ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

DEVELOPMENT OF A HIGHLY EFFICIENT OBJECT TRACKING SYSTEM USING MODIFIED MEAN SHIFT TRACKING

Abhishek Kesharwani[1], Preeti Tuli[2]

PG Student, Dept. of CSE, Disha Institute of Management & Technology, Raipur, Chhattisgarh, India[1]

Reader, Dept. of CSE, Disha Institute of Management & Technology, Raipur, Chhattisgarh, India [2]

ABSTRACT: Object tracking is the process of locating a moving object (or multiple objects) over time using a camera. It has a variety of uses, some of which are: human-computer interaction, security and surveillance, video communication and compression, augmented reality, traffic control, medical imaging and video editing. Video tracking can be a time consuming process due to the amount of data that is contained in video. A lot of object tracking algorithms have been reported in literatures, but the area is still lacking with an efficient algorithm which can, not only track the objects but at the same time able to recognize the orientation and movement of object. In this paper an efficient object tracking system is proposed based on Modified mean shift tracking (MMST) algorithm. This work basically deals with how to address the problem to estimate the scale and orientation changes of the target under the mean shift tracking framework. In the original mean shift tracking algorithm, the position of the target can be well estimated, while the scale and orientation changes cannot be adaptively estimated. This paper presents an efficient modification on available mean shift tracking technique, that the weight image derived from the target model and the candidate model can represent the possibility that a pixel belongs to the target, in this work it will be shown that the original mean shift tracking algorithm can be derived using the zeroth and the first order moments of the weight image.

Key Words: Object Tracking, Mean Shift Tracking, Scale and Orientation, MMST.

I. INTRODUCTION

The objective of object tracking is to associate target objects in consecutive video frames. The association can be especially difficult when the objects are moving fast relative to the frame rate. Another situation that increases the complexity of the problem is when the tracked object changes scale and orientation over time. For these situations object tracking systems usually employ a motion model which describes how the image of the target might change for different possible motions of the object.

In the classical mean shift tracking algorithm [9], the estimation of scale and orientation changes of the target is not solved. Although it is not robust, the CAMSHIFT algorithm [6], as the earliest mean shift based tracking scheme, could actually deal with various types of movements of the object. In CAMSHIFT, the moment of the weight image determined by the target model was used to estimate the scale (also called area) and orientation of the object being tracked. Based on Comaniciu *et al*'s work in [9], many tracking schemes [10, 11, 17, 18, and 23] were proposed to solve the problem of target scale and/or orientation estimation. Collins [10] adopted Lindeberg *et al*'s scale space theory [19, 20] for kernel scale selection in mean-shift based blob tracking. However, it cannot handle the rotation changes of the target. An EM-shift algorithm was proposed by Zivkovic and Krose in [11], which simultaneously estimates the position of the local mode and the covariance matrix that can approximately describe the shape of the local mode. In [23], a distance transform based asymmetric kernel is used to fit the object shape through a scale adaptation followed by a segmentation process. Hu *et al* [17] developed a scheme to estimate the scale and orientation changes of the object by using spatial-color features and a novel similarity measure function [12, 16]. In this paper, a Modified mean shift tracking (MMST) algorithm is proposed under the mean shift framework.

II. MEAN SHIFT TRACKING ALGORITHM

A. *Target Representation*

In object tracking, a target is usually defined as a rectangle or an ellipsoidal region in the image. Currently, a widely used target representation is the color histogram because of its independence of scaling and rotation and its robustness to partial occlusions [9, 21]. Denote by $\{X_i^*\}_{i=1\cdots n}$ the normalized pixels in the target region, which is

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

supposed to be centered at the origin point and have n pixels. The probability of the feature u (u=1, 2… m) in the target model is computed as [9].

$$\begin{cases} \hat{\mathrm{q}} = \{\hat{q}_u\}_{u=1\cdots m} \\ \hat{q}_u = C\sum_{i=1}^{n} k\left(\left\|\mathrm{x}_i^*\right\|^2\right)\delta\left[b\left(\mathrm{x}_i^*\right)-u\right] \end{cases} \tag{2.1}$$

