Proceeding of the IEEE
International Conference on Robotics and Biomimetics (ROBIO)
Shenzhen, China, December 2013

Visual target tracking via weighted non-sparse representation and online metric learning

Jingdi Duan[1],Baojie Fan*[2],Yang Cong[3]
1. Neusoft Corporation, Shenyang 110179, China
2. College of Automation, Nanjing University of Posts and Telecommunications, Nanjing, 210046, China
3. State Key Laboratory of Robotics, Shenyang Institute Automation, Chinese Academy of Sciences, Shenyang, 110016, China

***Abstract*—In this paper, we propose online metric learning tracking method that consider visual tracking as a similarity measurement problem, and incorporates adaptive metric learning and generative histogram model based on non-sparse linear representation into the target tracking framework. We propose a generative histogram model based on non-sparse linear representation, which make full use of the non-sparse coefficients to discriminate between the target and the background. The similarity metric is adaptively learned online to maximize the margin of the distance between the foreground target and background. A bi-linear graph is defined accordingly to propagate the label of each sample. The model can also self-update using the more confident new samples. Numerous experiments on various challenging videos demonstrate that the proposed tracker performs favorably against several state-of-the-art algorithms.**

Keywords—non-sparse representation; online metric learning; bi-linear graph; target tracking

I. INTRODUCTION

Visual tracking plays a critical role in computer vision field, especially for the applications of motion analysis, activity recognition, video surveillance, traffic monitoring and human-computer interaction. While much progress has been made in recent years, it is still a challenging problem to develop a robust algorithm for complex and dynamic scenes due to factors such as partial occlusions, varying illumination, background clutter, pose variation and shape deformation.

Tracking algorithms can be classified as either generative [1-6] or discriminative [7-14] methods. Generative methods focus on searching for the regions which are the most similar to the tracked targets, while discriminative methods cast tracking as a classification problem that distinguishes the tracked targets from the surrounding backgrounds.

Motivated by the success of sparse representation-based face recognition [15], Mei et al. [16] develop a novel L1 tracker that uses a series of target templates and trivial templates to model the tracked target, where the target templates are used to describe the object class to be tracked and trivial templates are used to deal with outliers (e.g., partial occlusion) with the sparsity constraints. For tracking, a candidate sample can be sparsely represented by both target and trivial templates, and its corresponding likelihood is determined by the reconstruction error with respect to target templates. We note that this formulation is a linear regression problem with sparsity constraints on the representation coefficients. Despite of demonstrated success, there are still several issues to be addressed. First, the algorithm is able to deal with occlusion with L1 minimization formulation using trivial templates at the expense of high computational cost. Second, the trivial templates can be used to model any kind of image regions whether they are from the target objects or the background. Thus, the reconstruction errors of images from the occluded target and the background may be both small. As a result of generative formulation where the sample with minimal reconstruction error is regarded as the tracking result, ambiguities are likely to accumulate and cause tracking failure.

Furthermore, several methods have been proposed to improve the L1 tracker in terms of both speed and accuracy [17-23], such as using accelerated proximal gradient algorithm [17], replacing raw pixel templates with orthogonal basis vectors [18, 19], modeling the similarity between different candidates [20], to name a few. Besides, Liu et al. [16] propose a method which selects a sparse and discriminative set of features to improve tracking efficiency and robustness. One potential problem with this approach is that the number of discriminative features is fixed, which may not be effective for tracking in dynamic and complex scenes. In [15], a tracking algorithm based on histograms of local sparse representation is proposed. The target object is located via mean-shift of voting maps constructed basing on reconstruction errors. Recently, research has revealed that the L1-norm induced sparsity does not in general help improve the accuracy of image classification; and non-sparse representation based methods are typically orders of magnitudes faster than the sparse representation based ones with competitive and sometimes even better accuracy [26－28].

