OBJECT TRACKING USING STRUCTURE-AWARE BINARY FEATURES

Haoyu Ren, Ze-Nian Li

Vision and Media Lab
School of Computing Science
Simon Fraser University
8888 University Drive
Vancouver, BC, CA
{hra15, li}@sfu.ca

ABSTRACT

Object tracking is one of the most important components in numerous applications of computer vision. In this paper, the target is represented by a series of binary patterns, where each binary pattern consists of several rectangle pairs in variable size and location. As complementary to traditional binary descriptors, these patterns are extracted in both the intensity domain and the gradient domain. In the tracking process, the RealAdaBoost algorithm is adopted frame by frame to select the meaningful patterns while considering the discriminative ability and the robustness. This is achieved by a penalty term based on the classification margin and structural diversity. As a result, the features good at describing the target and robust to noises will be selected. Experimental results on 10 challenging video sequences demonstrate that the tracking accuracy is significantly improved compared to traditional binary descriptors. It also achieves competitive results with the commonly-used algorithms.

***Index Terms*—** Object tracking, Binary pattern, RealAdaBoost, Structural diversity

1. INTRODUCTION

Object tracking is one of the most important tasks in computer vision. It is widely used in video surveillance, human computer interaction, and medical imaging. The goal of object tracking is to estimate the states of the target in the subsequent frames based on an initialized state. This task is difficult because there are numerous factors affecting the performance, such as illumination variation, occlusion, as well as background clutters. All these factors will increase the difficulty of the classification between the target object and surrounding backgrounds. To solve this issue, how to represent the target object by meaningful descriptors is the key step.

In consideration of the efficiency, binary descriptors such as Haar or Local Binary Pattern (LBP) are the most commonly-used descriptors in object tracking. Unfortunately, the discriminative ability of traditional binary descriptor is insufficient when dealing with objects with complicate appearance or occlusion, such as multi-view pedestrians or occluded faces. To solve this issue, a number of extensions based on traditional binary descriptors have been proposed. Some of them focus on the definition of the location where the gray value measurement is taken [1][2]. But the improvement of the discriminative ability is still limited because these patterns are artificially designed, and using intensity information might not be sufficient to solve complicated object tracking tasks. Others focus on the combination of multiple descriptors [3][4], which also decreases the efficiency. In addition, using multiple descriptors will increase the risk of low robustness. The descriptors succeed in one scene might fail in another scene due to the appearance and illumination variance.

This paper presents a method with the following contributions. First, we describe the target object by the proposed Boosted Binary Pattern (BBP). BBP reflects the local structural information by rectangle pairs in variable number, size and location. These patterns are applied on both the intensity domain and the gradient domain, so that they could well capture the key object characteristic. In addition, we propose a framework for online selecting the BBP descriptors to describe the target. The confidence for each descriptor is calculated based on the cumulative score in 3 consecutive frames, while considering both the discriminative ability and the robustness. As a result, confident descriptors will be selected, so that the constructed tracking system will be effective and efficient in ever-changing and cluttered scenes. We evaluate our system using 10 public challenging video sequences. The experimental results show that the proposed descriptor has significantly better performance in comparison with traditional binary descriptors and conventional tracking methods.

2. RELATED WORK

In general, the object tracking system consists of two modules, object representation and model construction. For object representation, many methods have been developed. The binary descriptors based on intensity templates are widely used. Grabner et al. [5] trained a real-time tracking system by adopting Haar descriptor in online AdaBoost learning. Ning et al. [1] used the joint color-texture histogram to represent a target and then applied it to the mean shift framework. Mei and Ling [6] integrated the sparse representation into binary template representation. Several improvements based on this work have been proposed recently [7][8]. Besides the template methods, many visual descriptors have also been utilized. Sun. et al [9] proposed a tracking framework using HOG descriptor. Wu et al. [10] utilized an incremental covariance matrix tensor learning on Riemann manifolds. In general, these visual descriptors have better description ability compared to binary descriptors. But due to the low efficiency, it is relatively difficult to construct a real-time tracking system.

