Adaptive Bandwidth Object Tracking Using Sparse Approximation

Chengjun Xie[1,2] Jieqing Tan[1]
1. School of Computer & Information
Hefei University of Technology
Hefei, China
E-mail: cjxie@iim.ac.cn

Jie Zhang[2] Yingqiao Bu[3] Jin Xie[2]
2. Institute of Intelligent Machines
Chinese Academy of Science
Hefei, China
3. Electronic Engineering Institute of PLA

***Abstract*— In this paper, a novel adaptive bandwidth object tracking using sparse approximation is proposed in mean shift framework. In this framework, occlusion, illumination and other challenging issues are addressed seamlessly in real time through bandwidth approximation. To find the tracking target at a new frame, a two step method is used iteratively to find the most probable target candidate position. The first step is to find an appropriate bandwidth which best describes the region range of the object using sparse representation, and the second step is to find the target position using mean shift object tracking framework which has been given the appropriate bandwidth in the first step. In addition, we improve the convergence properties of the mean-shift by approximating the region range of target in the iterative update mechanism. The proposed approach shows excellent performance in comparison with originally proposed trackers.**

Keywords-mean shift; object tracking; sparse approximation; bandwidth

I. INTRODUCTION

The goal of object tracking is to estimate the locations and motion parameters of a target in an image sequence given the initialized position in the first frame. There are two different approaches to address the problem of object tracking relying on a variety of mathematical tools. Discriminative methods formulate the tracking as a classification problem [4]. The generative methods represent the target observations as an appearance model [9]. The tracking problem is formulated as searching for the region within the highest probability generated from the appearance model [11]. Among the above various tracking algorithms, mean shift tracking algorithm is the most typical one which has recently become popular due to their simplicity and efficiency in the generative methods. It was originally developed by Fukunaga and Hostetler [3] for data analysis, and later Comaniciu and Meer successfully [1, 2] introduced it to the field of object tracking. Currently, many of mean shift tracking algorithms are based on their work. The key idea of Mean Shift is an iterative kernel-based deterministic procedure which converges to a local maximum of the measurement function with certain assumptions on the kernel behaviors [17]. Since mean shift tracking algorithm may lead to misdirected mean vectors or undesirable local optima when an improper bandwidth is used, a key to implementing mean shift tracking algorithm is to find an optimal bandwidth for the kernel. Although kernel bandwidth is a crucial parameter for the mean-shift algorithm, there is currently no sound mechanism for choosing this bandwidth within the framework. The traditional mean shift algorithm is generally based on the initial size of the target with fixed bandwidth [10,11] which can lead to smoothing density estimation function when the objects move fast and undergo illumination. As a result, the target tracking results are finally affected. Fixed bandwidth often can lead to the drift of the target [6] when there is significant scale changes in the target, especially target size increases beyond the fixed bandwidth. By artificially changing the bandwidth parameter of the target to test the new window size, one selects the maximum Bhattacharyya coefficient corresponding to the bandwidth for the new one [8, 11, 12].It is able to obtain better results when the target size is gradually reduced. However, it is difficult to expand for the bandwidth when the target size increases. In this case, the bandwidth often becomes smaller and smaller because of the similarity coefficient measure based on Bhattacharyya often gets the local maximum in smaller tracking window [13]. The above problem we address in this paper is selecting the bandwidth of the mean-shift kernel, which directly determines the size of the window. We show how to combine a well-developed theory of bandwidth selection based on sparse approximation with the mean-shift algorithm.