Without considering the issue of object representation, visual tracking may be considered as a similarity measurement problem, i.e. the issue of distance metric that is used to determine the closet match in the feature space. Thus, most existing tracking methods employ a fixed pre-specified metric, e.g. the Euclidean metric, the Matusita metric, the Bhattacharyya coefficient , the Kullback-Leibler divergence, the information-theoretic similarity measures. However, simply using such a pre-defined metric is problematic and limited in practice, which often leads to a false positive match or turning fails the tracker. In order to choice a robust metric adaptively, the metric learning is incorporated recently [29, 30, 31,32,33].

In this paper, we consider visual tracking as a similarity measurement problem. The adaptive online metric learning and generative histogram model based on non-sparse linear re-

978-1-4799-2744-9/13/$31.00 ©2013 IEEE

presentation are incorporated into the target tracking framework. A novel histogram model is as target representation, which make full use of the non-sparse coefficients to discriminate the target from the background. Given a number of labeled data followed by a sequential input of unseen testing samples, the similarity metric is learned to maximize the margin of the distance between the foreground object and background, and a new bi-linear graph is defined accordingly to propagate the label of each sample. while the model can also self-update using the more confident new samples.

II. GENERATIVE HISTOGRAM MODEL BASED ON NON-SPARSE LINEAR REPRESENTATION

Motivated by the success of sparse coding for image classification [15] as well as object tracking [16], we present a generative model as object representation that considers the location information of patches and takes occlusion into account. In each frame, we draw the candidates around the tracked result in the previous frame with a particle filter. To better track the target, we employ affine transformation to model object motion. In addition, we assume that the affine parameters are independent and can be modeled with six scalar Gaussian distributions.

In the first frame, the tracked target is chosen mannualy. We sample a set of M overlapped local image patches inside the target region, and its gray-scale features are used to represent the local information. Each patch is converted to a vector $y_i \in R^{P\times 1}$, where P is the size of the patch. In this paper, we naturally integrates the patch importance into the online learning procedure. The weight for each patch near the object location is larger than that far from the object location which means the patch near the object location contributes larger to the tracking metric criterion. The weight function is a monotone decreasing function with respect to the Euclidean distance between the locations of sample x_{1j} and sample x_{10}

$$w_{j0} = \frac{1}{c} e^{-\left|l(x_{1j})-l(x_{10})\right|} \tag{1}$$

where $l(\ )$ is the location function and c is a normalization constant. Eq.(1) weighs the patch according to their importance to the tracking metric criterion, i.e.,the instances near the tracking location at the current frame contribute more to tracking metric criterion than those far from the tracking location.

The coefficient vector α of each patch is computed by the following non-sparse linear representation model:

$$\min_{\alpha_i} \left\| y_i - w_{j0} D\alpha_i \right\|_2 + \lambda \left\| \alpha_i \right\|_{l2} \tag{2}$$

where the dictionary $D \in R^{P\times J}$ is generated from k-means cluster centers (J denotes the number of cluster centers) via the patches belonging to the labeled target object in the first frame and it consists of the most representative patterns of the target object. Eq.2 can be regarded as a least-square problem, which admits an extremely simple and efficient closed form solution.The solution of can be analytically derived as

$$\alpha_i = (A^T A + \lambda I)^{-1} A^T y_i \,. \tag{3}$$

The sparse coefficient vector α_i of each patch is concatenated to form a histogram $f = [\alpha_1 \ \ \alpha_2 \ \ .... \ \ \alpha_M]'$ for one candidate.

In order to deal with occlusions, we adopt the stategy in[22] to modify the constructed histogram to exclude the occluded patches when describing the target object. The patch with large reconstruction error is regarded as occlusion and the corresponding sparse coefficient vector is set to be zero. A weighted histogram is generated by

$$\rho = f \otimes \pi \tag{4}$$

where ⊗ denotes the element-wise multiplication. Each element of o is an indicator of occlusion of the corresponding patch and is obtained by

$$\pi_i = \begin{cases} 1, \varepsilon_i < \varepsilon_0 \\ 0, \ otherwise \end{cases} \tag{5}$$

where $\pi_i = \left\| y_i - w_{j0} D\alpha_i \right\|_2$ is the reconstruction error of patch y_i , and ε0 is a predefined threshold which determines the patch is occluded or not.

