Computer Engineering and Applications Vol. 02, No. 03, December 2013

Improvement and Comparison of Mean Shift Tracker using Convex Kernel Function and Motion Information

S. B.Chaudhari[1], K.K.Warhade[2], V.M. Wadhai[3], N.K. Choudhari[4]

[1]R.T.M.NagpurUniversity
[2,3] MITCOE Pune
[4]SBCCOE Nagpur
Email: [1]shilpa.wakode@gmail.com

ABSTRAKSI

Algoritma pelacakan harus mampu mendeteksi objek bergerak tertarik dengan bidang pandang dan kemudian melacak dari frame ke frame. Pelacakan algoritma berdasarkan berarti pergeseran kuat dan efisien. Tapi mereka memiliki keterbatasan seperti ketidaktepatan target lokalisasi, objek yang dilacak harus tidak lulus oleh orang lain objek dengan serupa memiliki yaitu oklusi dan cepat benda gerak. Karya ini mengusulkan dan membandingkan algoritma peningkatan adaptif berarti pergeseran dan adaptif berarti pergeseran menggunakan fungsi cembung kernel melalui gerak informasi. Hasil penelitian menunjukkan bahwa kedua metode melacak objek tanpa pelacakan kesalahan. Metode adaptif memberikan perhitungan biaya kurang dan tepat sasaran lokalisasi dan berarti pergeseran menggunakan fungsi cembung kernel menunjukkan hasil yang baik untuk tantangan pelacakan seperti oklusi parsial dan gerak cepat objek yang dihadapi oleh basic berarti pergeseran algoritma.

Kata kunci: Pergeseran Rata – Rata (PR), Pelacakan Objek Berbasis Kernel (POBK)

ABSTRACT

Any tracking algorithm must be able to detect interested moving objects in its field of view and then track it from frame to frame. The tracking algorithms based on mean shift are robust and efficient. But they have limitations like inaccuracy of target localization, object being tracked must not pass by another object with similar features i.e. occlusion and fast object motion. This paper proposes and compares an improved adaptive mean shift algorithm and adaptive mean shift using a convex kernel function through motion information. Experimental results show that both methods track the object without tracking errors. Adaptive method gives less computation cost and proper target localization and Mean shift using convex kernel function shows good results for the tracking challenges like partial occlusion and fast object motion faced by basic Mean shift algorithm.

Keywords: Mean Shift (MS), Kernel Based Object Tracking (KBOT)

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

1. INTRODUCTION

Object tracking is a challenging problem in various video processing and computer vision applications. Adaptability under occlusions is the major challenge in object tracking process [1]. Real time object tracking has many practical applications, both commercial and military, such as visual surveillance, traffic monitoring, vehicle navigation, precision targeting, perceptual user interfaces and artificial intelligence. In real time applications kernel based object tracking (KBOT) [1] is a robust and efficient tracking technique. Comaniciu in [1][2] used kernel based mean shift algorithm to track moving object. There Bhattacharyya coefficient is used as a tool to compare between object models and object candidate and used mean shift algorithm to search the optimum object candidate. As this method was good for object tracking, many researchers use it as a base for advance tracking algorithms. Peng [3] and Comaniciu [4] respectively proposed the automatic selection of bandwidth for mean shift-based object tracking; Peng [5] also propose an updating method of object model in mean shift algorithm. However mean shift algorithm has many flaws like the tracking errors or object lost. Background pixels in object model increase localization error of object tracking. Little background pixels in object model, give good result of object tracking. But in order to let the object contained in object model, some background pixels will certainly get introduced in object model.

To overcome existing problems a robust tracking algorithm need to be developed using multiple features. In this paper two algorithms are proposed and compared viz., an improved adaptive mean shift algorithm and adaptive mean shift using a convex kernel function through motion information of desired video sequences. The robustness of proposed algorithm can be increased by integrating both color feature and motion information, under the MS tracking algorithm framework. This paper is organized as follows. In Section 2 the basic mean shift algorithm is presented. In Section 3 the adaptive implementation of mean shift is discussed and Section 4 contains the proposed algorithm using convex kernel function and motion information. Section 5 and 6 comprises experimental results and concluding remarks respectively.