Where $\hat{q}$ is the target model, $\hat{q}_u$ is the probability of the u^{th} element of $\hat{q}$, δ is the Kronecker delta function, $b\{X_i^*\}$ associates the pixel X_i^* to the histogram $\mathrm{b_{in}}$, and $k(x)$ is an isotropic kernel profile. Constant C is a normalization function defined by

$$C = 1\Big/\sum_{i=1}^{n} k\left(\left\|\mathrm{x}_i^*\right\|^2\right) \tag{2.2}$$

Similarly, the probability of the feature u in the target candidate model from the candidate region centered at position y is given by

$$\begin{cases} \hat{\mathrm{p}}(\mathrm{y}) = \{\hat{p}_u(\mathrm{y})\}_{u=1\cdots m} \\ \hat{p}_u(\mathrm{y}) = C_h\sum_{i=1}^{n_h} k\left(\left\|\frac{\mathrm{y}-\mathrm{x}_i}{h}\right\|^2\right)\delta\left[b(\mathrm{x}_i)-u\right] \end{cases} \tag{2.3}$$

$$C_h = 1\Big/\sum_{i=1}^{n_h} k\left(\left\|\frac{\mathrm{y}-\mathrm{x}_i}{h}\right\|^2\right) \tag{2.4}$$

Where $\hat{p}(y)$ is the target candidate model, $\hat{p}_u(y)$ is the probability of the u^{th} element of $\hat{p}(y)$. $\{X_i\}_{n=1\cdots n_h}$ are pixels in the target candidate region centered at y, h is the bandwidth and C_h is the normalization function which is independent of y [9].

In order to calculate the likelihood of the target model and the candidate model, a metric based on the Bhattacharyya coefficient [1] is defined by using the two normalized histograms $\hat{p}(y)$ and $\hat{q}$ as follows

$$\rho[\hat{\mathrm{p}}(\mathrm{y}),\hat{\mathrm{q}}] = \sum_{u=1}^{m}\sqrt{\hat{p}_u(\mathrm{y})\hat{q}_u} \tag{2.5}$$

The distance between $\hat{p}(y)$ and $\hat{q}$ is then defined as

$$d[\hat{\mathrm{p}}(\mathrm{y}),\hat{\mathrm{q}}] = \sqrt{1-\rho[\hat{\mathrm{p}}(\mathrm{y}),\hat{\mathrm{q}}]} \tag{2.6}$$

Minimizing the distance $d[\hat{p}(y),\hat{q}]$ in Eq. (2.6) is equivalent to maximizing the Bhattacharyya coefficient $\rho[\hat{p}(y),\hat{q}]$ in Eq. (2.5). The optimization process is an iterative process and is initialized with the target position, denoted by $\mathrm{y_o}$ in the previous frame. By using the Taylor expansion around coefficient, $\hat{p}_u(y_o)$, the linear approximation of the Bhattacharyya in Eq. (2.5) can be obtained as:

$$\rho[\hat{\mathrm{p}}(\mathrm{y}),\hat{\mathrm{q}}] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\mathrm{y}_0)\hat{q}_u} + \frac{C_h}{2}\sum_{u=1}^{n_h} w_i k\left(\left\|\frac{\mathrm{y}-\mathrm{x}_i}{h}\right\|^2\right) \tag{2.7}$$

Where,

$$w_i = \sum_{u=1}^{m}\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\mathrm{y}_0)}}\delta[b(\mathrm{x}_i)-u] \tag{2.8}$$

Since the first term in Eq. (2.7) is independent of y, to minimize the distance in Eq. (2.6) is to maximize the second term in Eq. (2.7). In the mean shift iteration, the estimated target moves from y to a new position $\mathrm{y_1}$, which is defined as

$$\mathrm{y}_1 = \frac{\sum_{i=1}^{n_h}\mathrm{x}_i w_i g\left(\left\|\frac{\mathrm{y}-\mathrm{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h} w_i g\left(\left\|\frac{\mathrm{y}-\mathrm{x}_i}{h}\right\|^2\right)} \tag{2.9}$$

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

When we choose the kernel $k(x)$ with the Epanechnikov profile, there is $g(x)$ =-$k(x)$ =1, and Eq. (2.9) can be reduced to [9].

$$\mathbf{y}_1 = \frac{\sum_{i=1}^{n_h} \mathbf{x}_i w_i}{\sum_{i=1}^{n_h} w_i} \tag{2.10}$$

By using Eq. (2.10), the mean shift tracking algorithm finds in the new frame the most similar region to the object. From Eq. (2.10) it can be observed that the key parameters in the mean shift tracking algorithm are the weights w_i. In this project we will focus on the analysis of w_i with which the scale and orientation of the tracked target can be well estimated, and then a scale and orientation adaptive mean shift tracking algorithm can be developed.