So, each candidate is represented by a sparsity-based histogram π_i . This improved representation scheme takes spatial information of local patches and occlusion into account, thereby making it more effective and robust.

III. ONLINE METRIC LEARNING(OML) FRAMEWORK FOR TARGET TRACKING

Based the above target representation, this section present the proposed online metric learning(OML) framework for target tracking, which uses metric learning to measure the similarity and adopts semisupervised learning to label the testing samples. specifically, we first learn a matrix W for similarity measurement, then classify new unlabeled data using W , lastly adds those new labeled data with high confidence scores to update W accordingly. Such a process iterates for online processing.

A. Online Metric Learning

The goal of online metric learning is to learn a similarity function $s_W(p_i, p_j)$ parameterized by matrix W for similarity measurement, which is a bi-linear form as

$$s_W(p_i, p_j) = p_i^T W p_j \,, \tag{6}$$

where $p_i, p_j \in R^P$ are the feature vectors and $W \in R^{P\times P}$, s_W assigns higher scores to more similar pairs of feature vectors and vice versa. For robustness, a soft margin is given as

$$s_W(p_i, p_i^+) > s_W(p_i, p_i^-) + 1, \forall p_i^+, p_i^- \in R^P \ . \tag{7}$$

Here, p_i is more similar to p_i^+ than p_i^- , in this case, p_i, p_i^+ belong to the same class, and p_i, p_i^- vice verse. The hinge loss function is used to measure the cost

$$l_W(p_i, p_i^+, p_i^-) = \max(0, 1 - s_W(p_i, p_i^+) + s_W(p_i, p_i^-)) \quad (8)$$

For the Online Algorithm for Scalable Image Similarity learning (OASIS) in [34]. The passive Aggressive algorithm is adopted to solve the above model iteratively. First of all, W is initialized to an identity matrix. Then, the algorithm iteratively draws a random triplet, and solves the following convex problem with a soft margin

$$W^i = \arg\min_W \frac{1}{2}\left\|W - W^{i-1}\right\|_{Fro}^2 + C\xi$$
$$s.t. l_W(p_i, p_i^+, p_i^-) \le \xi and \xi \ge 0 \quad (10)$$

where $\| \|_{Fro}$ is the Frobenius norm (point wise L2 norm) and C is the tuning parameter. p_i, p_i^+ belong to the same class,. In the *ith* iteration, W^i is updated to optimized a trade-off between staying close to the previous parameter W^{i-1} and minimizing the loss on the current triplet(C = 0:2):

$$W = W^{i-1} + \tau V_i, \tau = \min\left\{C, \frac{l_{W^{i-1}}(p_i, p_i^+, p_i^-)}{\|V_i\|^2}\right\} \quad (11)$$

Depending on this, we define the bi-linear graph as

$$S_{i,j} = \max(0, s_W(i,j)) = \max(0, p_i^T W p_j) \quad (12)$$

B. Label Propagation

In the following, we adopt the graph-based semi-supervised learning (also called label propagation) to make a more accurate prediction, which associates the information of both the labeled data and unlabeled data. The graph $G = (V; E)$ is defined, where node V denotes $N = n + m$ feature vector, (n and m are the number of training and testing samples and m = 1 in this paper); E contains the edges of every pair of nodes measuring the pairwise similarity. Suppose we have K classes, let $F = \begin{bmatrix} F_l \\ F_u \end{bmatrix} \in R^{(n_l+n_u)\times K}$, where $F_l = \begin{bmatrix} f_1 & f_2 & \dots & f_{n_l} \end{bmatrix} \in R^{n_l \times K}$ denotes the labeled data, and $F_u = \begin{bmatrix} f_1 & f_2 & \dots & f_{n_u} \end{bmatrix} \in R^{n_u \times K}$ is the label matrix of unlabeled data needed to be predicted. We first normalize the similarity matrix S as:

$$P_{ij} = P(i \to j) = \frac{S_{ij}}{\sum_{i=1}^{n} S_{ik}} \quad (13)$$

This matrix can be split into labeled and unlabeled sub-matrices,

$$P = \begin{bmatrix} P_{ll} & P_{lu} \\ P_{ul} & P_{uu} \end{bmatrix} \quad (14)$$