2. MEAN-SHIFT ALGORITHM

Kernel based MS algorithm approach is broadly classified into two components

viz. target model representation and candidate model representation.

2.1 TARGET MODEL

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online

The target model is represented in its feature space by its probability density function (PDF), which is calculated using kernel density estimation [3] given by

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

Where h is the bandwidth of the kernel and x_i is the center of the d dimensional kernel while n is total number of points in the kernel. Kernel density can be determined with the application of Epanechnikov kernel [3] which is defined as

$$K_{E}=\begin{cases}\frac{1}{2}C_{d}^{-1}(d+2)\left(1-|x|^{2}\right) if\ |x|\leq 1\\ 0 \qquad \text{otherwise}\end{cases} \qquad (2)$$

Where C_d is the volume of the d-dimensional space. The target is selected manually in the first frame and its PDF is calculated by considering its location centered at x_0. To track target in the next frame its PDF in the next frame is calculated at the same location as

$$\hat{q}_{u}=C\sum_{i=1}^{n}K\left(\left|\frac{x_{0}-x_{i}}{h}\right|^{2}\right)\delta[b(x_{i})-u] \qquad (3)$$

2.2 CANDIDATE MODEL

The candidate is the area containing the moving object in the subsequent frames. Candidate model can be described as the probability density distribution of the pixel's feature value in the candidate area centered at y. The PDF of the target candidate is calculated as

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

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

where $$C_h = \frac{1}{\sum_{i=1}^{n_h} K\left[\left|\frac{y-x_i}{h}\right|^2\right]}$$

and $u = 1...m$. Here m is the number of bins used for the calculation of PDF for target representation, h is the bandwidth of the kernel and $x_i$is the center of the d dimensional kernel. While n is total number of points in the kernel and $\delta\left[b\left(x_i\right)-u\right]$ is Kroneckor delta function. $b(x_i)$ is image feature value at spatial location $x_i$and C is the normalization constant. Bhattacharya coefficient is used to derive the similarity or correlation between the target model and target candidate. It is specified in the form of a distance given by

$$d = \sqrt{1-\hat{\rho}(y)} \quad (5)$$

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

The term $\hat{\rho}(y)$ is referred as Bhattacharya coefficient. New target location y_1 in current frame is found by iteratively proceeding towards the maxima in the neighborhood. The new target location y_1 is obtained by recursively traveling from its initial location y_0 using following relation, where w_i are the respective weights

$$y_1 = \frac{\sum_{i=1}^{n_h} x_i w_i}{\sum_{i=1}^{n_h} w_i} \quad (6)$$

3. AN ADAPTIVE IMPLEMENTATION OF MEAN-SHIFT ALGORITHM

Any object tracking algorithm process image sequence. If tracking algorithms does not have an adaptive stop threshold in searching the target procedure, then even though they give proper target localization their computation time is much more. Because of effect of limit word-length, the program may run into endless cycle. If they do not have an adaptive updating principle for

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online

target model then tracking error occurs in case of camera motion, target partial occlusions, and target scale variations. In such cases the difference between target model and true target gets more and more, which ultimately increases searching time or lose tracking target. Therefore, an algorithm is implemented with improved mean shift approach in order to improve robustness and reduce computational time of the object tracking algorithm.