Recently, the sparse representation has been utilized in many areas [14, 15].Reference [16]successfully develops a robust visual tracking framework by casting the tracking problem as finding a sparse approximation in a template subspace. Sparse representation codes a target template y over a dictionary Φ such that $y \approx \Phi\alpha$ and α is a sparse vector. The l0-norm counts the number of non-zeros in α . The l1-minimization is widely employed because of the NP problem of the combinatorial l_0–min*imization* in sparse coding: $\min_{\alpha} \|\alpha\|_1$ s.t. $\|y-\Phi\alpha\|_2^2 \le \varepsilon$, where ε is a small constant. In our paper, we extend the traditional mean shift object tracking framework to an adaptive bandwidth one by sparse approximation. Firstly, an adaptive bandwidth size and target position is approximated in the first step using sparse representation. Secondly, the mean shift step is implemented for seeking the best position in each iteration process with an adaptive bandwidth. The tracking effectiveness and efficiency are verified by experiments.

The rest of the paper is organized as follows. Section II introduces the classical mean shift algorithm. Section III extends the mean shift object tracking algorithm presented in

978-1-4673-0089-6/12/$26.00 ©2012 IEEE

[2] and an adaptive bandwidth mean shift is proposed. Extensive experiments are performed to test the proposed algorithm in comparison with state-of-the-art schemes in Section IV. Section V draws the conclusion.

II. MEAN SHIFT OBJECT TRACKING

In [2], both the target model and target candidate are characterized by a kernel-based histogram vector, which is a kind of image-probability distribution. Most of existing target tracking schemes use the color histogram to represent the target. A kernel-based histogram can be denoted as follows:

$$\vec{q} = [q_u]_{u=1,\dots,m}\ ,$$
$$q_u = \frac{1}{C_h}\sum_{i=1}^{n} Kernel\left(\left\|\frac{X_i - c^k}{h}\right\|^2\right)\delta[b(X_i),u], \quad (1)$$

where q is the target model vector, q_u is the probability of the u^{th} element of q_u, δ is the Kronecker delta function, $b(X_i)$ associates the pixel X_i to the histogram bin, and $Kernel$ is a spatially weighting function centered c^k. Constant C_h is a normalization term which makes $\sum_u^m q_u = 1$. $\{X_i\}_{i=1,\dots,n}$ are the normalized pixels positions in the target region, which have n pixels. Similarly, the target candidate model $\vec{p}$ corresponding to the candidate region is given by

$$\vec{p}(c^k) = [p_u]_{u=1,\dots,m}\ ,$$
$$p_u = \frac{1}{C_h}\sum_{i=1}^{n} Kernel\left(\left\|\frac{X_i - c^k}{h}\right\|^2\right)\delta[b(X_i),u] \quad , (2)$$

where the center of candidate region in sequent frame k is denoted as c^k.

In order to calculate the likelihood of the target model and the candidate model, [2] is derived from second-order Taylor expansion of Bhattacharyya coefficient, which is defined as follows:

$$\rho[\vec{p}(c^k),\vec{q}] = \sum_{u=1}^{m}\sqrt{p_u(c^k)q_u} \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{p_u(c^k)q_u} + \frac{1}{2}C_h\sum_{i=1}^{n} w_i Kernel\left(\left\|\frac{X_i - c^k}{h}\right\|^2\right) \quad , (3)$$

Where

$$w_i = \sum_{u=1}^{m}\sqrt{\frac{p_u(c^k)}{q_u}}\delta[b(X_i),u] \quad , \quad (4)$$

The distance between $\vec{p}(c^k)$ and $\vec{q}$ is then defined as

$$d[\vec{p}(c^k),\vec{q}] = \sqrt{1-\rho[\vec{p}(c^k),\vec{q}]} \quad , \quad (5)$$

The kernel-based method realizes target model tracking through minimizing the distance $d[\vec{p}(c^k),\vec{q}]$, as in

$$\Delta c^* = \arg\min_{k=1,\dots,t}(d[\vec{p}(c^k),\vec{q}]) \quad (6)$$

In the mean shift iteration, the estimated target moves from c^k to a new position Δc^k, which is defined as:

$$\Delta c^k = \frac{\sum_{i=1}^{n} X_i w_i Kernel\left(\left\|\frac{X_i - c^k}{h}\right\|^2\right)}{\sum_{i=1}^{n} w_i Kernel\left(\left\|\frac{X_i - c^k}{h}\right\|^2\right)}. \quad (7)$$

By using (7), the mean shift tracking algorithm finds in the new frame the most similar region to the object. From (7) it can be observed that two key parameters in the mean shift tracking algorithm are the bandwidth h and the target position c^k. In this paper we will focus on the analysis of h and target position using sparse approximation, and then bandwidth h and the target position c^k adaptive mean shift tracking algorithm can be developed.