To construct the tracking model, many learning methods have been adapted, such as SVM [11][12], boosting [5][13], and multi-instance learning [14]. To make trackers more robust to pose variation and partial occlusion, some researchers focus on using part model to represent the objects. In [15] Choi et al. combined histogram-wise matching and pixel-wise template matching via constrained optimization problem. Kwon and Lee [16] proposed an approach to automatically update the topology of local patches to handle large pose changes. In addition, updating the constructed model to account for appearance variations is also important. Oron et al. [13] proposed the Extended Lucas Kanade algorithm, which cast the original LK algorithm as a maximum likelihood optimization and then extended it by considering pixel object / background likelihoods in the optimization. Ji et al. [17] utilized the subspace dynamic sparse tracking model with the error term of Gaussian-Laplacian distribution, which fully considered the correlation of object representations between consecutive frames by compressive sensing.

3. BOOSTED BINARY PATTERNS

3.1. Traditional Binary Descriptors

LBP [18] is developed for texture classification and the success is due to its computational simplicity and robustness under illumination variations. The traditional LBP is constructed by the binary coding of the intensity contrast of a center pixel and several surrounding pixels. If the intensity of the surrounding pixels are higher than the center one, the corresponding bit will be assigned 1, otherwise it will be assigned 0. Given a center pixel c, the number of surrounding pixels d, and the distance r between the center pixel and the surrounding pixels, the LBP response is defined by

$$LBP_{d,r} = \sum_{i=1}^{d} sign(I_i - I_c) \times 2^{i-1}, \quad (1)$$

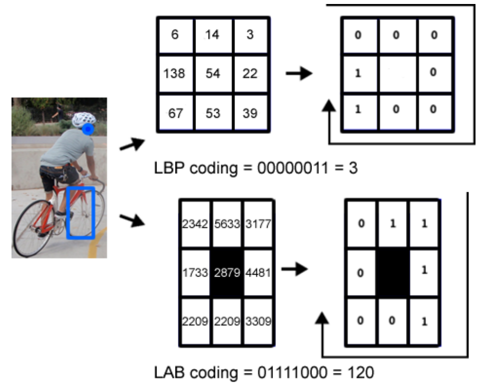

Fig. 1. $LBP_{8,1}$ descriptor and LAB descriptor.

where $sign(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$. Fig. 1 illustrates an example of LBP with $d = 8, r = 1$.

Local Assembled Binaries (LAB) [19] replaces single pixels with rectangles, as shown in Fig. 1. The encoding scheme is the same: if the intensity sum of the surrounding rectangle $R_1, \ldots, R_d$ is larger than the center rectangle C, the corresponding bit will be assigned 1; otherwise it will be assigned 0

$$LAB = \sum_{i=1}^{d} sign(I_{R_i} - I_C) \times 2^{i-1}. \quad (2)$$

The computation of LBP is efficient because the binary coding could be calculated by bit-shift operations. Although LAB uses rectangles instead of single pixels, the intensity sum in any rectangles could also be easily extracted by the integral image [20].

3.2. Boosted Binary Patterns

Traditional binary descriptors have certain drawbacks when employed to encode general object's appearance. A notable disadvantage is the insufficient discriminative ability. Most of the traditional binary descriptors depend on the intensities of artificially designed locations. They will be easily influenced by illumination, occlusion, and noises.

The BBP is proposed to solve this issue. BBP comprises of the center rectangle C and n non-overlapped surrounding rectangles $R_1, R_2, \ldots, R_n$. We denote it by BBP-n in equation (3), where the (x_i, y_i) is the left-top corner of each rectangle, (w_i, h_i) is the width and height

$$BBP_n = \{C, R\},\ R_i = \{x_i, y_i, w_i, h_i\},\ i = 1, \ldots, n. \quad (3)$$

The surrounding rectangles R_i are numbered from 1 to n, which starts at the top-middle one and increases in clockwise way. In our case, the n could be 4, 6, or 8. Fig. 2 illustrates the examples for BBP-4, BBP-6, and BBP-8 respectively.

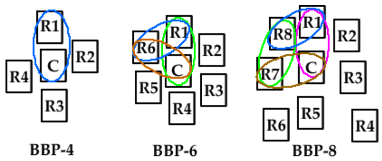

Fig. 2. BBP descriptors with adjacent pairs. Each ellipse shows an example of adjacent pairs. There are 4 adjacent pairs in BBP-4, 12 in BBP-6, and 12 in BBP-8.

