INTRODUCING NEW PARAMETERS TO COMPARE THE ACCURACY AND RELIABILITY OF MEAN-SHIFT BASED TRACKING ALGORITHMS

S.A. Mohammadi[1] and M.R. Mahzoun[2]

[1]Department of ICT Engineering, IHU University, Tehran, Iran
seyed_amir_mohammadi@yahoo.com
[2]Department of ICT Engineering, IHU University, Tehran, Iran
rezamahzoun@yahoo.com

ABSTRACT

Mean shift algorithms are among the most functional tracking methods which are accurate and have almost simple computation. Different versions of this algorithm are developed which are differ in template updating and their window sizes. To measure the reliability and accuracy of these methods one should normally rely on visual results or number of iteration. In this paper we introduce two new parameters which can be used to compare the algorithms especially when their results are close to each other.

KEYWORDS

Object tracking, Mean-shift algorithm, Adaptive template, Bhattacharyya coefficients

1. INTRODUCTION

Real time object tracking is an important practical application of computer vision with a wide variety of usages in different subjects such as surveillance systems, road traffics, weather forecasting, machine and human interfaces, video compression, astronomy, and a lot of military issues [1-10]. Among different tracking algorithms, mean shift based methods have their special places because of their simplicity in computation while have good accuracies. Updating the target model and making the tracking window size dynamic are two important factors which have made authors develop versions of mean shift tracking algorithms [11-14]. To have a good judgment about the accuracy and reliability of different methods we should use acceptable robust parameters. The comparisons are often done by visual observation while considering the number of iteration in the process. Normally the lesser iteration indicates on higher speed of the tracker, although sometimes it is not satisfactory agument. For example the tracker may loss the target during its process, causing its window be fix in one position which can result in unreliable iteration number. To cope with these kind of problems, one solution is to make and use of more related comparison parametrs.

In this paper after a survey on mean shift algorithm and two developed version of that, we introduce two comparison factors naming them *MCC* and *NV*. Using mentioned factors we compare the results of the algorithms.

DOI : 10.5121/sipij.2011.2305

2. BASIC MEAN-SHIFT (MS) TRACKING ALGORITHM

We use the basis of *MS* algorithm to introduce our parameters. Target features and way of modelling it, target candidates and geometric relations between target model and candidate are the main subject of the algorithm.

2.1. Target features and modelling

In *MS* tracking algorithm the target feature is the colour PDF[1] of target locale which is shown by q in colour space. The probable place of target with center of *y* in next frame contains target candidate. To make target candidate model the PDF of its locale shown by *p(y)* is used. Both PDFs (target and candidate), contains a good approximation of intrinsic feature of target. As one of the trackers evaluation parameters is their processing time, normally to reduce that only some of histogram bars are selected and used. So the target and candidate model is defined as:

Target Model: $\{\hat{q}_u\} \quad u = 1...m \quad \sum_{u=1}^{m} \hat{q}_u = 1$

Candidate Model: $\hat{p}(y) = \{\hat{p}_u(y)\} \quad u = 1...m \quad \sum_{u=1}^{m} \hat{p}_u = 1$

Where *u* indicates on colours exist in target and candidate locales, and $\hat{q}_u$ and $\hat{p}_u$ are normalized target and candidate models. Although histograms are not the best nonparametric approximation of densities but they can be good criterions for target tracking [2]. The similarity of two patterns determine (explained later) by Equation (1).

$$\hat{p}(y) \equiv \rho[\hat{p}(y), \hat{q}] \qquad (1)$$

In this equation $\hat{p}(y)$ measures the similarity between two models while its local maximums indicate on similarity value between target candidate in new frame and the target in first frame.

2.2. Target Model

In our image, target model is shown by a rectangular or circular zone. The function $b: R^2 \rightarrow \{1...m\}$ sets index $b(x_i^*)$ corresponds to quantized histogram bar in colour space to a pixel locates at x_i^*. The values corresponding to existed colour is determines by:

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

where δ is kronecker delta function. The normalization coefficient, *C*, under condition $\sum_{u=1}^{m} \hat{q}_u = 1$ is defined as

$$C = \frac{1}{\sum_{i=1}^{n} k(\|x_i^*\|^2)} \qquad (3)$$

[1] Probability Density Function

2.3. Target candidate

It is assumed that $\{X_i\}i = 1...n_h$ are the loci of normalized target candidate pixels centered at y on current frame. Using weighted function *k(x)* which has similar window sizes similar to target's, the values for target candidate model related to colours *u=1,...m* can be computed as below:

$$\hat{p}_u(y) = C_h \sum_{i=1}^{nh} k(\left\|\frac{y - x_i}{h}\right\|^2)\delta[b(X_i) - u] \qquad (4)$$

where C_h is normalization coefficient (*y* independent), computed by;

$$C_h = \frac{1}{\sum_{i=1}^{nh} k(\left\|\frac{y - x_i}{h}\right\|^2)} \qquad (5)$$

2.4. Bhattacharyya coefficients

In order to compute the similarity between target and a candidate we can use the Bhattacharyya coefficients. Using Bhattacharyya coefficients the similarity will be computed in the form of metric distance between discrete distribution of target and candidate [5] pixels as below;

$$d(y) = \sqrt{1 - \rho[\hat{p}(y), \hat{q}]} \qquad (6)$$

where $\rho[\hat{p}(y), \hat{q}]$ can be found using Equation (7).

$$\hat{p}(y) \equiv \rho[\hat{p}(y), \hat{q}] = \sum_{u=1}^{m} \sqrt{\hat{p}_u \hat{q}_u} \qquad (7)$$

For the ideal similarity the value of (7) approach to one. A complete explanation about Bhattacharyya coefficients can be found in [2].

2.5. Target localization

In order to find the new position of target in current frame, the distance obtained using Equation 6 should reach its minimum value. The tracking process starts from a point (and its vicinity) which it was the target place in previous frame and continue until the mentioned distance reach to predefined minimum (Equation 6) or maximum iteration happens.

3. DEVELOPED MEAN-SHIFT ALGORITHMS

3.1. CBWH tracking algorithm

Using background histogram ($\hat{o}_u$) a transfer function (*Vu*) has developed to decrease the effects of background colours on target model [1]. The related algorithm named *BWH* [2]. It is shown that the performance of BWM algorithm will increase if we use Equation 8 for target model only in the first frame and using Equation 4 for other frames [3]. This algorithm known as *CBWH* [3]. Equation 8 is a transformation that decreases the effect of background colour on target model computed by Equation 9.

[2] Background Weighted Histogram

[3] Corrected Background Weighted Histogram

$$\{v_u = \min(\frac{\hat{o}^*}{\hat{o}_u},1)\}_{u=1...m} \qquad (8)$$

$$\hat{q}'_u = C\acute{v}_u \sum_{i=1}^{n} k(\|x_i^*\|^2)\delta[b(x_i^*)-u] \qquad (9)$$

where

$$C' = \frac{1}{\sum_{i=1}^{n} k(\|x_i^*\|^2)\sum_{u=1}^{m} v_u \delta[b(x_i^*)-u]} \qquad (10)$$

In *CBWH* firstly the target background $\hat{o}_u$ is computed then we make the transfer function v_u using Equation 8.

3.2. Self-Adaptive size tracking algorithm

The size of tracking window in basic *MS* is not target adaptive. In [1] the coloured model is computed within +/-10% of target window. Using Equation 11 the new size of the window is ermined. Here h_{opt} is the window size corresponds to the most probable target selected by Bhattacharyya coefficients.

$$h_{new} = \gamma h_{prew} + (1-\gamma)h_{opt} \qquad (11)$$

The results obtained by this method show that this algorithm works well in the case of targets with decreasing size but fails when the target is being zoomed. Moreover as the value of γ is constant, the rate of window size change is not proportion to target size change. To cope with this problem an algorithm is proposed [4] which adaptively changes the size of track window base on a parameter named *MSIIM*.

4. INTRODUCING NEW COMPARISON PARAMETERS

Evaluation of algorithms is a challenging issue especially when the results are close to each other. Here we are to introduce two parameters which are extracted from the structure of *MS* algorithm. The first one is the mean of confidence coefficient (MCC) shown in Equation 12,

$$MCC = \frac{\sum_{i=1}^{n} CC_i}{n} \qquad (12)$$

Where

$$CC_n = \frac{\sum_{i=1}^{x}(MS(i)_n - MinDist)}{x} \qquad (13)$$

Here CC_n is the confidence coefficient in frame *n* which represents the mean distance between the center of target candidate in frame *n* and process exit criterion (*MinDist*). *MS* in frame *n* is the difference between the centers of target candidate (found *i* time in current frame) and target (found in frame *n* of previous try).

