Research Article

Feature Selection Tracking Algorithm Based on Sparse Representation

Hui-dong Lou,¹ Wei-guang Li,¹ Yue-en Hou,² Qing-he Yao,³ Guo-qiang Ye,¹ and Hao Wan¹

¹*School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou, Guangdong, China*

²*School of Computer Science, Jiaying University, Meizhou, Guangdong, China*

³*School of Engineering, Sun Yat-sen University, Guangzhou, Guangdong, China*

Correspondence should be addressed to Hui-dong Lou; loudong@mail.gdufs.edu.cn

Received 11 September 2015; Accepted 28 September 2015

Academic Editor: Xinguang Zhang

Copyright © 2015 Hui-dong Lou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to enhance the robustness of visual tracking algorithm in complex environment, a novel visual tracking algorithm based on multifeature selection and sparse representation is proposed. In the framework of particles filter, particles with low target similarity are first filtered out by a fast algorithm; then, based on the principle of sparsely reconstructing the sample label, the features with high differentiation against the background are involved in the computation so as to reduce the disturbance of occlusions and noises. Finally, candidate targets are linearly reconstructed via sparse representation and the sparse equation is solved by using APG method to obtain the state of the target. Four comparative experiments demonstrate that the proposed algorithm in this paper has effectively improved the robustness of the target tracking algorithm.

1. Introduction

Visual target tracking algorithm, due to its wide application in fields such as robotics visual control, human-machine interaction, intelligent assistance driving, and video surveillance, has attracted increasing attention from the researchers. However, in complex environment, due to changes in illumination and expression of the target object, object occlusion, and noises disturbance, design of an accurate and real-time visual tracker remains a challenging problem [1].

In recent years, a kind of appearance modeling technology called sparse representation has been widely used for information compression and pattern recognition. Agarwal and Roth [2] have achieved good results in object recognition via sparse representation. Researches [3–5] demonstrate that high recognition rate has been obtained in face recognition via sparse representation. Mei and Ling [6, 7] first introduced sparse representation theory into visual tracking and developed a sparse representation tracking algorithm based on particles filter. The algorithm uses the template dictionary to linearly reconstruct the candidate targets and imposes

sparsity constraints on the reconstruction coefficients. Apart from the target template, the algorithm uses the trivial template to construct the template dictionary, showing good robustness to occlusion; however, it requires a large amount of calculation because the algorithm adopts the LESSO method to solve the sparse equation. Bai and Li [8] developed a structural sparse representation model, which divides the sample into blocks and uses the Block Orthogonal Matching Pursuit (BOMP) algorithm to solve the sparse equation. It effectively improves the calculation speed, but the algorithm is not robust to illumination change. Hou et al. [9] provided a tracking algorithm of block sparse representation, in which each block is given the corresponding weights to improve the robustness to occlusion, but there may be drift due to large noises. A target tracking algorithm based on the structural sparse representation is proposed in [10], which constructs a vector pool of sparse representation coefficients by blocking the target sample and identifies the target state through the similarity information in the vector pool. But this algorithm fails to effectively use the residual error information; its robustness needs to be further improved. Based on this,

Hou et al. [11] developed a target tracking algorithm with sparse representation based on ranking. The algorithm gives full consideration to the sparse representation coefficients and residual error information while locating the target, which improves the robustness of the tracking algorithm. In the framework of structural sparse representation, the sparse coefficients of the candidate targets is classified in [12] by training the Naive Bayes Classifier, strengthening the algorithm's ability to differentiate between the target and its background. But the algorithm fails to extract the features with differentiation; its robustness needs to be further improved. Wang et al. [13] propose a least soft-threshold squares tracking algorithm based on sparse representation, which models the reconstructed residual error terms with the Gaussian-Laplacian distribution and finds the optimal solution of the objective equation by using iterative and soft-threshold methods. Bao et al. [14] impose constraints on the target template coefficients and the trivial template with L_1 -norm and L_2 -norm, respectively. Additionally, it adopts the Accelerated Proximal Gradient (APG) method to solve the objective equation with sparsity constraints. As a result, the accuracy of the tracker and the speed of calculation are significantly improved. Zhuang et al. [15] propose a multitask concept that is similar to [14]; the algorithm also uses the Accelerated Proximal Gradient (APG) method to solve the objective equation by iteration, but it requires a fair amount of calculation. In the framework of sparse representation, Lan et al. [16] propose an objective equation based on adaptive multifeature selection and solve the equation by using the Accelerated Proximal Gradient (APG) method. The multifeature selection method has effectively improved the robustness of the tracking algorithm, but it requires a large amount of calculation.