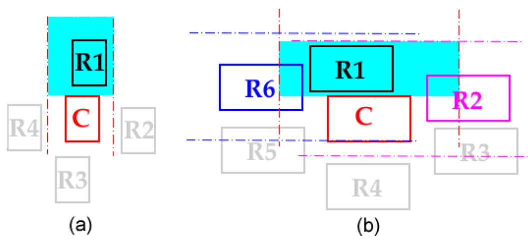

Fig. 3. Constraint of the adjacent pairs. For BBP-4, the rectangles in adjacent pair should overlap at least 50%. For BBP-6, the rectangles in adjacent pair should overlap at least 25%.

In general, these surrounding rectangles could spread anywhere. It has been proved that the geometric relationship between rectangles pairs could contribute to the overall accuracy in the classification [21]. So we define the *adjacent pairs* illustrated as the ellipses in Fig. 2. There are 4 adjacent pairs in BBP-4, 12 in BBP-6. For BBP-8, we consider 12 pairs between surrounding rectangles (e.g., $\{R_1, R_2\}$), or between the center rectangle and nearby surrounding rectangles (e.g., $\{C, R_1\}$). We add a constraint that the two rectangles in an adjacent pair should overlap either on their width or their height. The overlap ratio of BBP-4, BBP-6, and BBP-8 is set to 0.5, 0.25, and 0.5 respectively. Fig. 3(a) illustrates a BBP-4 descriptor. In the adjacent pair $\{C, R_1\}$, R_1 should overlap at least 50% with C's width, so the available region of R_1 restricted by C is the cyan region bounded by the dot-dashed lines around C. Similarly, in Fig. 3(b), R_1 is included in three related adjacent pairs, $\{C, R_1\}, \{R_6, R_1\}, \{R_2, R_1\}$, so it should overlap at least 25% with C's width, R_2's height, and R_6's height. These three constraints correspond to the dot-dashed lines around C, R_2 and R_6 respectively. As a result, the possible region of R_1 is the cyan region crossed by these dot-dashed lines.

With the above constraint, the surrounding rectangles may lay far away from the center rectangle. In this case, the two rectangles in an adjacent pair will have few contextual relationship, so that the descriptor response might be influenced by noises. In the tracking, the confidence of such descriptors will be lower. Then we define the structural diversity as

$$D = \frac{1}{n}\sum_{i=1}^{n}\frac{2 \cdot dis(R_i, C)}{w_i + h_i}, \quad (4)$$

where the 'dis' is the Euclidean distance between the centers of R_i and C. If the surrounding rectangles lay far away from the center one, D will be larger. This factor will be considered in the online feature selection procedure.

We use both the intensity image and the gradient image for BBP extraction. In consideration of the efficiency, we generate the x-direction gradient image and y-direction gradient image respectively. Given a BBP-n descriptor, the final descriptor response is given by

$$f(BBP_n, t) = \sum_{i=1}^{n} sign(g(R_i, t) - g(C, t)) \times 2^{i-1}$$

$$g(R_i, t) = \begin{cases} \sum_{j\in R_i} I(j) & t = 0 \\ \sum_{j\in R_i} Gradx(j) & t = 1. \\ \sum_{j\in R_i} Grady(j) & t = 2 \end{cases} \quad (5)$$

In (5), the sum of a region in both the intensity image and the gradient image could be calculated by the integral image [20]. So BBP extraction is also efficient.

4. SELECTING BBP WITH REALADABOOST

In the tracking procedure, the RealAdaBoost is used to select the meaningful BBP descriptors to describe the target frame by frame. The BBP descriptor can be seen as a function from the image space to a real valued range $f : \mathbf{x} \to [f_{min}, f_{max}]$. For the binary object/background classification problem, suppose the input data as $(\mathbf{x_1}, y_1), \ldots, (\mathbf{x_n}, y_n)$ where $\mathbf{x}_i$ is the training sample and $y \in \{-1, 1\}$ is the class label, we divide the sample space into N_b equal sized sub-ranges B_j, the weak classifier is defined as a piecewise function

$$h(\mathbf{x}) = \frac{1}{2}ln(\frac{W_+^j + \epsilon}{W_-^j + \epsilon}), \quad (6)$$

where ϵ is the smoothing factor, $W_\pm$ is the probability distribution of the descriptor response for positive/negative samples, implemented as a histogram

$$W_\pm^j = P(\mathbf{x} \in X_j, y \in \{-1, 1\}), j = 1, \ldots, N_b. \quad (7)$$