B. Modified Mean Shift Tracking For Scale And Orientation Of Target.

In this section, we first analyse how to calculate adaptively the scale and orientation of the target in sub-sections II.II.I ~ II.II.V, then in sub-section II.II.VI, a modified mean shift tracking (MMST) algorithm for scale and orientation of target is presented.

The enlarging or shrinking of the target is usually a gradual process in consecutive frames. Thus we can assume that the scale change of the target is smooth and this assumption holds reasonably well in most video sequences. If the scale of the target changes abruptly in adjacent frames, no general tracking algorithm can track it effectively. With this assumption, we can make a small modification of the original mean shift tracking algorithm. Suppose that we have estimated the area of the target (the area estimation will be discussed in sub-section II.II.II) in the previous frame, in the current frame we let the window size or the area of the target candidate region be a little bigger than the estimated area of the target. Therefore, no matter how the scale and orientation of the target change, it should be still in this bigger target candidate region in the current frame. Now the problem turns to how to estimate the real area and orientation from the target candidate region.

1. The Weight Images For Target Scale Changing

In the CAMSHIFT and the mean shift tracking algorithms, the estimation of the target location is actually obtained by using a weight image [10, 24]. In CAMSHIFT, the weight image is determined using a hue-based object histogram where the weight of a pixel is the probability of its hue in the object model. While in the mean shift tracking algorithm, the weight image is defined by Eq. (2.8) where the weight of a pixel is the square root of the ratio of its color probability in the target model to its color probability in the target candidate model.
Moreover, it is not accurate to use the weight image by CAMSHIFT to estimate the location of the target, and the mean shift tracking algorithm can have better estimation results. That is to say, the weight image in the mean shift tracking algorithm is more reliable than that in the CAMSHIFT algorithm.

2. Estimating The Target Area

Since the weight value of a pixel in the target candidate region represents the probability that it belongs to the target, the sum of the weights of all pixels, i.e., the zeroth order moment, can be considered as the weighted area of the target in the target candidate region:

$$M_{00} = \sum_{i=1}^{n} w(\mathbf{x}_i) \tag{2.11}$$

In mean shift tracking, the target is usually in the big target candidate region. Due to the existence of the background features in the target candidate region, the probability of the target features is less than that in the target model. So Eq. (2.8) will enlarge the weights of target pixels and suppress the weight of background pixels. Thus, the pixels from the target will contribute more to target area estimation, while the pixels from the background will contribute less. On the other hand, the Bhattacharyya coefficient (referring to Eq. (2.5)) is an indicator of the similarity between the target model $\hat{q}$ and the target candidate model $\hat{p}(y)$. A smaller Bhattacharyya coefficient means that there are more features from the background and fewer features from the target in the target candidate region, vice versa. If we take M_{oo} as the estimation of the target area, then according to Eq. (11), when the weights from the target become bigger, the estimation error by taking M_{00} as the area of the target will be bigger, vice versa. Therefore, the Bhattacharyya coefficient is a good indicator of how reliable it is by taking M_{00} as the target area.
We propose the following equation to estimate it:

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

$$A = c(\rho) M_{00} \tag{2.12}$$

Where $c(\rho)$ is a monotonically increasing function with respect to the Bhattacharyya coefficient $\rho(0 \leq \rho \leq 1)$.Here we choose the exponential function as $c(\rho)$ based on our experimental experience:

$$c(\rho) = \exp\left(\frac{\rho - 1}{\sigma}\right) \tag{2.13}$$

From Eqs. (2.12) and (2.13) we can see that when ρ approaches to the upper bound 1, i.e., when the target candidate model approaches to the target model, $c(\rho)$ approaches to 1 and in this case it is more reliable to use M_{00} as the estimation of target area. When ρ decreases, i.e. the candidate model is not identical to the target model, M_{00} will be much bigger than the target area but $c(\rho)$ is less than 1 so that A can avoid being biased too much from the real target area. When ρ approaches to 0, i.e., the tracked target gets lost, $c(\rho)$ will be very small so that A is close to zero.