In order to improve the robustness of visual tracking algorithm, this paper proposes a novel tracking algorithm based on multifeature fusion and sparse representation. According to the image intensity, the proposed algorithm uses the simple algorithm to filter out the candidate targets largely dissimilar to the target. Then, the discriminative features, which come from multifeature, are selected by using a method of sparsely reconstructing sample label. Finally, it uses L_2 -norm to linearly reconstruct the candidate targets and obtains the state of the target.

2. Overview of the Tracking Algorithm

The theorem of proposed tracking algorithm based on multifeature fusion and sparse representation in this paper is shown in Figure 1. Compared with other tracking algorithms, it has the following contributions:

- (1) This paper proposes a fast algorithm that can quickly filter out the particles with low similarity to the target template, improving the calculation speed of the algorithm.
- (2) Based on the sample feature constructed from multifeature, the proposed algorithm uses the method of sparsely reconstructing positive and negative sample

label to extract sample features, which can help these extracted features more discriminatively.

- (3) The conventional tracking algorithms based on sparse representation always use L_1 -norm to impose the constraints on the linear reconstruction function, but they require a large amount of calculation, whereas the proposed algorithm in this paper uses APG method to solve the sparse function with nonnegative constraints, greatly improving the calculation speed of the algorithm.

3. Fast Particle Filter Algorithm

In the framework of particle filtering, most particles bear little resemblance to the target. So they can be filtered out through simple algorithm. Based on this idea, a fast algorithm is proposed to filter out the particles with low similarity to the target.

Let $Z = [Z_1, Z_2, \dots, Z_n]$ denote the sample set of the candidate targets, and $\bar{Z} \in R^m$ denote the target template, where $Z_i \in R^m$ is one of the candidate target samples, m is the dimension of the sample, and n represents the number of particles. In order to reduce the amount of calculation, only the gray image information of the sample is involved in the calculation. By normalizing the candidate target Z_i and target template $\bar{Z}$, we can get $z'_i = z_i / (\sum_{j=1}^m z_{i,j})$ and $\bar{z}' = \bar{z} / (\sum_{j=1}^m \bar{z}_j)$, where $z_{i,j}$ is the j th feature of z_i , $\bar{z}_j$ is the j th feature of $\bar{z}$, and u_i is the similarity measure of z_i and $\bar{z}$. If z_i bears much resemblance to $\bar{z}$, then there will be no remarkable difference between the values of u_i . So we can filter out the particles according to the fluctuation of values of u_i :

$$\bar{u}_i = u_{i,j} - \text{mean}(u_i), \quad (1)$$

where $u_{i,j}$ is the j th element of u_i and $\text{mean}(u_i)$ is the mean of all elements of u_i . The value of $\bar{u}_i$ reflects the fluctuation of values of u_i . If the target is disturbed by occlusions or noises, the real target can possibly be filtered out when the value of $\bar{u}_i$ is too small. In order to solve this problem, only the elements with 50% variance value in $\bar{u}_i$ will be involved in the calculation. Sort the elements in $\bar{u}_i$ in descent order according to their values and eliminate the elements with values in the top half; then we can get the vector $\bar{u}_i$. Suppose $y_i = \sum_{j=1}^{m/2} \bar{u}_{i,j}$, where $\bar{u}_{i,j}$ is the j th element of $\bar{u}_i$; then we can filter out the particles according to the value of y_i . Through experiment, we select the y_i with smaller value ($n_0 = n/3$) from all the particles to participate in the later operation. Through the algorithm, we can quickly eliminate two-thirds of particles with low similarity to the object, which effectively improves the calculation speed of the tracking algorithm.