The best descriptor is selected according to the classification error Z of the piecewise function in equation (8)

$$Z = 2\sum_j \sqrt{W_+^j W_-^j}. \quad (8)$$

If we choose the descriptors only based on the discriminative ability, using the descriptors minimizing (8) is a good choice. But in the tracking process, the robustness of the descriptors are also important. Using confident descriptors clearly reduces the possibility of losing target or false alarms, so that it will contribute to the stableness of the whole system. That means, it is better to consider the robustness for each descriptor in addition to the discriminative ability. In each frame, the RealAdaBoost solves the classification task of several positive patches located on the target and negative patches around the target. We know that the margin of the weak classifier h on $\mathbf{x}$ is $y \cdot h(\mathbf{x})$, where the h is normalized to $[-1, 1]$. This margin represents the classification ability. Larger margins imply lower generalization error and better robustness, so that it could better deal with the cases not in the training samples such as illumination or pose variance. We define the normalized margin as

$$K(h, \mathbf{x}) = \frac{y \cdot h(\mathbf{x})}{k}, \quad (9)$$

where k is a parameter related with the structural diversity D in equation (4), calculated as a sigmoid function

$$k = \begin{cases} 1 & D \leq 3 \\ 1.5 - \dfrac{1}{1 + \exp^{3-D}} & D > 3 \end{cases}. \quad (10)$$

The equation (10) means that we believe the BBP descriptors with $D \leq 3$ are more confident, which include $LBP_{8,1}, LBP_{16,1}, LBP_{16,2}$, LAB, and BBPs whose average distance between the surrounding rectangles and the center rectangle is about 3 times smaller than the rectangle size. For the BBPs with larger structural diversity, the confidence in the tracking will gradually decrease.

Balancing the discriminative ability and robustness requires us to evaluate both (8) and (9). So we add (9) as a penalty term into (8), where α is the structure-aware factor

$$Z = 2 \sum_j \sqrt{W_+^j W_-^j} - \frac{\alpha}{n} \sum_{i=1}^{n} K(h, \mathbf{x}_i). \quad (11)$$

If the confidence of the selected descriptor is lower, which corresponds to a smaller margin and larger k. Then the second term will be smaller, and Z will be larger. As a result, this descriptor will have less probability to be selected.

In consideration of the information in time domain, we accumulatively calculate (11) in three consecutive frames after the initialization for the first and second frames. In each frame, we collect 50 positive and 50 negative samples around the target. 20 random BBP descriptors are evaluated and 10 of them with minimum Z will be selected to describe the target. The detail algorithm is illustrated in Fig. 4. In our experiment, the structure-aware factor α is set to 0.1.

Input: Training set $\{(\mathbf{x}_i, y_i)\}, \mathbf{x}_i \in R^d, y_i \in \{-1, 1\}$
1. Initialize sample weight and classifier output $w_i = 0.01, F(\mathbf{x}_i) = 0$
2. Repeat for $t = 1, 2, \dots, 10$
  2.1 Update the sample weight w_i using the t^{th} weak classifier output $w_i = w_i e^{-y_i h_t(\mathbf{x}_i)}$
  2.2 For $m = 1$ to 20
    2.2.1 Randomly generate a BBP descriptor
    2.2.2 Calculate the BBP response following (5)
    2.2.3 Build the predict distribution function $W_{\pm}$ (7)
    2.2.4 Select the best BBP which minimizes the sum of Z (11) in three consecutive frames
  2.3 Update weak classifier $h_t(x)$ (6)
  2.4 Update strong classifier $F_{t+1}(\mathbf{x}_i) = F_t(\mathbf{x}_i) + h_t(\mathbf{x}_i)$

Fig. 4. Selecting BBP descriptor using RealAdaBoost.

5. EXPERIMENTS

We evaluate our tracking algorithm on 10 challenging publicly available sequences. These sequences are from [22], [23], and [14]. The Success Rate(SR) are utilized to evaluate the performance of our algorithm, which is calculated based on the following score

$$score = \frac{area(R_T \cap R_G)}{area(R_T \cup R_G)}, \quad \dots (12)$$

where R_T is the tracking bounding box and R_G is the ground truth bounding box. The tracking result is considered as a success if the score is larger than 0.5.