3. *The Moment Features In Mean Shift Tracking*

In this sub-section, we analyze the moment features in mean shift tracking and then combine them with the estimated target area to further estimate the width, height and orientation of the target in the next sub-section. Like in CAMSHIFT, we can easily calculate the moments of the weight image as follows:

$$M_{10} = \sum_{i}^{n_k} w_i \mathrm{x}_{i,1} \qquad M_{01} = \sum_{i=1}^{n_h} w_i \mathrm{x}_{i,2} \tag{2.14}$$

$$M_{20} = \sum_{i=1}^{n_h} w_i \mathrm{x}_{i,1}^2, M_{02} = \sum_{i=1}^{n_h} w_i \mathrm{x}_{i,2}^2, M_{11} = \sum_{i=1}^{n_h} w_i \mathrm{x}_{i,1} \mathrm{x}_{i,2} \tag{2.15}$$

Where pair $(\mathrm{x}_{i,1}, \mathrm{x}_{i,2})$ is the coordinate of pixel i in the candidate region. Comparing Eq. (2.10) with Eqs. (2.11) and (2.14), we can find that y_1 is actually the ratio of the first order moment to the zeroth order moment:

$$\mathrm{y}_1 = (\bar{\mathrm{x}}_1, \bar{\mathrm{x}}_2) = (M_{10} / M_{00}, M_{01} / M_{00}) \tag{2.16}$$

Where $(\bar{\mathrm{x}}_1, \bar{\mathrm{x}}_2)$ represents the centroid of the target candidate region. The second order center moment could describe the shape and orientation of an object. By using Eqs. (10), (11), (15) and (16), we can convert Eq. (9) to the second order center moment as follows

$$\mu_{20} = M_{20} / M_{00} - \bar{x}_1^2 \qquad \mu_{11} = M_{11} / M_{00} - \bar{x}_1 \bar{x}_2 \qquad \mu_{02} = M_{02} / M_{00} - \bar{x}_2^2 \tag{2.17}$$

Eq. (2.17) can be rewritten as the following covariance matrix in order to estimate the width, height and orientation of the target:

$$Cov = \begin{bmatrix} \mu_{20} & \mu_{11} \\ \mu_{11} & \mu_{02} \end{bmatrix} \tag{2.18}$$

4. *Estimating The Width, Height And Orientation Of The Target*

By using the estimated area (sub-section 2.2.2) and the moment features (sub-section 2.2.3), the width, height and orientation of the target can be well estimated. The covariance matrix in Eq.(2.18) can be decomposed by using the singular value decomposition (SVD) [22] as follows

$$Cov = U \times S \times U^T = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} \times \begin{bmatrix} \lambda_1^2 & 0 \\ 0 & \lambda_2^2 \end{bmatrix} \times \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}^{\mathrm{T}} \tag{2.19}$$

Where $U = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}$ and $S = \begin{bmatrix} \lambda_1^2 & 0 \\ 0 & \lambda_2^2 \end{bmatrix}$. λ_1^2 and λ_2^2 are the Eigen values of Cov. The vectors $(u_{11}, u_{21})^{\mathrm{T}}$ and $(u_{12}, u_{22})^{\mathrm{T}}$ represent, respectively, the orientation of the two main axes of the real target in the target candidate region.

Because the weight image is a reliable density distribution function, the orientation estimation of the target provided by matrix U is more reliable than that by CAMSHIFT. Moreover, in the CAMSHIFT algorithm, λ_1 and λ_2 height of the target, which is actually improper? Next, we present a new scheme to more accurately estimate the width and height of the target.