4. Feature Selection

In order to improve the robustness of the tracking algorithm, we use the image intensity and LBP feature $s_L \in R^{m_2}$ to construct the sample features. Denote the gray features of the sample by $s_G \in R^{m_1}$ and the LBP feature of the sample

FIGURE 1: Proposed tracking algorithm.

by $s_L \in R^{m_2}$. When s_G and s_L are combined, we can get $s = [s_G, s_L]^T$, where $s \in R^k$ represents the sample feature, $k = m_1 + m_2$. Because the dimension of s is high, it contains a fair amount of redundant information. In order that the algorithm can differentiate between target and background better, we need to select the discriminative features from s .

During the tracking process, according to the target sample in the first frame and the tracking result, we can get the positive and negative sample set of the target $T = [t_p, t_n]$, where the positive sample of the target is $T_p \in R^{k \times p}$ and the negative sample is $T_n \in R^{k \times q}$, where p represents the number of the positive samples and q indicates the number of negative samples. Use the sparse reconstruction theory to select the sample features; then we can get

$$\min_A \left(\frac{1}{2} \|T^T A - l\|_2^2 + \lambda \|A\|_2 \right), \quad (2)$$

where $\|\cdot\|$ is the norm operator and l represents the label vectors of the positive and negative samples. The positive and negative samples are set to 1 and -1, respectively. λ is the sparse adjustment coefficient and $A \in R^k$ represents the reconstruction vectors, $A = [A_1, A_2, \dots, A_k]$. It should be noted that, in (2), we use the 2-norm to impose constraints on the sparsity of A . The benefit is that it can effectively reduce the amount of calculation in solving the sparse function. Set threshold as τ , and when $A_i > \tau$, the corresponding feature has a strong ability to differentiate between the target and its background.

Construct the mapped vector of feature selection $M = [M_1, M_2, \dots, M_k]^T$, $M \in R^k$. Consider

$$\begin{aligned} M_i &= 0, & A_i &\leq \tau, \\ M_i &= 1, & A_i &> \tau. \end{aligned} \quad (3)$$

Here, s is the feature vector before the feature selection. After the feature selection, the feature vector $\bar{s}$ is defined as

$$\bar{s} = s \otimes M, \quad (4)$$

where $\otimes$ denotes the multiplication of the corresponding elements within the vector.

5. Solution to the Sparse Equation with APG Approach

After feature selection, we get the feature vector set of the candidate targets $S = [\bar{s}_1, \bar{s}_2, \dots, \bar{s}_{n_0}]$, $\bar{s} \in R^{k_0}$, and k_0 denotes the dimension of each sample after feature selection. Let $D = [D_0, I, -I]$ denote the template dictionary, where D_0 is the target sample set, $D_0 \in R^{k_0 \times f}$, and $I \in R^{k_0 \times k_0}$ represents the unit diagonal matrix used to reduce the disturbance of occlusions and noises.

Using template dictionary D to make sparse linear reconstruction for the candidate targets, we can get

$$\begin{aligned} \arg \min_C & \quad \frac{1}{2} \|\bar{s} - DC\|_2^2 + \lambda \|C\|_1 \\ \text{s.t.} & \quad C_T \geq 0, \end{aligned} \quad (5)$$

where $C = [C_T, C_I]^T$ represents the sparse coefficients, C_T is the sparse coefficients corresponding to the target template, and C_I is the sparse coefficients corresponding to I and $-I$. Impose the nonnegative constraints on the elements in C_T to improve the robustness of the tracker [7].

After adding a penalty term, (5) with nonnegative constraints can be updated by

$$\arg \min_C \frac{1}{2} \|\bar{s} - DC\|_2^2 + \lambda \|C\|_1 + J(C_T), \quad (6)$$

where $J(\cdot)$ is the penalty term defined by

$$\begin{aligned} J(a) &= 1, \quad a \geq 0; \\ J(a) &= +\infty, \quad a < 0. \end{aligned} \quad (7)$$

The solution to (6) is equivalent to optimizing the convex function. This paper uses the Accelerated Proximal Gradient (APG) approach to find the optimal solution to (6). Set

$$\begin{aligned} F(C) &= \frac{1}{2} \|\bar{s} - DC\|_2^2 + \lambda \|C\|_1, \\ G(C) &= J(C_T), \end{aligned} \quad (8)$$

where $F(C)$ is a differentiable convex function and $G(C)$ is a discontinuous convex function. The specific algorithm is shown in Algorithm 1.