The computational speed of KBOT algorithm mainly depends on the number of iterations per frame required for searching the target candidates. So in order to improve the tracking velocity, number of iterations should be reduced. Thus according to tracking precision and velocity a reasonable searching stop threshold value should be set. Let *E is* set as searching stop threshold value. When the distance between the new target candidate and the previous target candidate is smaller than *E*, the iteration for searching newer target candidate stops. For example, when tracking precision is 0.5 pixels then the *E* can be set as 0.71. The number of iterations can be set as stop threshold value. To avoid endless cycle in real-time object tracking, maximum number of iterations can be set to *N*max. If the target model is too small it will impact the robustness of the algorithm which based on gradient of intensity information. Target model must update in the implementation procedure of a tracking algorithm [6]. But its updating should have a reasonable principle which is neither extreme sensitive nor extreme slow. If updating principle is extreme sensitive, then some model updates which are not necessary will waste lots of computational time for computing feature representations. There are two situations. One is that it has only minute difference between new model and previous model. The other is that the difference of features are huge, there the new model may be affected by some factors such as partial occlusions and background clutter. If the target model is updated in this situation the next tracking result deviation will be huge even tracking is failure. Therefore the metric distance *D* derived from Bhattacharyya coefficient [7] is set as a standard for updating target model. The $T_a$is denoted as an upper limit and $T_b$as a lower limit. $T_a$is the distance *D* value when the difference between target model and target candidate is *a*. $T_b$is also the distance *D* value when the difference is *b*.

The complete adaptive implementation for the improved kernel-based object tracking algorithm is presented below.

Given: The target model is $\{\hat{q}_u\}_{u=1\ldots m,}$and its location is $\mathbf{y_0}$in the previous frame. Initialize the location of the target in the current frame with $\mathbf{y_0}$, compute $\{p_u(y_0)\}_{u=1\ldots m,}$ and evaluate

$$\rho = [p(y_0,q)] = \sum_{u-1}^{m} \sqrt{p_u}(y_0)q_u] \qquad (7)$$

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

1. Derive the weights
2. Find the next location of the target candidate
3. Compute $\{p_u(y_1)\}_{u=1..m}$ and evaluate Bhattacharya distance between $p_u(y_1)$ and q
4. If $|y_1 - y_0| < E$ then stop and go to step 7
5. If The maximum iteration number (in the current frame) ≥ Nmax, Then go to Step 7, Otherwise set $\mathbf{y_0} \leftarrow \mathbf{y_1}$ and go to Step 2;
6. If $D \geq T_b$ and $D \leq T_a$, Then $\{\hat{q}_u\}_{u=1...m} \leftarrow \{p_u(y_1)\}_{u=1...m}$ i.e. update target model
7. If no new images then stop otherwise read new image and go to step 1

4. ADAPTIVE MEAN-SHIFT ALGORITHMUSING CONVEX KERNEL FUNCTION AND MOTION INFORMATION

4.1 USING CONVEX KERNEL FUN

One of the existing problems, in the process of tracking, is the partial occlusion, where a part of object's body is covered by another object's body or by some structure in the scene. So the object might be lost, because the likeness between both the object and its candidate. The original MS tracking algorithm fails in case of partial occlusion as the used kernel function k(.) is vulnerable. In several available researches, equation (2) is used as the kernel function, the central and marginal pixels weight are about 1 and 0.5, respectively. As a consequence, this problem has been led to tracking failure, because the marginal pixels weight is high, while partial occlusion occurs, in a particular manner. Here a convex kernel function [8] is used which is based upon Gaussian function as defined in equation (8).

$$k(x,\hat{x},y,\hat{y}) = A \exp\left|-\frac{(x-\hat{x})^2}{h_y} - \frac{(y-\hat{y})^2}{h_x}\right| \qquad (8)$$

Where h_x and h_y are given as the rectangle size surrounded the object, $\hat{x}$ and $\hat{y}$ are also given as the rectangle center and A is finally given as a weight matrix. This kernel allocates a higher accurate weight to the pixels, which are close to the center of the rectangle (is about 1) and a smaller weight to the pixels, which are far from its center (is about 10^{-16}). It means that, practically, the marginal pixels have not much effect on the tracker performance and are disregarded. Values of (h_x, h_y) and ($\hat{x}$, $\widehat{y}$) depend on the size and also the center of rectangle surrounded the object, respectively. If the positions ofh_x and h_y in (8) are changed, then obtained results will be different and disappointing.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online

4.2 USING MOTION INFORMATION

4.2.1. The reasons of combining the color feature and the motion information

To overcome existing problems either one robust tracking algorithm should be developed or multiple features should be considered. If some visual features are only used for the object introduction, then the process of tracking could be easy to realize. But a single feature might not introduce the desired objectappropriately. In color-based MS tracking algorithm color feature is easy to be extracted, but it includes some similarly colored background areas that distract tracking. Also, it is weak against the illumination variations and occlusions. On the other hand, the motion information obtained from the difference between successive frames holds all the moving objects, even those areas are out of tracking. Hence combining the color feature with motion information will eliminate uninterested regions.