Firstly, we compare the performance of different BBP descriptors, including BBP-4, BBP-6, BBP-8, and the combination of all BBP descriptors (BBP-All) using the algorithm described in Fig. 4. The block size of BBP varies from 2×2 to one third of the size of tracking target. The tracking accuracy are given in Table. 1. It shows that the accuracy of the BBP descriptor increases along with the number of the surrounding rectangles. The BBP-8 consistently achieves better accuracy compared to BBP-6 and BBP-4. If we combine all BBP descriptors together, the accuracy could be further improved. This result shows the effectiveness of variable binary patterns. It could also be seen that using the gradient information, the tracking accuracy is clearly improved for all BBP descriptors. We know that in general object tracking, it is relatively difficult to classify the target and background only through the intensity information. The advantage of utilizing gradient information is clear.

In addition, we compare the performance of different binary descriptors by integrating them into the same RealAdaBoost framework in Fig. 4. To improve the accuracy, the Haar, LBP, and LAB are also applied on both the intensity domain and the gradient domain. For Haar and LAB descriptor, the block size is the same as BBP. For LBP descriptor, the $LBP_{8,1}, LBP_{16,1}, LBP_{16,2}$ are integrated together. The first 6 columns in Table 2 show the experimental results. It could be seen that in all 10 sequences, BBP consistently achieves at

Table 1. Success rate SR(%) of different BBP descriptors. Bold fonts indicate the best performance.

Sequence	BBP-4	BBP-4+gra	BBP-6	BBP-6+gra	BBP-8	BBP-8+gra	BBP-ALL	BBP-All+gra
Bolt	74	77	76	79	79	82	80	**85**
Biker	55	59	58	64	70	74	73	**76**
Cliff bar	64	66	65	69	74	77	77	**80**
David	72	77	76	80	82	85	84	**88**
Kitesurf	58	60	63	68	71	75	77	**82**
Occluded face	84	86	85	89	92	96	95	**100**
Skiing	62	64	63	66	68	70	73	**77**
Tiger	61	63	63	64	64	**68**	66	**68**
Twinings	79	82	83	85	85	88	87	**92**
Walking person	80	84	80	82	86	88	87	**92**
Average	69	72	71	74	77	80	79	**84**

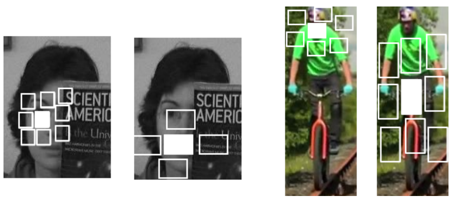

Fig. 5. The first selected two descriptors of faces and bikers.

least 5% better accuracy compared to Haar, LBP, and LAB. This signifies the advantage of variable binary patterns. We also notice that using the penalty term of robustness, the average accuracy could be further improved 4%, especially for those targets with large pose variant such as bolt, kitesurf, and skiing. This implies the importance of balancing the discriminative ability and robustness in the tracking system.

Moreover, we compare our method with commonly-used algorithms, including compressive tracking [23], online AdaBoost (OAB) [5], MIL [14], and TLD [24]. These algorithms achieve the state-of-the-art results in these sequences. Since all of these trackers involve randomness, we run them 5 times and report the average result for each video clip. The last four columns in Table 2 give these quantitative results. It can be seen that the proposed algorithm achieves the best or second best results in most sequences in terms of the success rate. We find that if the tracking target has clear region contrast or rigid shape, the tracking accuracy will be better. As shown in Fig. 5, in the ‘Occluded face’ sequence, the eye and mouth region show a clear contrast to the surrounding regions. In the ‘Biker’ sequence, the body contour of the biker is also a clear feature. These typical structures are well captured by the first select two BBPs, which could not be covered by traditional binary descriptors. We test our tracker on a Intal Dual-Core 3.0 GHz PC with 4GB RAM. It runs at 30 frames per second (FPS), which also shows comparable efficiency with the state-of-the-art methods.

6. CONCLUSIONS

In this paper, we tackle the object tracking problem by a proposed binary descriptor with variable-location and variable-size blocks extracted on both the intensity domain and the gradient domain. A boosting framework which is capable of balancing the discriminative ability and robustness of the binary descriptors is further proposed to online update the tracking model. Experimental results show that our approach achieves very good results in the 10 public challenging videos.

7. ACKNOWLEDGEMENT

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under the Grant RGP36726.

8. REFERENCES

[1] Jifeng Ning, Lei Zhang, David Zhang, and Chengke Wu, “Robust object tracking using joint color-texture histogram,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 7, pp. 1245–1263, 2009.

