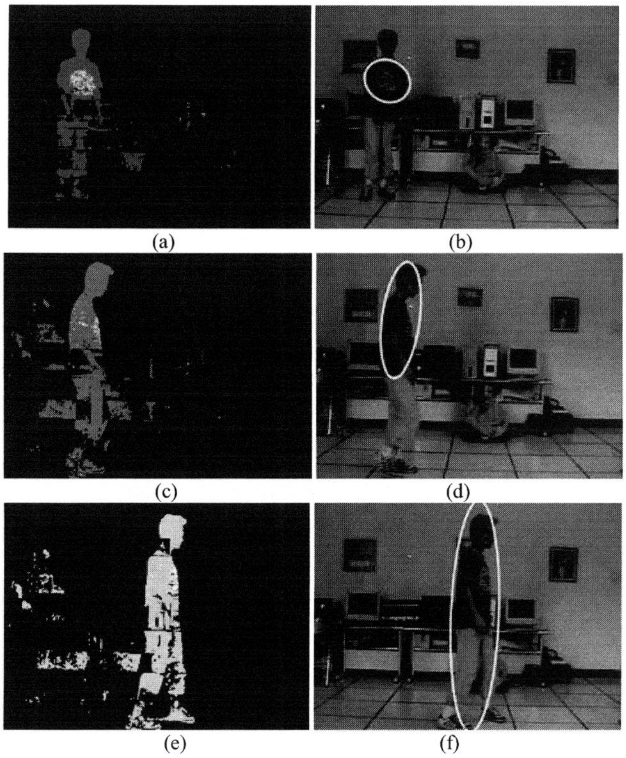


Fig. 11. Adaptive color-motion integration: object changes appearance. (a), (c) and (e) are combined probability density maps. (b), (d) and (f) are the corresponding tracking results. Refer to Fig.10.

This experiment also demonstrates that the motion probabilistic distribution map is suitable to be used under the Mean Shift framework. Note the distractor in the left of Fig.11(e) which is brought in by motion probabilistic distribution map. Mean Shift is robust to these small distractors..

V. CONCLUSIONS

In this paper, it has been demonstrated that the motion cue can be integrated with color cue to solve the problems of tracking low saturation color and tracking under background color clutter in the color-based Mean Shift algorithm. Based on the direct cue integration, an occlusion handler is proposed to handle complete occlusion for a couple of frames in the Mean Shift algorithm as well. The method of adaptive multi-cue integration is more robust to detection holes than the direct integration. A quality function is advanced to evaluate the reliability of each cue in the adaptive method. When color cue is more reliable, its weight will become higher than motion cue. When the color cue is less reliable, it is compensated by the motion cue. This adaptive cue integration method can handle the problem of tracked color becoming invisible as well. However, this adaptive cue integration method can not handle full occlusion between moving objects, as it will increase the weight of motion cue when occlusion occurs. Handing full occlusion under the adaptive cue integration mechanism is our future work.

REFERENCES

[1] M. Isard and A. Blake, "CONDENSATION - conditional density propagation for visual tracking", Int. Journal of Computer Vision, pp.5-28, 1998.

[2] K. Nummiaro, E. Koller-Meier and L. Van Gool, "Object tracking with an adaptative color-based Particle Filter", Image and Vision Computing, pp.99-111, 2002.

[3] P. Perez, C. Hue, J. Vermaak and M. Gangnet, "Color-based probabilistic tracking", European Conference on Computer Vision, pp 661-675, 2002.

[4] G.R. Bradski, "Computer vision face tracking for use in a perceptual user interface", IEEE Workshop on Applications of Computer Vision, pp.214-219, 1998.

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

[6] T.-L. Liu, and H.-T. Chen, "Real-time tracking using trust-region methods", IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 397 - 402 , vol. 26, Issue 3, 2004.

[7] R.T. Collins, "Mean-shift Blob Tracking through Scale Space", IEEE Conference on Computer Vision and Pattern Recognition, Proceedings, pp. 234-240, vol.2, 2003.

[8] Z. Zivkovic and B. Krose, "An EM-like algorithm for color-histogram-based object tracking", IEEE Conference on Computer Vision and Pattern Recognition, Proceedings, pp. 798-803, 2004.

[9] T. Wang, Q. Diao, Y. Zhang, G. Song, C. Lai and G. Bradski, "A Dynamic Bayesian Network Approach to Multi-cue based Visual Tracking", Proc. of Int. Conference on Pattern Recognition, pp. 167 - 170, 2004.

[10] Y. Wu and T.S. Huang, "A Co-inference Approach to Robust Visual Tracking", Proc. of Int. Conference on Computer Vision, pp. 26-33, Vol.2, 2001.

[11] M. Spengler and B. Schiele, "Toward robust multi-cue integration for visual tracking", Machine Vision and Applications, pp.50-58, vol.14, 2003.

[12] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld and H. Wechsler, "Tracking groups of people", Computer Vision and Image Understanding, pp.42-56, vol. 80, 2000.

[13] M. Swain and D. Ballard, "Color indexing", Int. Journal of Computer Vision, pp. 11-32, 1991.

[14] S. J. McKenna, Y. Raja and Shaogang Gong, "Tracking colour objects using adaptive mixture models", Image Vision Computing. pp. 225-23, 1999.

[15] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld and H. Wechsler, "Tracking groups of people", Computer Vision and Image Understanding, pp.42-56, vol. 80, 2000.

[16] J. Triesch and C. Malsburg, "Self-organized integration of adaptive visual cues for face tracking", Proc. of the 4th IEEE Int. Conference on Automatic Face and Gesture Recognition, pp. 102 - 107, 2000.

[17] M.S. Arulampalam, S. Maskell, N. Gordon and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking", IEEE Transactions on Signal Processing, pp. 174-188, Vol. 50, 2002.

[18] D. Comaniciu, V. Ramesh and P. Meer, "Kernel-based object tracking", IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.564-577, Vol. 25, 2003.

[19] J. Vermaak, P. Perez, M. Gangnet, and A. Blake, "Towards improved observation models for visual tracking: selective adaptation", Proc. of the European Conference on Computer Vision, pp.645-660, 2002.

[20] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models", Proc. 1st Int. Conference on Computer Vision, pp. 259-268, 1987.

[21] I. Haritaoglu, D. Harwood and L. S. Davis, "W4: Real-Time Surveillance of People and Their Activities", IEEE Trans. On Pattern Analysis and Machine Intelligence, pp. 809-830, Vol. 22, July 2000.

[22] C. R. Wren and A. P. Pentland, "Pfinder: Real-Time Tracking of Human Body", IEEE Trans. On Pattern Analysis and Machine Intelligence, pp. 780-785, Vol. 19, July 1997.