For label propagation, we have

$$F_u^{t+1} \leftarrow P_{uu} F_u^t + P_{ul} F_l, \quad (15)$$

When t approach infinity, we have

$$F_u = \lim_{t\to\infty} (P_{uu})^t F_u^0 + (\sum_{i=1}^{t} P_{uu}^{i-1}) P_{ul} F_l \quad (16)$$

where F_u^0 is the initial value of F_u. Since the sum of each row of P equals to 1, we have $(P_{uu})^n$ converge to zero. Using the Taylor Equation, the second item can be written as $F_u = (I - P_{uu})^{-1} P_{ul} Y_l$. Due to P_{uu} is a fixed real number in our case, $(I - P_{uu})^{-1}$ is also a real number and invertible. Thus, F_u can be calculated using the largest values of each row, which is also consistent with the simplified function:

$$c_x^* = \arg\max_c E_c(x), E_c(x_i) = \sum_{j=1}^{n} \delta_c(j) S_{i,j} \quad (17)$$

where $c \in \{1, 2\}$, x_i is the query sample and $\delta_c(i)$ is a indicate function. $E_c(x_i)$ is the energy function, which measures the cost of x belonging to class c. Thus, given x, the optimal solution of c is the one with maximize the cost of $E_c(x_i)$.

IV. EXPERIMENTS

The proposed algorithm is implemented in MATLAB on a Pentium 2.3 GHz Dual Core laptop with 2GB memory. For each sequence, the location of the target object is manually labeled in the first frame. In order to evaluate the performance of our tracker, we conduct experiments on six challenging image sequences. These sequences cover most challenging situations in object tracking: occlusion, motion blur, in-plane and out-of-plane rotation, large illumination change, scale variation and complex background. The proposed approach is compared with six state-of-the-art tracking methods with the same initial position of the target. These algorithms are the Frag tracking [2], IVT tracking [3], MIL tracking [9], VTD tracking [6], PN tracking [14], and L1 tracking [16] methods. For fair evaluation, we evaluate those methods using the source codes provided by the authors.

The parameters are presented as follows. Note that they are fixed for all sequences. The variable λ in Eq. 2 is fixed to be 0.01, The threshold ε_0 in Eq. 8 is 0.04.

A. Quantitative Comparison

For quantitative performance comparison, two popular evaluation criteria are used, namely, center location error as well as the overlapping rate[35], and the results are shown in Table 1 and Table 2. Overall, the proposed tracker performs well against the other state-of-the-art algorithms. The best and second best results are shown in red and blue fonts.

Table 1. Average center location error (in pixel).

	Frag	IVT	MIL	L1	PN	VTD	Our
Singer1	22.1	8.5	15.2	4.6	32.7	**4.1**	**3.45**
Faceocc2	15.5	**10.3**	14.1	11.2	18.6	10.5	**4.5**
Car11	64.0	**2.2**	43.5	33.3	25.2	27.1	**1.8**
Jumping	58.5	36.9	**9.9**	12.5	**3.6**	63.0	11.8
Stone	65.9	**2.2**	32.3	19.2	8.0	31.4	**2.34**
Caviar	5.7	45.2	48.5	119.9	5.6	**3.9**	**2.6**
Animals	92.1	127.5	66.5	15.3	–	**12.0**	**9.7**
Girl	**18.1**	48.5	32.3	62.5	23.2	21.5	**10.8**

Table 2. Average overlap rate based on [35].