Suppose that the target is represented by an ellipse, for which the lengths of the semi-major axis and semi-minor axis are denoted by a and b, respectively. Instead of using and λ_1 and λ_2 directly as the width a and height b, it

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

has been shown that the ratio of λ_1 and λ_2 can well approximate the ratio of *a* to *b*, i.e., $\lambda_1/\lambda_2 \approx a/b$ Thus we can set a = k λ_1 and *b*= k λ_2 , where *k* is a scale factor. Since we have estimated the target area *A*, there is $\pi ab = \pi (k\lambda_1)(k\lambda_2) = A$. Then it can be easily derived that

$$k = \sqrt{A/(\pi\lambda_1\lambda_2)} \tag{2.20}$$

$$a = \sqrt{\lambda_1 A/(\pi\lambda_2)} \qquad b = \sqrt{\lambda_2 A/(\pi\lambda_1)} \tag{2.21}$$

Now the covariance matrix becomes

$$Cov = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} \times \begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix} \times \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}^{\mathrm{T}} \tag{2.22}$$

The adjustment of covariance matrix *Cov* in Eq. (22) is a key step of the proposed algorithm. It should be noted that the EM-like algorithm by Zivkovic and Krose [11] estimates iteratively the covariance matrix for each frame based on the mean shift tracking algorithm. Unlike the EM-like algorithm, our algorithm combines the area of target, i.e., *A*, with the covariance matrix to estimate the width, height and orientation of the target.

5. Determining The Candidate Region In Next Frame

Once the location, scale and orientation of the target are estimated in the current frame, we need to determine the location of the target candidate region in the next frame. With Eq. (2.22), we define the following covariance matrix to represent the size of the target candidate region in the next frame

$$Cov_2 = U \times \begin{bmatrix} (a+\Delta d)^2 & 0 \\ 0 & (b+\Delta d)^2 \end{bmatrix} \times U^{\mathrm{T}} \tag{2.23}$$

where Δd is the increment of the target candidate region in the next frame. The position of the initial target candidate region is defined by the following ellipse region

$$(\mathrm{x}-\mathrm{y}_1) \times Cov_2^{-1} \times (\mathrm{x}-\mathrm{y}_1)^{\mathrm{T}} \le 1 \tag{2.24}$$

6. Implementation Of The Mmst Algorithm

Based on the above analyses in sub-sections 2.2.1 ~ 2.2.5, the scale and orientation of the target can be estimated and then a scale and orientation adaptive mean shift tracking algorithm, i.e. the MMST algorithm, can be developed. The implementation of the whole algorithm is summarized as follows.

Algorithm of Modified Mean Shift Tracking (MMST)

1) Initialization: calculate the target model $\hat{q}$ and initialize the position y_0 of the target candidate model in the previous frame.
2) Initialize the iteration number $k \leftarrow 0$.
3) Calculate the target candidate model $\hat{p}(y_0)$ in the current frame.
4) Calculate the weight vector $\{w_i\}_{i=1\cdots n}$ using Eq. (2.8).
5) Calculate the new position y_1 of the target candidate model using Eq. (2.10).
6) Let $d \leftarrow \| y_1 - y_0 \|$, $y_0 \leftarrow y_1$.Set the error threshold ε (default 0.1) and the maximum Iteration number N (default 15).
   $If\ (d < \varepsilon\ or\ k \ge N)$ Stop and go to step 7;
   $Otherwise$ $k \leftarrow k+1$, and go to step 3.
7) Estimate the width, height and orientation from the target candidate model using Eq. (2.22).
8) Estimate the initial target candidate model for next frame using Eq. (2.24).

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

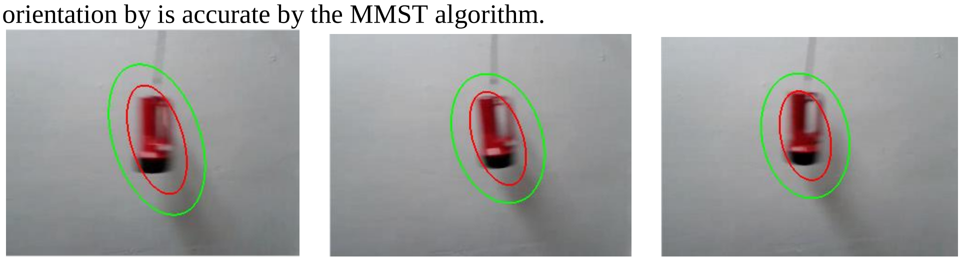
III. EXPERIMENTAL RESULTS AND DISCUSSIONS

This section evaluates the Developed MMST algorithm in comparison with the original mean shift algorithm, i.e., mean shift tracking with a fixed scale, the adaptive scale algorithm [9] and the EM-shift algorithm [11, 25]. The adaptive scale algorithm and the EM-shift algorithm are two representative schemes to address the scale and orientation changes of the targets under the mean shift framework. Because the weight image estimated by CAMSHIFT is not reliable, it is prone to errors in estimating the scale and orientation of the object. So CAMSHIFT is not used in the experiments.