The second parameter, *NV*, is the normalized variance of CC_n with respect to *MCC*, defined as:

$$NV = \frac{\sum_{i=1}^{n} (CC_i - MCC)^2}{n} \qquad (14)$$

If the *MS* distance was smaller than *MinDist* the center of target candidate (correspond to *MS*) will be selected as new target position, else, this position is used in place of target center in previous frame to compute the target position in current frame. The mean of target candidates distances from *MinDist* computed by Equation 14, represents the accuracy of tracking process on that frame. Variance is used to show the reliability of the method. The smaller values indicate on more repeatability of the results. Considering presented parameters and the results of *MS* trackers, we evaluate them as below:

1. Using process iteration to show the accuracy and speed of tracking,

2. Using visual judgment,

3. Using Equations 12 and 14.

Here we compare different *MS* algorithms by tracking the target locale of Figure 1 over test video frames.

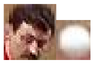

Figure 1. The locale of tracked targets in test videos

Figure 2 shows the comparison of *MS* methods over BALL video test. The target has quick motion while its size changes. It can be seen that the *CBWH* algorithm has the best performance while the other two methods had some fails due to their low accuracies, and the high speed of the ball.

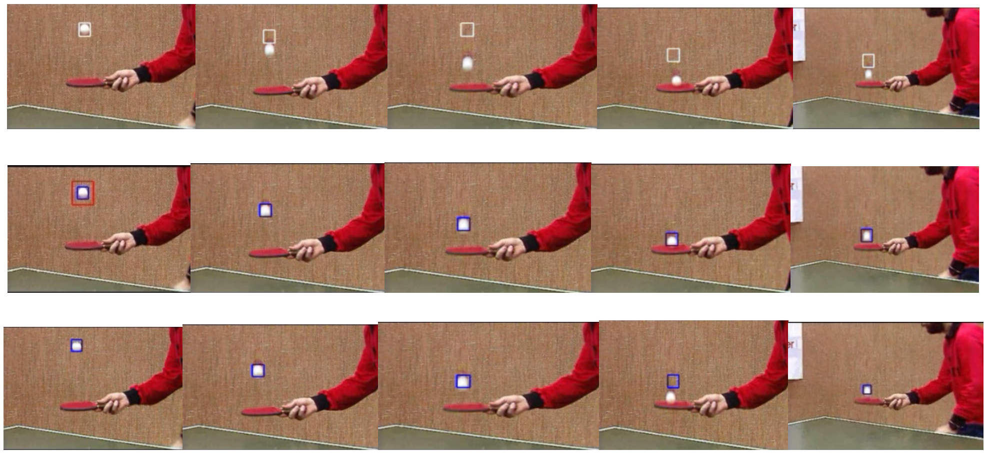

Figure 2. Ball tracking in BALL video test, using (from the top) basic Mean-Shif, CBWH and Self-adaptive size algorithms

Figure 3 shows another test using PLAYER video frames. Here the target is the player head and the purpose is comparing the mentioned algorithms using their average value of iterations. Although the results are similar to each other, but as in *CBWH* the target model made more accurate its *MCC*, *NV* and iteration are better than others'.

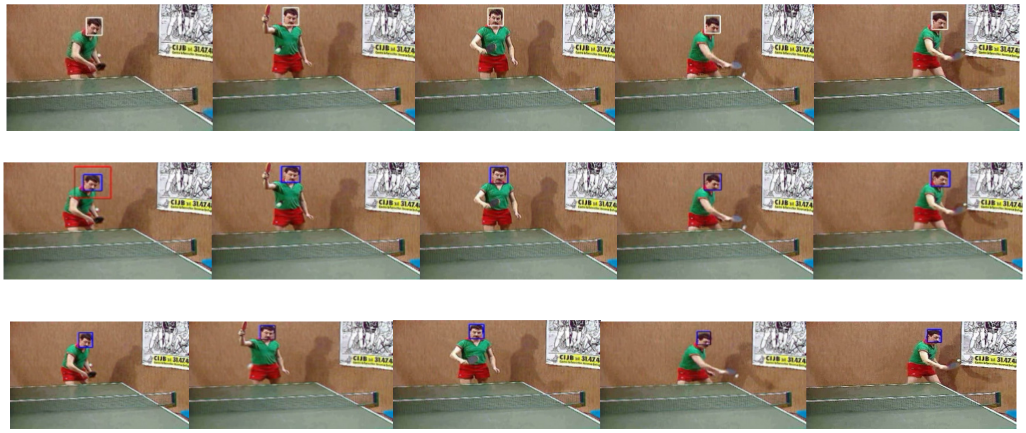

Figure 3. Head tracking in PLAYER video test, using (from the top) basic Mean-Shift, CBWH and Self-adaptive-size algorithms

Table 1 illustrate the stochastic results of the above tests. The values of *MCC*, *NV*, and iteration for each of the three algorithms for both BALL and PLAYER test videos are colleted in this table.