[2] Prajna Parimita Dash, Dipti Patra, and Sudhansu Kumar Mishra, “Local binary pattern as a texture feature descriptor in object tracking algorithm,” in *Intelligent Computing, Networking, and Informatics*. 2014.

[3] Baopu Li, Can Yang, Qi Zhang, and Guoqing Xu, “Condensation-based multi-person detection and tracking with hog and lbp,” in *IEEE Conference on Information and Automation*, 2014.

[4] Weihua Chen, Lijun Cao, Junge Zhang, and Kaiqi Huang, “An adaptive combination of multiple features for robust tracking in real scene,” in *IEEE International Conference on Computer Vision Workshops*, 2013.

[5] Helmut Grabner, Michael Grabner, and Horst Bischof, “Real-time tracking via on-line boosting,” in *British Machine Vision Conference*, 2006.

[6] Xue Mei and Haibin Ling, “Robust visual tracking using l1 minimization,” in *IEEE International Conference on Computer Vision*, 2009.

Table 2. Success rate SR(%) in comparison with the traditional binary descriptors and state-of-the-arts. Bold fonts indicate the best result. Italic fonts indicate the second best result.

Sequence	Haar	LBP	LAB	BBP-All+gra	BBP-All+gra(no penalty)	CT [23]	MIL [14]	OAB [5]	TLD [24]
Bolt	71	76	73	**85**	79	79	*83*	61	41
Biker	59	69	62	**76**	74	*75*	66	42	42
Cliff bar	64	67	71	*80*	77	**89**	65	23	77
David	78	59	66	88	84	*89*	68	31	**98**
Kitesurf	47	44	55	*82*	75	68	**90**	31	65
Occluded face	*96*	94	90	**100**	*96*	**100**	99	47	56
Skiing	70	*71*	68	**77**	70	70	62	69	59
Tiger	60	59	65	**68**	*66*	60	55	37	41
Twinings	90	77	88	*92*	90	89	72	**98**	46
Walking person	86	65	67	**92**	*89*	*89*	62	86	60
Average	72	74	73	**84**	80	*81*	72	51	59

[7] Dong Wang, Huchuan Lu, and Ming-Hsuan Yang, "Online object tracking with sparse prototypes," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 314–325, 2013.

[8] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[9] Shuifa Sun, Qing Guo, Fangmin Dong, and Bangjun Lei, "On-line boosting based real-time tracking with efficient hog," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

[10] Yi Wu, Jian Cheng, Jinqiao Wang, Hanqing Lu, Jun Wang, Haibin Ling, Erik Blasch, and Li Bai, "Real-time probabilistic covariance tracking with efficient model update," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2824–2837, 2012.

[11] Yancheng Bai and Ming Tang, "Robust tracking via weakly supervised ranking svm," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[12] Sam Hare, Amir Saffari, and Philip HS Torr, "Struck: Structured output tracking with kernels," in *IEEE International Conference on Computer Vision*, 2011.

[13] Shaul Oron, Aharon Bar-Hille, and Shai Avidan, "Extended lucas-kanade tracking," in *European Conference on Computer Vision*. 2014.

[14] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, 2011.

[15] Hong Seok Choi, In Su Kim, and Jin Young Choi, "Combining histogram-wise and pixel-wise matchings for kernel tracking through constrained optimization," *Computer Vision and Image Understanding*, vol. 118, pp. 61–70, 2014.

[16] Junseok Kwon and Kyoung Mu Lee, "Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[17] Zhangjian Ji, Weiqiang Wang, and Ning Xu, "Robust object tracking via incremental subspace dynamic sparse model," in *IEEE Internation Conference on Multimedia and Expo*, 2014.

[18] Guoying Zhao and Matti Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, 2007.

[19] Shengye Yan, Shiguang Shan, Xilin Chen, and Wen Gao, "Locally assembled binary feature with feature-centric cascade for fast and accurate face detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[20] Paul Viola and Michael Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[21] Haoyu Ren and Ze-Nian Li, "Boosted local binaries for object detection," in *IEEE International Conference on Multimedia and Expo*, 2014.

[22] Jakob Santner, Christian Leistner, Amir Saffari, Thomas Pock, and Horst Bischof, "Prost: Parallel robust online simple tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[23] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, "Real-time compressive tracking," in *European Conference on Computer Vision*. 2012.

[24] Zdenek Kalal, Jiri Matas, and Krystian Mikolajczyk, "Tracking-learning-detection," in *IEEE Conference on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.