We selected RGB color space as the feature space and it was quantized into 16×16×16 bins for a fair comparison between different algorithms. It should be noted that other color space such as the HSV color space can also be used in MMST. One synthetic video sequence and three real video sequences are used in the experiments.

A. Experiments On Real Video Sequences

The developed MMST algorithm is tested by using four real video sequences. The first video is a Torch sequence recorded in home (Figure 3.1) where the object has clearly scale and orientation changes. To show the efficiency of developed MMST algorithm figure (3.1) consists subsequent frames 20, 40, 80. The second video is a palm sequence (Figure 3.2) of 26 frames, where the object has clearly scale and orientation changes, the estimated target scale and orientation by is accurate by the MMST algorithm.

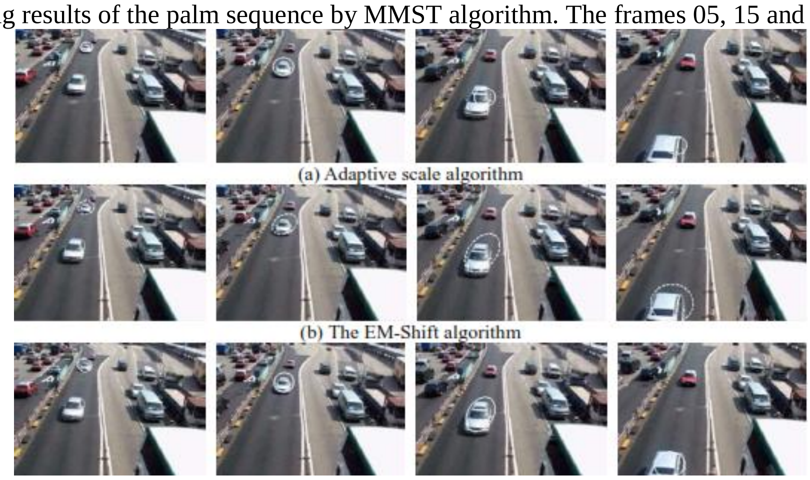


(a) (b) (c)

Fig. 3.1: Tracking results of the torch sequence for MMST algorithms. The frames 20, 40 and 80 are displayed.

(a) (b) (c)

Fig. 3.2: Tracking results of the palm sequence by MMST algorithm. The frames 05, 15 and 26 are displayed.



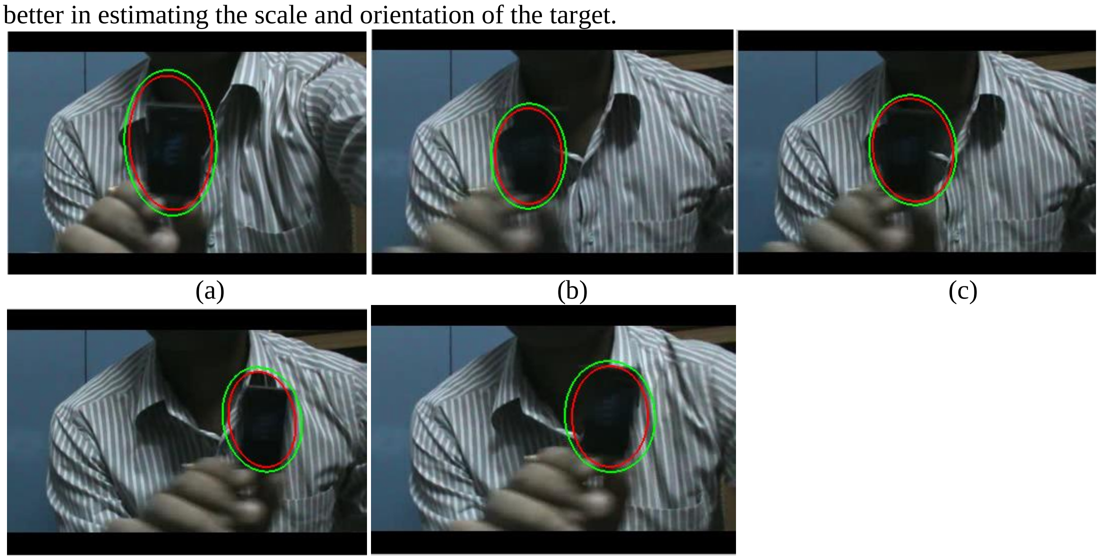
(c) The developed MMST algorithm

Fig. 3.3: Tracking results of the car sequence by different tracking algorithms. The frames 15, 40, 60 and 75 are displayed.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