	Frag	IVT	MIL	L1	PN	VTD	Our
Singer1	0.34	0.66	0.33	0.70	0.41	0.79	**0.86**
Faceocc2	0.60	0.58	0.61	**0.67**	0.49	0.59	**0.83**
Car11	0.08	**0.80**	0.17	0.43	0.37	0.43	**0.79**
Jumping	0.13	0.28	0.52	0.55	**0.69**	0.07	**0.63**
Stone	0.15	**0.66**	0.32	0.29	0.41	0.42	**0.62**
Caviar	0.68	0.28	0.25	0.27	0.70	**0.83**	**0.84**
Animals	0.07	0.21	0.21	0.53	0.41	0.57	**0.60**
Girl	**0.68**	0.42	0.51	0.32	0.57	0.51	**0.69**

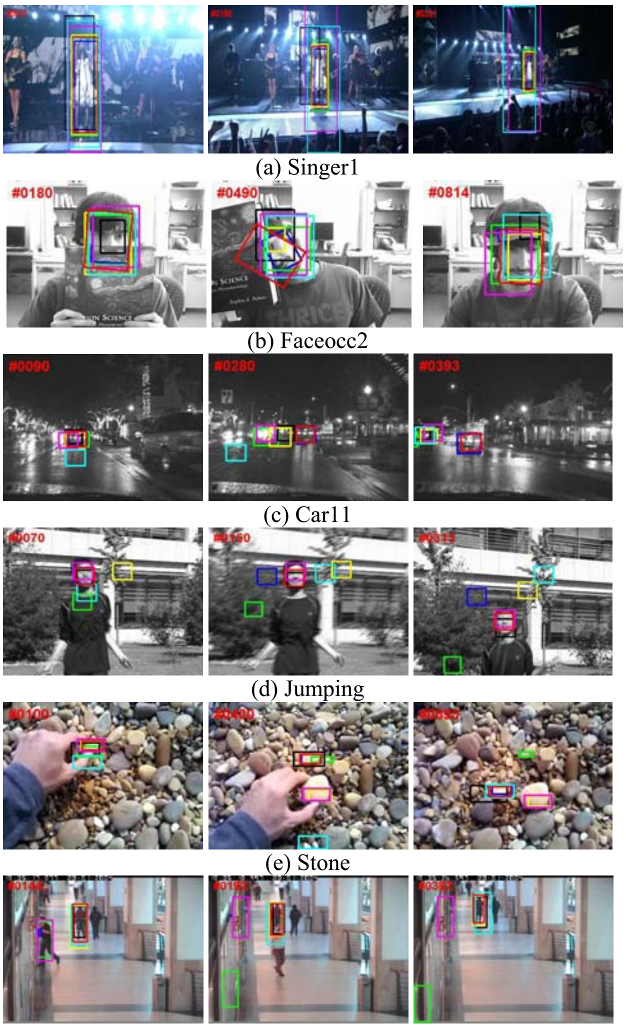
B. *Online Metric Learning*

Illumination change: Figure.2(a) presents the tracking results on sequences with dramatic illumination changes. In the singer1 sequence, the stage light changes drastically seen from frame 70 and frame 321. The PN tracking method [14] is not able to detect and track the target object frame 130. On the other hand, our tracker accurately locates the target object even when there is a large scale change at frame 321. Figure 2(c) presents the tracking results in the sequences with large illumination variation. In the Car 11 sequence, the contrast between the target and the background is low. The IVT tracker and our method perform well in tracking the vehicle while the other methods drift to the cluttered background or other vehicles when drastic illumination variation occurs. This can be attributed to the fact that our module introduces the backgrounds and the images with parts of the target as negative templates so the confidence values of these candidates calculated are small. Thus, the tracking result is accurately located on the true target without much offset. For the car11 sequence, there is low contrast between the foreground and the background as well as illumination change. The FragTrack method [2] fails at the beginning at frame 30. because it only uses the local information. The IVT tracking method achieves good results in this sequence. It can be attributed to the fact that subspace learning method is robust to illumination changes. In our algorithm we select several discriminative features which can better separate the target from the background. Thus, our tracker performs well in spite of the low contrast between the foreground and the background.