III. THE ADAPTIVE BANDWIDTH MEAN SHIFT TRACKING ALGORITHM WITH THE SPARSE APPROXIMATION

A. Bandwidth approximation by Sparse Model

In given an image of the current frame, artificially given target template set $\Phi = [\Phi_1 \dots \Phi_n] \in R^{d\times n} (d >> n)$ containing n target templates such that each template $\Phi_i \in R^d$,tracking results $y_i \in R^d$ is approximated by the linear span of Φ,

$$y \approx \Phi\alpha \quad , (8)$$

where $\alpha = (\alpha_1 \dots \alpha_n)^{\neg} \in R^n$ is target coefficient vector. Since target region are corrupted by noise or partially occluded in variable scenarios, (8) is rewritten as

$$y = \Phi\alpha + \varepsilon \quad , (9)$$

Where ε is the nonzero error vector. The target coefficient vector α can be computed by optimizing the l_1 regularized least square problem, which typically provides a sparse solution [7]:

$$\alpha^* = \arg\min_a \|y - \Phi\alpha - \varepsilon\| + \lambda\|\alpha'\|_1 \quad , (10)$$

where the parameter λ controls the sparsity of both coefficient vector and noise.

To estimate the bandwidth h and the target position c^k in Mean Shift framework, we manually select target templates as the sparse set of features that are most discriminative in separating the target from background in the first step. Then position of each sample is estimated with sparse representation using the generative likelihood. The details of the algorithm are shown in Algorithm 1.

Algorithm 1. The estimation of adaptive bandwidth and target position using sparse approximation

1. Initialize: manually construct n target samples $\{\Phi \in R^{d\times n}\}$, where Φ is the sample feature matrix, generate T candidate samples in frames K and T=20, $c^k=0$, $\Delta c_k^*=0$.
2. Input: a newly chosen tracking target y_i.
3. for $i=1:T$
4. $\alpha_i^* \leftarrow$ solution of (10).
5. $\gamma_i = \left\| y_i - \Phi\alpha_i^* \right\|_2$.
6. $\Delta c_k^* \leftarrow \gamma_i(y)$.
7. $c_i^k = \Delta c_k^* + c_i^k$.
8. end for.
9. $c^k \leftarrow \frac{c_i^k}{T}, h \leftarrow \arg\min_{i=1,\dots,T} \gamma_i(y)$.
10. Output: c^k, h.

B. *The Adaptive Bandwidth Mean Shift tracking algorithm*

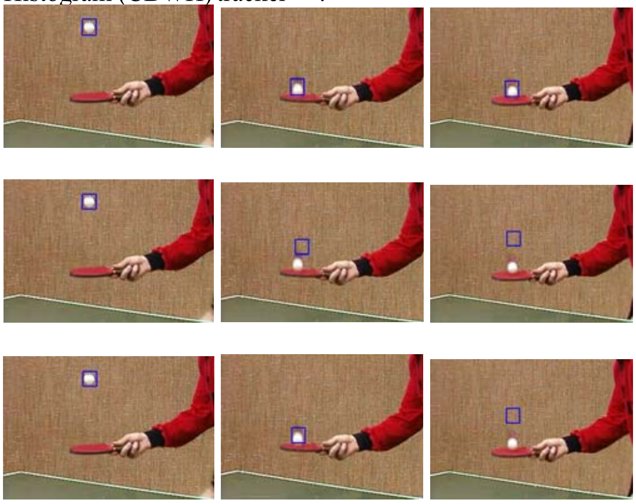
In mean shift tracking, the Algorithm 1 is firstly implemented to estimate bandwidth and target position range, and then the approximate bandwidth and target range is embedded into mean shift tracking framework to improve the convergence properties of the mean-shift and get better tracking result. The whole tracking algorithm is summarized as follows:

Algorithm 2. The adaptive bandwidth mean shift tracking algorithm

1. Initialize the iteration number $m \leftarrow 0$.
2. Input: the target model q and its location in the previous frame Δc_0^k.
3. In the current frame, the algorithm 1 is implemented to estimate bandwidth h and target position c^k, and then calculate the distribution of the target candidate model $\vec{p}(c^k)$.
4. Calculate the weights $\{w_i\}_{i=1,\dots,k}$ using (4).
5. Calculate the new location Δc^k of the target candidate using (7).
6. $m \leftarrow m+1, d \leftarrow \left\| \Delta c^k - \Delta c_0^k \right\|, \Delta c_0^k \leftarrow \Delta c^k$. Set the threshold η and the maximum iteration number N.
7. *If* $d < \eta$ or $m \geq N$, the iteration stops and go to step 8, otherwise go to step 4.
8. Update the target samples Φ with tracking results.
9. Update the target model, $\vec{q} = \vec{q} + \vec{p}(c^k) * \exp(-\arg\min(\gamma_i))$.
10. Load the next frame as the current frame and go to step 1.

IV. Experimental Results

In this section, the proposed tracking algorithm is evaluated using three challenging sequences. The method is compared with two representative tracking methods named Mean Shift (MS) tracker [2], Corrected Background-Weighted Histogram (CBWH) tracker [5].