The last experiment is on a Card reader sequence which is complex because the object is small and having scale and orientation change. The object exhibits large scale changes with partial occlusion. The MMST scheme works much better in estimating the scale and orientation of the target.



(a) (b) (c)

(d) (e)

Fig. 3.4: Tracking results of the Card Reader sequence with MMST algorithms. The frames 10, 40, 50, 60 and 82 are displayed.

Table I. The average number of iterations by different methods on the four sequences.

Methods	Fixed-scale mean shift	Adaptive scale	EM-shift	MMST
Synthetic ellipse	2.34	13.62	6.27	2.59
Car sequence	3.82	11.25	6.34	3.34

Table 1 lists the average numbers of iterations by different schemes on the four video sequences. The average number of iterations of the developed MMST is approximately equal to that of the original mean shift algorithm with fixed scale. The iteration number of the modified scale algorithm is the highest because it runs mean shift algorithm three times. The main factors which affect the convergence speed of the EM-shift and the MMST algorithms are the computation of the covariance matrix. EM-shift estimates it in each iteration while MMST only estimates it once for each frame. So MMST is faster than EM-shift. In general, the developed MMST algorithm, which is motivated by the CAMSHIFT algorithm [6], extends the mean shift algorithm when the target has large scale and orientation variations. It inherits the simplicity and effectiveness of the original mean shift algorithm while being adaptive to the scale and orientation changes of the target.

IV. CONCLUSIONS

By analyzing the moment features of the weight image of the target candidate region and the Bhattacharyya coefficients, we developed a scale and orientation adaptive mean shift tracking (MMST) algorithm. It can well solve the problem of how to estimate robustly the scale and orientation changes of the target under the mean shift tracking framework.

The weight of a pixel in the candidate region represents its probability of belonging to the target, while the zeroth order moment of the weights image can represent the weighted area of the candidate region. By using the zeroth order moment and the Bhattacharyya coefficient between the target model and the candidate model, a simple and effective method to estimate the target area was proposed. Then a new approach, which is based on the area of the target and the corrected second order center moments, was proposed to adaptively estimate the width, height and orientation changes of the target.

The developed MMST method inherits the merits of mean shift tracking, such as simplicity, efficiency and robustness. Extensive experiments were performed and the results showed that MMST can reliably track the objects with scale and orientation changes, which is difficult to achieve by other state-of-the-art schemes. In the future research, we will focus on how to detect and use the true shape of the target, instead of an ellipse or a rectangle model, for a more robust tracking.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 7, July 2013