Algorithm 1. Minimization algorithm for (6) by using APG method is as follows:

- (1) Initialize, $\alpha_0 = \alpha_{-1} = 0, t_0 = t_{-1} = 1$
- (2) For $i = 1:4$
- (3) $\beta_{i+1} = \alpha_i + ((t_{i-1} - 1)/t_i)(\alpha_i - \alpha_{i-1});$
- (4) $\alpha_{i+1} = \arg \min_C (\gamma/2) \|C - \beta_{i+1} + \nabla F(\beta_{i+1})/\gamma\|_2^2 + G(C);$
- (5) $t_{i+1} = (1 + \sqrt{1 + 4t_i^2})/2;$
- (6) end for

In Algorithm 1, step (4) needs to find the minimum value of the function, so it can be updated by

$$\arg \min_C \frac{\gamma}{2} \|C - h_{i+1}\|_2^2 + J(C_T), \quad (9)$$

where $h_{i+1} = \beta_{i+1} - \nabla F(\beta_{i+1})/\gamma$. Equation (9) can be seen as to solve the minimum values of the two functions as follow:

$$\arg \min_{C_T} \frac{\gamma}{2} \|C_T - h_{i+1}^T\|_2^2 + J(C_T), \quad (10)$$

$$\arg \min_{C_I} \frac{\gamma}{2} \|C_I - h_{i+1}^I\|_2^2, \quad (11)$$

where h_{i+1}^T and h_{i+1}^I are the elements corresponding to T and $[I, -I]$ in h_{i+1} . It can be seen that the optimal solution to (10) is $C_T = \max(0, h_{i+1}^T)$ and the optimal solution to (11) is $C_I = h_{i+1}^I$.

6. Object Tracking

The proposed algorithm in this paper is implemented in the framework of particles filter. In the first frame of the video, the initial state of the target is picked by mouse or captured through target recognition. Let $\{o_1, o_2, \dots, o_t\}$ denote the observation values from the first frame to the t th frame of

the video. x_t^i is the state of the i th particle in the t th frame. The target state in t th frame is

$$x_t = \arg \max_{x_t^i} (p(x_t^i | o_{1:t})). \quad (12)$$

Here, $p(x_t^i | o_{1:t})$ can be obtained by solving the following equations:

$$p(x_t | o_{1:t-1}) = \int p(x_t | x_{t-1}) p(x_{t-1} | o_{1:t-1}) dx_{t-1}, \quad (13)$$

$$p(x_t | o_{1:t}) = \frac{p(o_t | x_t) p(x_t | o_{1:t-1})}{p(o_t | o_{1:t-1})}. \quad (14)$$

In (13), $p(x_t | x_{t-1})$ is the state transfer function. The state of the sample is defined by six-dimensional affine vector $[\tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6]$, which represents the x coordinate, y coordinate, length-width ratio, rotation angle, torsion angle, and scale, respectively. Suppose that the six parameters are mutually independent; then the state transfer function $p(x_t | x_{t-1})$ can be represented as

$$p(x_t | x_{t-1}) = N(x_t | x_{t-1}, \Sigma), \quad (15)$$

where Σ is the diagonal matrix constituted by $[\tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6]$.

The relation between the similarity measure $p(o_t | x_t)$ in (14) and the residual error e based on sparse representation can be represented as

$$p(o_t | x_t) \propto \frac{1}{e}. \quad (16)$$

The proposed algorithm is shown in Algorithm 2.

Algorithm 2. The proposed algorithm in this paper is as follows.

Input. Video, the target state in the first frame

- (1) Obtain the target dictionary of the initial target and positive and negative samples
- (2) for $i = 1: f$ (f is the number of the frames of the video)
- (3) Update particles with normal distribution
- (4) Extract the intensity of the particles s_G and filter out particles
- (5) Extract LBP feature of the sample s_L and get s
- (6) Use sparse reconstruction theory to select the sample features, and then get the feature vector $\bar{s}$
- (7) Construct a sparse equation and solve it with APG method
- (8) if f can be divided by 5
- (9) Update the feature vector $\bar{s}$, update template dictionary, and update the positive and negative samples
- (10) end if
- (11) end for

Output. The state of target in each frame.

FIGURE 2: Tracking results on sequence “Panda.”