Table 1. Comparison of the Mean-Shift based algorithms.

Video Tests	Algorithms	MCC	NV	Iteration
BALL	MeanShift(std)	0.157247	0.004017	6.3462
	CBWH	0.125175	0.003644	4.2885
	Self-Adapt	0.126628	0.003684	6.4615
PLAYER	MeanShift(std)	0.122522	0.003572	3.7414
	CBWH	0.118467	0.002577	3.2414
	Adapt	0.168233	0.020356	3.5517

As mentioned before, for those results which are similar to vision (e.g. Figure 3) it is difficult to say which algorithm works better. The values in Table 1 help us to compare the algorithms, quantitatively. Here the overall result shows the better performance of *CBWH* algorithm.

5. Conclusion

Number of iteration in each frame and visual observation are basic parameters to compare the performances of Mean-Shift based tracking methods. In some cases such as when the results are high correlated it is hard to compare the algorithms by mentioned parameters. In this paper we presented two new parameters in order to compare the accuracy and reliability of *MS* based trackers. The result of examined video tests show even in the case of closeness of tracker outputs it is possible to have a good assessment regarding algorithms performances using proposed parameters.

REFERENCES

[1] Comaniciu D., Ramesh V. and Meer P.: ‘Kernel-Based Object Tracking’, IEEE Trans. Pattern Anal. Machine Intell., 2003, 25, (2), pp. 564-577.

[2] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, “Multi-camera multi-person tracking for EasyLiving,” in Proc. IEEE Intl. Workshop on Visual Surveillance, Dublin, Ireland, 2000, pp. 3–10.

[3] J. Ning, Lei Zhang, David Zhang and C. Wu, “Robust Mean Shift Tracking with Corrected Background-Weighted Histogram,” to appear in IET Computer Vision.(2011)

[4] Huimin Qian, Yaobin Mao, Jason Geng, and Zhiquan Wang,“Object tracking with self-updating tracking window,” in PAISI,2007, vol. 4430, pp. 82–93.

[5] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, “Multi-camera multi-person tracking for Easy Living,” in Proc. IEEE Intl. Workshop on Visual Surveillance, Dublin, Ireland, 2000, pp. 3–10.

[6] L.Li, "An Efficient Sequential Approach to Tracking Multiple Objects Through Crowds for Real-Time Intelligent CCTV Systems", IEEE transactions on system, man, and cybernetics—part B: cybernetics, vol. 38, no. 5, October 2008.

[7] P.Bai,"Person-Tracking with Occlusion Using Appearance Filters", Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems October 9 - 15, 2006, Beijing, China.

[8] Y.Wu,F.Lian, T.Chang ,"Traffic Monitoring and Vehicle Tracking using Roadside Cameras",2006 IEEE International Conference on Systems, Man, and Cybernetics, Taipei, Taiwan.

[9] P.Vadakkepat ,"Multimodal Approach to Human-Face Detection and Tracking",IEEE tracsaction on industrial electronics, vol. 55,no. 3, March 2008.

[10] K.Zhang ,"Research on the Image Matching and Tracking Algorithm for the End of Infrared Target Tracking",ICALIP2008.

[11] W.Yang, "A Novel Layered Object Tracking Algorithm for Forward-looking Infrared Imagery Based on Mean Shift and Feature Matching",978-1-4244-4520-2/09/$25.00 ©2009 IEEE.

[12] L.Zhang ,"Real Time Mean Shift Tracking Using the GaborWavelet", Proceedings of the 2007
IEEE International Conference on Mechatronics and Automation August 5 - 8, 2007, Harbin, China.

[13] A.Chen, "Mean Shift Tracking Combining SIFT",ICSP2008 Proceedings.

[14] P.Guha"Efficient Occlusion Handling for Multiple Agent Tracking by Reasoning with Surveillance Event Primitives",Proceedings 2nd Joint IEEE International Workshop on VS-PETS, Beijing, October 15-16, 2005.