REFERENCES

[1] Kailath T.: 'The Divergence and Bhattacharyya Distance Measures in Signal Selection', IEEE Trans. Communication Technology, 1967, 15, (1), pp. 52-60.
[2] Fukunaga F., Hostetler L. D.: 'The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition', IEEE Trans. on Information Theory, 1975, 21, (1), pp. 32-40.
[3] Cheng Y.: 'Mean Shift, Mode Seeking, and Clustering', IEEE Trans on Pattern Anal. Machine Intell., 1995, 17, (8), pp. 790-799.
[4] Mukundan R., Ramakrishnan K. R.: 'Moment Functions in Image Analysis: Theory and Applications', World Scientific, Singapore, 1996.
[5] Wren C., Azarbayejani A., Darrell T., Pentland A.: 'Pfinder: Real-Time Tracking of the Human Body', IEEE Trans. Pattern Anal. Machine Intell, 1997, 19, (7), pp. 780-785.
[6] Bradski G.: 'Computer Vision Face Tracking for Use in a Perceptual User Interface', Intel Technology Journal, 1998, 2(Q2), pp. 1-15.
[7] Comaniciu D., Ramesh V., Meer P.: 'Real-Time Tracking of Non-Rigid Objects Using Mean Shift'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Hilton Head, SC, June, 2000, vol. 2, pp. 142-149.
[8] Comaniciu D., Meer P.: 'Mean Shift: a Robust Approach toward Feature Space Analysis', IEEE Trans Pattern Anal. Machine Intell., 2002, 24, (5), pp. 603-619.
[9] Comaniciu D., Ramesh V., Meer P.: 'Kernel-Based Object Tracking', IEEE Trans. Pattern Anal. Machine Intell. 2003, 25, (2), pp. 564-577.
[10] Collins R.: 'Mean-Shift Blob Tracking through Scale Space', Proc. IEEE Conf. Computer Vision and Pattern Recognition, Wisconsin, USA, 2003, pp. 234-240.
[11] Zivkovic Z., Krose B.: 'An EM-like Algorithm for Color-Histogram-Based Object Tracking', Proc. IEEE Conf. Computer Vision and Pattern Recognition, Washington, DC, USA, 2004, vol.1, pp. 798-803.
[12] Yang C., Ramani D., Davis L.: 'Efficient Mean-Shift Tracking via a New Similarity Measure', Proc. IEEE Conf. Computer Vision and Pattern Recognition, San Diego, CA, 2005, vol. 1, pp.176-183.
[13] Fashing M., Tomasi C.: 'Mean Shift is a Bound Optimization', IEEE Trans. Pattern Anal. Machine Intell., 2005, 27, (3), pp. 471-474.
[14] Yilmaz A., Javed O., Shah M.: 'Object Tracking: a Survey', ACM Computing Surveys, 2006, 38, (4), Article 13.
[15] Carreira-Perpinan M. A. 'Gaussian Mean-Shift is an EM Algorithm', IEEE Trans. Pattern Anal. Machine Intell., 2007, 29, (5), pp. 767-776.
[16] Birchfield S., Rangarajan S.: 'Spatiograms versus histograms for region-based tracking', Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2005, vol. 2, pp. 1158–1163, 2005.
[17] Hu J., Juan C., Wang J.: 'A spatial-color mean-shift object tracking algorithm with scale and orientation estimation', Pattern Recognition Letters, 2008, 29, (16), pp. 2165-2173.
[18] Srikrishnan V., Nagaraj T., Chaudhuri S.: 'Fragment Based Tracking for Scale and Orientation Adaption', Proc. Indian Conf. on In Computer Vision, Graphics & Image Processing, 2008, pp. 328-335.
[19] Linderberg T.: 'Feature Detection with Automatic Scale Selection', International Journal of Computer Vision. 1998, 30, (2), pp. 79-116.
[20] Bretzner L., Lindeberg T.: 'Qualitative Multi-Scale Feature Hierarchies for Object Tracking', Journal of Visual Communication and Image Representation, 2000, 11, (2), pp.115-129.
[21] Nummiaro K., Koller-Meier E., Gool L. V.: 'An Adaptive Color-Based Particle Filter', Image and Vision Computing, 2003, 21, (1), pp. 99-110.
[22] Horn R. A., Johnson C. R., Topics in Matrix Analysis, Cambridge University Press, U.K., 1991.
[23] Quast K., Kaup A.: 'Scale and Shape adaptive Mean Shift Object Tracking in Video Sequences', Proc. European Signal Processing Conference, Glasgow, Scotland, 2009, pp. 1513-1517.

BIOGRAPHY

Abhishek Kesahrwani was born in Bilaspur, chhatisgarh, India, on September 21, 1987. He received his B.E degree in Computer Science & Engineering from Institute of Technology, GGU, Bilaspur in 2009: currently he is doing M.Tech degree in Information Security from CSVTU, Bhilai (C.G). His area of interest are digital image processing, software engineering, data mining.

Preeti Tuli received B.E. degree in Computer Engineering from DAVV, Indore in 2000. The M.Tech degree in Computer Science and Engineering from CSVTU, Bhilai in 2010.She is currently working as an Assistant Professor in the Department of Computer Science and Engineering at Disha Institute of Management and Technology, Raipur. She is a Life member of CSI. Her research interests are Natural language processing and image processing.