7. Experiments and Discussion

The proposed algorithm in this paper is implemented in Matlab 2009b. In order to make the tracking results of the proposed algorithm more convincing, the experiment selects other three representative tracking algorithms for comparison, including L_1 tracking algorithm [7], IVT tracking algorithm [17], and ASLSAM tracking algorithm. The four algorithms will be tested by four internationally used tracking test videos including “Panda,” “Woman Square,” “Trellis,” and “ThreePassShop2cor”. It is a challenging task to track these targets because all the videos pose challenging factors such as partial occlusion and variations in illumination, pose, and scale.

The first test video is Panda with a cartoon panda as the tracked target. For this video, the difficulty in tracking the target lies in the partial occlusion and the target rotation. Figure 2 shows the test results of the four algorithms in the 4th, 55th, 108th, 176th, 201th, and 241th frame. The test results of the proposed algorithm, ASLSAM algorithm, L_1 algorithm, and IVT algorithm are marked by the blue, red, gray, and purple boxes, respectively. As shown in Figure 2, our proposed algorithm can handle the target very well before the 201th frame but drift from the target in the 201th and 241th frame due to the occlusion. However, it has never missed the target throughout the tracking process. While the other three algorithms miss the target when there are occlusion and target rotation.

The second test video is Trellis with the face as the target. There is a great difficulty in tracking the target because of the drastic illumination and variation in pose in the video. As shown in Figure 3, our proposed algorithm can track the target faithfully throughout the tracking process without

being affected by the pose and illumination change. But L_1 algorithm and IVT algorithm miss the target in the 276th frame and ASLSAM algorithm misses the target in the 532th frame.

The third test video is Woman Square with a pedestrian as the target. The main challenge of tracking the target comes from the partial occlusion. As shown in Figure 4, only our proposed algorithm has successfully tracked the target without being affected by the partial occlusion while all the other three algorithms miss the target in the 143th frame as a result of the partial occlusion.

The fourth test video is ThreePassShop2Cor. The difficulty of tracking the target lies in the occlusion, the disturbance of the similar object, and scale variation. The tracking results of the four algorithms are shown in Figure 5. Because of the disturbance of occlusion and similar objects, our proposed tracker drifts from the target in the 125th frame but relocates the target later. While the ASLSAM and IVT algorithms locate the similar object instead of the tracked target in the 125th frame and the L_1 algorithm misses the target in the 303th frame.

In order to compare the tracking results better, Table 1 lists the maximum, mean, and standard variance values of the tracking error of the four algorithms in four image sequences. The tracking error refers to the Euclidean distance between the center point of the target derived from the tracking algorithm and the center point of the actual target. It can be seen that the tracking results on all the four videos demonstrate that our proposed algorithm achieves more favorable performance than the other three.

Apart from handling the tracking error, in order to reflect the relation between the tracking results and the actual

TABLE 1: Maximum, mean, and standard variance values of the tracking error.

Video	Proposed algorithm			ASL algorithm			L_1 algorithm			IVT algorithm		
	Max	Mean	STD	Max	Mean	STD	Max	Mean	STD	Max	Mean	STD
TPS	24.2	3.9	4.6	115.3	65.6	38.0	40.9	18.2	14.1	114.9	65.8	38.4
WS	12.6	3.4	2.2	259.8	15.2	90.3	225.3	129.4	75.2	343.8	156.8	98.8
Panda	23.4	8.6	5.1	385.8	136.3	158.9	410.1	129.9	112.2	190.4	68.3	68.8
Trellis	13.6	3.6	2.9	191.2	21.6	45.4	222.1	59.2	54.4	151.6	72.3	51.0
Average	18.5	4.9	3.7	238.0	59.7	83.2	224.6	84.2	64.0	200.2	90.8	64.25

FIGURE 3: Tracking results on sequence “Trellis.”

appearance of the target, Table 2 lists the success rate of the four tracking algorithms based on the PASCAL VOC standard [18]. It can be seen that the success rate of our proposed algorithm is remarkably higher than that of the other three.

8. Conclusion

A new sparse representation-based tracking algorithm in the framework of particles filter is proposed in this paper. The new method first uses a fast algorithm to filter out

FIGURE 4: Tracking results on sequence "Woman Square."

TABLE 2: Success rate of tracking algorithms.