4.2.2. Extracting the motion information

Temporal differencing is generally used to obtain the movement of objectin the video sequences. The resulted image is a binary mask which divides the current frame into two parts. The first part contains light regions, which are the points of the current frame having lightness more than background. The probability of object's existence in these regions is high. The second part is darkness regions where the probability of object's non-existence is high.Fig. 1 shows the result obtained after applying temporal differencing method to successive frames in a video scene. The producedimage is not suitable in object tracking as there are a number of holes on it and isolating the points into the background. These problems are more observed into the complex sequences. Due to these problems the stationary regions are considered as moving regions. In such cases, a circular structure element within a 5-pixel radius can be designed for eliminating these holes and isolating points. So, the whole of holes fill and the resulted mask will be ready to be employed into the kernel of the MS tracking algorithm.

Figure 1.a)current frame   b) result of temporal differencing

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

In order to reduce the computational time, the process of movement detection can be limited to the rectangle which would surround the desired object. Then suitable binary mask can be extracted using temporal differencing. In the first framethe desired object is marked by a rectangle and in succeeding frames it is considered as background’s color model. Difference between the rectangles, which is surrounded the object in the first frame and the rectangle surrounding the object in the next frames, should be taken in order to realize a suitable mask.The resulted binary mask B_m from nth frame can be given as

$$B_m(x_i) = \begin{cases} 1 & if\ x_i \epsilon f \\ 0 & if x_i \epsilon b \end{cases} \quad (9)$$

where the parameters *f* and *b* indicate the foreground and the background respectively.

4.2.3. Applying the motion information

The obtained binary mask is multiplied by the improved kernel function before using into MS tracking algorithm. A new kernel is developed by considering weight of the probability of object’s existences. So in calculation of both the object and candidate models, only lightness regions need to be examined. Thus in this method the most important thing is to acquire a binary mask in case of motion information extraction and also combining the result with (8). In this process equation (10)–(12) are all achieved.

$$K_{new}(\|x_i^*\|^2) = K(\|x_i^*\|^2) \times B_m(x_i) \quad (10)$$

And also for candidate model could be given as

$$K_{new}\left(\left\|\frac{y-x_i}{h}\right\|^2\right) = K\left(\left\|\frac{y-x_i}{h}\right\|^2\right) \times B_m(x_i) \quad (11)$$

Moreover, in calculation of the new location, w_i has been acquired by the following equation