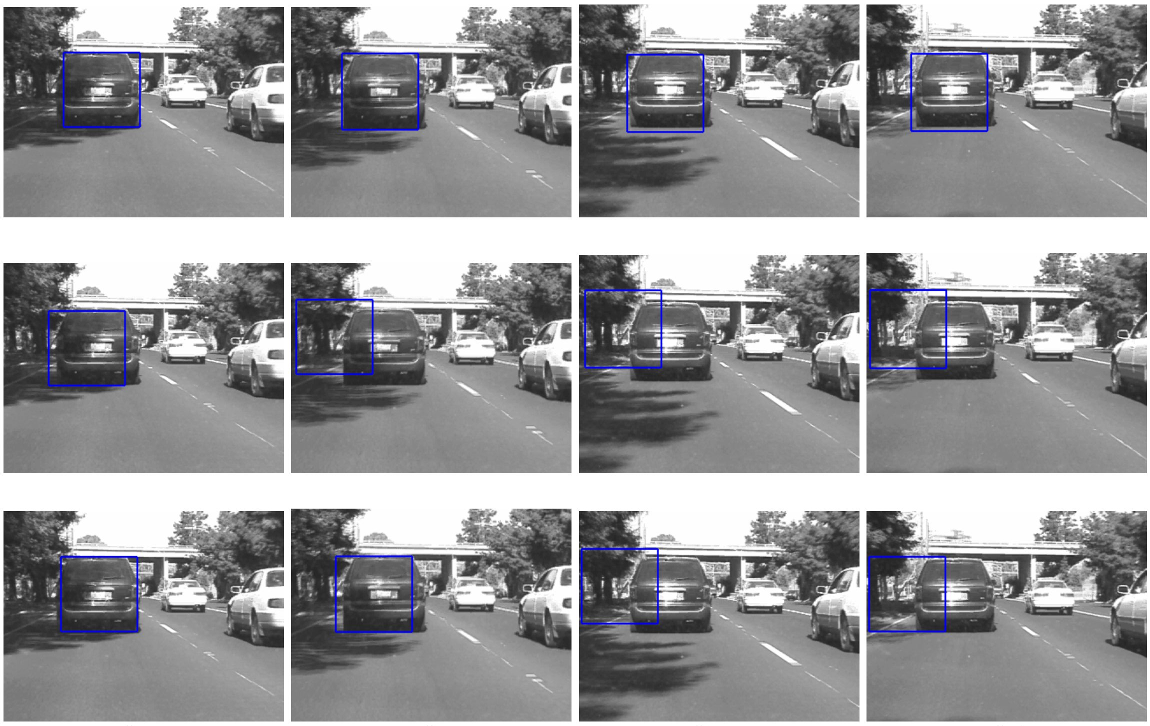
#2 #11 #26

Fig. 1. Results from our tracker, MS tracker, and CBWH tracker are respectively given in the first, second, and third row.

The first experiment is on the benchmark ping-pang ball video sequence with 52 frames, which was used in [2]. The tracking target is the ball that moves quickly. Some samples of the final tracking results are demonstrated in Fig.1, in which three representative frames of the video sequences are shown. Our proposed tracker, MS tracker and CBWH tracker are respectively shown in rows 1,2and 3.From Fig. 1, we find that our tracker is able to track the object when the target ball moves quickly. The MS tracker fails to track the target from 11-th frame and 26-th frame and the CBWH tracker also starts to lose the target in 26-th frame.

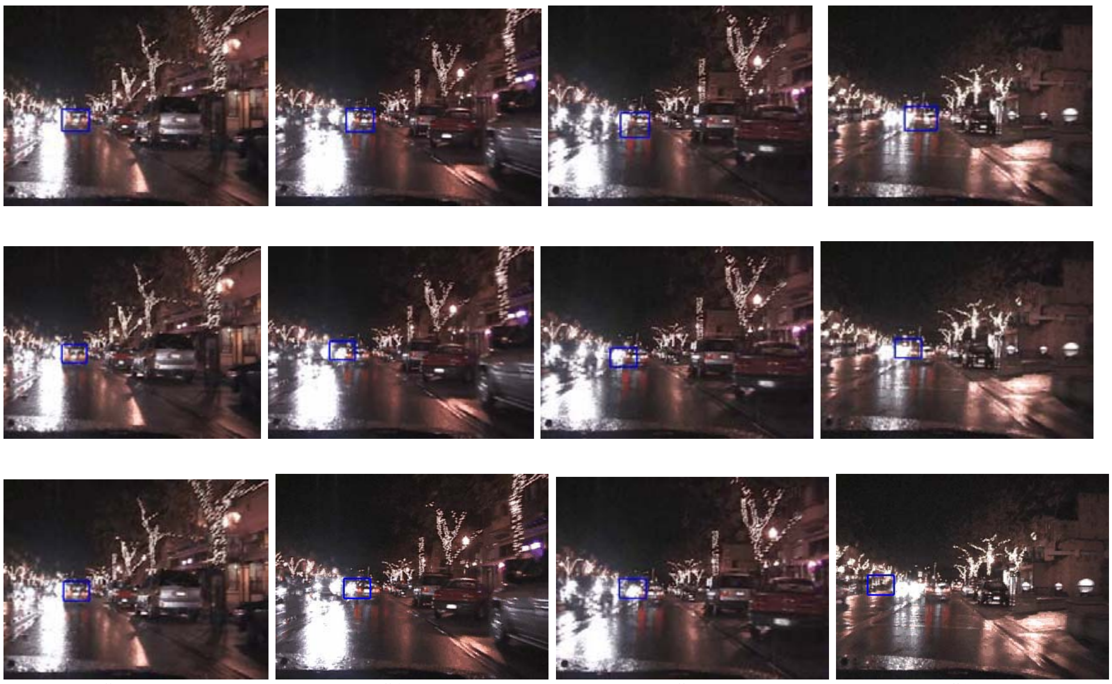
The second sequence with frame 3, 7, 13, and 24 is vehicle video sequence, which drastic illumination changes occur as it passes under trees. The MS tracker starts to show some drifting from 7-th frame and the CBWH also starts to have some drift problem from 13-th frame because of the drastic illumination changes. Our tracker can successfully track the target in the above sequences.