Video	Proposed algorithm	ASLSAM algorithm	L_1 algorithm	IVT algorithm
TPS	0.99	0.21	0.49	0.24
WS	0.76	0.21	0.21	0.20
Panda	0.85	0.52	0.15	0.42
Trellis	0.94	0.74	0.24	0.26
Average	0.86	0.42	0.27	0.28

the particles largely dissimilar to the tracked target, which reduces the amount of calculation. Then, with the intensity feature and LBP feature combined, the algorithm extracts the features with discriminative ability via L_2 sparse representation, improving its robustness to the occlusion and disturbance. Furthermore, it adopts the APG method to solve

the sparse equation with nonnegative coefficient constraints, improving the computational speed and robustness of the proposed algorithm. Finally, the experiments demonstrate that the proposed tracker achieves more favorable tracking results.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publishing of this paper.

Acknowledgment

This work is supported by the Foundation of Innovation Demonstration Project for NC mechanical product in Guangdong, China (no. 2013B011301026).

FIGURE 5: Tracking results on sequence “ThreePassShop2Cor.”

References

- [1] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: a survey,” *ACM Computing Surveys*, vol. 38, no. 4, pp. 1158–1166, 2006.
- [2] S. Agarwal and D. Roth, “Learning a sparse representation for object detection,” *Computer Science*, vol. 2353, pp. 97–101, 2006.
- [3] J. Wright, Y. Yang Allen, A. Ganesh et al., “Robust face recognition via sparse representation,” in *Proceedings of the 8th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 210–227, Amsterdam, The Netherlands, September 2008.
- [4] L. Zhang, M. Yang, and X. Feng, “Sparse representation or collaborative representation: which helps face recognition?” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV ’11)*, pp. 471–478, IEEE, Barcelona, Spain, November 2011.
- [5] M. Yang, L. Zhang, D. Zhang, and S. Wang, “Relaxed collaborative representation for pattern classification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR ’12)*, pp. 2224–2231, IEEE, Providence, RI, USA, June 2012.
- [6] X. Mei and H. B. Ling, “Robust visual tracking and vehicle classification via sparse representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2259–2272, 2011.
- [7] X. Mei and H. B. Ling, “Robust visual tracking using l_1 minimization,” in *Proceedings of the 12th IEEE Computer Vision*, pp. 1436–1443, IEEE, Kyoto, Japan, September–October 2009.
- [8] T. X. Bai and Y. F. Li, “Robust visual tracking with structured sparse representation appearance model,” *Pattern Recognition*, vol. 45, no. 6, pp. 2390–2404, 2012.
- [9] Y.-E. Hou, W.-G. Li, A.-Q. Rong, and G.-Q. Ye, “Tracking algorithm of block sparse representation with background information,” *Journal of South China University of Technology*, vol. 41, no. 8, pp. 21–27, 2013.
- [10] X. Jia, H. C. Lu, and M.-H. Yang, “Visual tracking via adaptive structural local sparse appearance model,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR ’12)*, pp. 1822–1829, IEEE, Providence, RI, USA, June 2012.

- [11] Y.-E. Hou, W.-G. Li, S. Sekou et al., "Target tracking algorithm with structured sparse representation based on ranks," *Journal of South China University of Technology (Natural Science Edition)*, vol. 41, no. 11, pp. 23–29, 2013.
- [12] Y. Hou, W. Li, A. Rong, H. Lou, and S. Quan, "Robust visual ℓ_2 -regularized least squares tracker with Bayes classifier and coding error," *Journal of Electronic Imaging*, vol. 22, no. 4, Article ID 043036, 2013.
- [13] D. Wang, H. Lu, and M.-H. Yang, "Least soft-threshold squares tracking," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 2371–2378, Portland, Ore, USA, June 2013.
- [14] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust L1 tracker using accelerated proximal gradient approach," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1830–1837, IEEE, Providence, RI, USA, June 2012.
- [15] B. Zhuang, H. Lu, Z. Xiao, and D. Wang, "Visual tracking via discriminative sparse similarity map," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1872–1881, 2014.
- [16] X. Lan, A. J. Ma, and P. C. Yuen, "Multi-cue visual tracking using robust feature-level fusion based on joint sparse representation," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 1194–1201, Columbus, Ohio, USA, June 2014.
- [17] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 125–141, 2008.
- [18] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.