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

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online

4.3 The Proposed Approach Flowchart

The flowchart of the proposed approach is illustrated in Fig 2

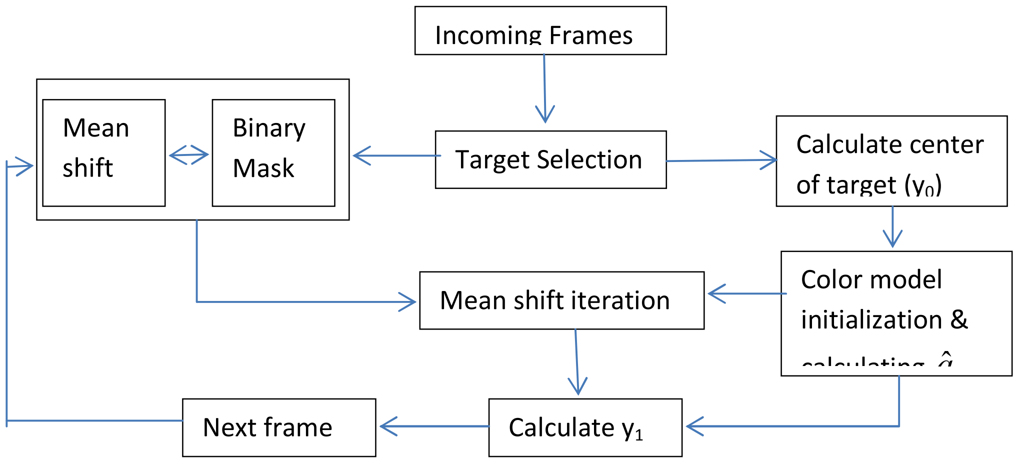

Figure 2 Flowchart of proposed algorithm

As per the proposed approach, target is selected manually in first frame. Then temporal differencing is used to acquire binary mask. The obtained binary mask is multiplied by improved MS kernel. The location of target in current frame is set at y_0. Color model is initialized and PDF of target in next frame ($\hat{q}_u$) is estimated. Then MS algorithm is applied and new target location at y_1 is obtained. Same procedure is repeated till last frame.

5. EXPERIMANTAL RESULTS

Figure 3, 4 and 5 are the results of basic MS algorithm, adaptive implementation of MS and the proposed algorithm using convex kernel function respectively. The experiments are carried out on our own generated videos and videos from movies. Programming is done in MATLAB.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

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

Figure 3 Results of basic MS algorithm for Frame Sequence: (a)2, (b)10, (c)46, (d)72

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

Figure 4 Results of adaptive MS algorithm for Frame Sequence: (a)2, (b)10, (c)46, (d)72

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

Figure 5 Results of proposed algorithm for Frame Sequence: (a)2, (b)10, (c)45 (d)72

It is observed that basic MS algorithm fails to track the object after frame no. 46 while the adaptive MS tracks the object properly till last frame except track is lost for few frames near frame number 46. But the proposed method tracks the object successfully till the last frame without any tracking error.

Figure 6 shows that the partial occlusion problem can be eliminated with the proposed approach using convex kernel function and motion information.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online

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

Figure 6. Results of proposed method for partial occlusion problem for Frame Sequence: (a)1, (b)33, (c)50 (d)7

6. CONCLUSION

In this paper, MS algorithm using a convex kernel function through motion information of desired video sequences is proposed. Also the performance of proposed algorithm is compared with adaptive mean shift algorithm. Adaptive method gives less computation time and proper target localization but fails for partial occlusion and fast object motion. While the proposed method using convex kernel function shows good results for all the tracking challenges faced by basic MS algorithm. Also in future combining color feature with the motion information will improve tracking even for cluttered background and sudden light changes.

REFERENCES

[1] AlperYilmaz, Omar Javed, and Mubarak Shah, "Object Tracking: A Survey," ACM Computing Surveys, Vo1.38, No.4, Article 13, December 2006. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-Based Object Tracking”, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25, No. 5, 2003 pp. 564-575.

[3] N. Peng “Automatic Selection of Kernel-Bandwidth for Mean-Shift Object Tracking”, Journal of Software, Vol. 16, No.9, 2005, pp.1542-1550..

[4] D. Comaniciu “An Algorithm for Data Driven Bandwidth Selection”, in IEEE Trans. On Pattern Analysis and machine intelligence, Vol. .25, No.2, 2003, pp.281-288.

[5] N. Peng, “Mean Shift Blob Tracking with Kernel Histogram Filtering and Hypothesis Testing”, Pattern Recognition Letters, Vol. .26, No.5, 2005, pp. 605-614.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

[6] M. Weng, H. Mingyi, and Y. Zhang, "An Adaptive Implementation of the Kernel-Based Object Tracking Method", IEEE International Conference on Innovative Computing, Information and Control (ICICIC'06), 2006.

[7] F. Aherne, N. Thacker, and P. Rockett, "The Bhattacharyya Metric as an Absolute Similarity Measure for Frequency Coded Data," Kybernetika, 1998, vol. 34, no. 4, pp. 363-368.

[8] A.H. Mazinan , A. Amir-Latifi, "Improvement of mean shift tracking performance using a convexkernel function and extractingmotion formation", Elsevier Journal for Computer and Electrical Engineering 2012, Article cited in Press: http://dx.doi.org/10.1016/j.compeleceng.2012.06.009

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online