In the third sequence, the vehicle was driven in a very dark environment in which this sequence has low resolution and poor contrast. The 2, 42, 58, and 115 frames are presented in Fig. 3. Our proposed tracker, MS tracker, and CBWH tracker are respectively shown in rows 1, 2, and 3.From Fig. 3, we find that the CBWH tracker starts show some drifting (the 42-th frame and the 58-th frame) and finally loses the target (the 115-th frame) .Although the MS tracker can roughly capture the target before, start to have some drift problem from 58-th frame because of the low resolution and poor contrast in dark environment. Our method performs consistently better results compared with the other tracker.



#3 #7 #13 #24

Fig. 2. The results of our proposed tracker, MS tracker, and CBWH tracker are respectively shown in rows 1, 2, and 3.



#2 #42 #58 #115

Fig. 3. The tracking results of the car sequence in a dark environment: our proposed tracker, MS tracker, CBWH tracker

V. CONCLUSION

Since the bandwidth and the target position range are very important to improve the adaptability of visual object tracking in mean shift framework, in this paper, we propose using a sparse representation to estimate them for robust tracking by exploiting the L1-norm minimization in sample space. Experimental results compared to other representative methods demonstrate the effectiveness of the proposed method.

ACKNOWLEDGMENT

This work was supported by the NSFC-Guangdong Joint Foundation Key Project under Grant (No. U1135003).

REFERENCES

[1] D.Comaniciu, V. Ramesh and P. Meer, "Real- Time Tracking of Non- Rigid Objects using Mean Shift, " IEEE Computer Vision and Pattern Recognition, Vol II, 2000, pp. 142- 149.

[2] D.Comaniciu, V. Ramesh and P. Meer, "Kernel-based object tracking, IEEE Trans," Patt. Anal. Mach. Intell. 25(5) (2003) 564–575.

[3] 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.

[4] Babenko, B., Yang, M.H., Belongie, S, "Visual tracking with online multiple instances learning," CVPR (2009).

[5] 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).

[6] Penning Song, Yang Jie, Liu Zhi, et al. "Automatic selection of kernel-bandwidth for mean-shift object tracking", Journal of Software, 2005, 16(9):1542-1550.

[7] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. "Robust face recognition via sparse representation," PAMI, 31(2):210–227, 2009.

[8] Comaniciu D,Ramesh V. "Mean shift and optimal prediction for efficient object tracking" IEEE International Conference on Image Processing[C] ,2000 ,3 :70-73.

[9] Matthews, I., Baker, S. "Active appearance models revisited." IJCV 60 (2004) 135–164.

[10] Yilmaz A, Shafique K, Shah M. "Target tracking in airborne forward looking infrared imagery." Image and Vision Computing, 2003, 21(7):623-635.

[11] Comaniciu, D., Meer, P. "Mean shift: a robust approach toward feature space analysis." PAMI 24 (2002) 603–619.

[12] Charay Lerdsudwichai, Mohamed Abdel-Mottaleb, A-Nasser Ansari. "Tracking multiple people with recovery from partial and total occlusion." Pattern Recognition, 2005, 38(7):1059-1070.

[13] Collins R T. "Mean shift blob tracking through scale space." IEEE International Conference on Computer Vision and Pattern Recognition[C]. 2003(234-240).

[14] Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A., "Discriminative learned dictionaries for local image analysis."CVPR (2008).

[15] Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y, "Robust face recognition via sparse representation."PAMI 31 (2009) 210–227.

[16] Mei, X., Ling, H. "Robust visual tracking using l1 minimization." ICCV (2009).

[17] Jifeng Ning, Lei Zhang, David Zhang, Chengke Wu, "Robust Object Tracking using Joint Color-Texture Histogram,"International Journal of Pattern Recognition and Artificial Intelligence, 23(9): 1245-1263, 2009.