Occlusion: Occlusion is one of the most general yet crucial problems in object tracking. Figure 2(b) (f) (h) demonstrates how the proposed method performs when the target undergoes heavy occlusion or long-time partial occlusion. In the Faceocc2 sequence, the IVT tracking method [3], the PN tracking method [14] and the VTD tracking system [6] drift away from the target or do not scale well when the face is heavily occluded. In contrast, the FragTrack method [2], the MIL tracking algorithm [9], the L1 tracking method [19] and our tracker are developed to solve this problem effectively.

Background clutter: Figure.2 (e) presents the tracking results where the target objects appear in background clutters. The Stone sequence is challenging as there there are numerous stones of different shape and color. The FragTrack, MIL and VTD trackers drift to stones when the target is occluded whereas the IVT tracker and our method successfully keep track of the target throughout the sequence. The PN tracker (based on object detection with global search) is able to re-acquire the target again after drifting to the background, but with higher tracking errors and lower success rate.

Motion blur: The appearance change caused by motion blur in the jumping sequence is drastic that the Frag [2] and VTD [6] methods fail before frame 31. The IVT [3] method is able to track the target in some frames (such as frame 90) but fails when the motion blur occurs (e.g., frame 240). Our tracker successfully keeps track of the target object with small errors. The tracking experiment in animal sequence also verifies the advantage of our proposed tracked on the motion blur.The main reason is that we use the proposed module which separates the foreground from the background. Meanwhile, Eq. 1 assigns smaller weights to the candidate of background. Thus, the tracking result will not drift to the background.



(a) Singer1

(b) Faceocc2

(c) Car11

(d) Jumping

(e) Stone

(f) Caviar1

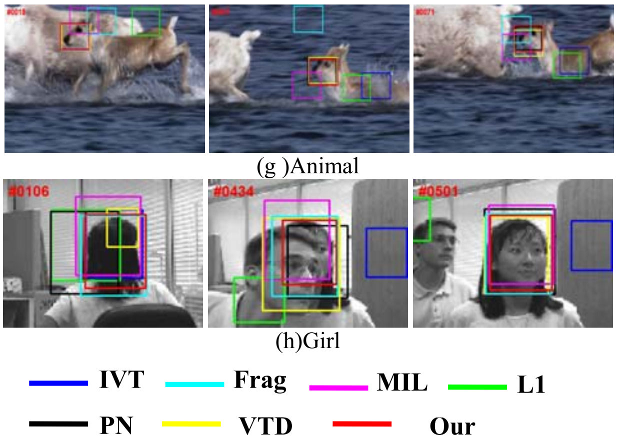


Figure 2. Tracking results of evaluated algorithms on six challenging image sequences

V. CONCLUSION

We propose an online learning target tracking method via online metric learning and non-sparse linear representation. Various challenging experiments show that the proposed tracker performs favorably against several state-of-the-art algorithms. Non-sparse linear representation make full use of the non-sparse coefficients to discriminate between the target and the background. Given a number of labeled data followed by a sequential input of unseen testing samples, the similarity metric is learnt by our model to maximum the margin between foreground and background samples. The pair-wise similarity is measured by our new bilinear graph for online label propagation the new data.

Acknowledge

This work has been funded by the Natural Science Foundation of China (grant no. 61203270) , the Introduction Foundation for the Talent of Nanjing University of Tele. and Com (no. NY212028).

REFERENCES

[1] M. J. Black. Eigen Tracking : Robust matching and tracking of articulated objects using a view based representation. IJCV, 26(1), pp: 63–84, 1998.

[2] A. Adam, E. Rivlin, and I. Shimshoni. Robust fragments-based tracking using the integral histogram. In CVPR, pp: 798–805, 2006.

[3] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental learning for robust visual tracking. IJCV, 77(1-3), pp:125–141, 2008.

[4] X. Mei and H. Ling. Robust visual tracking using L1 minimization. In ICCV, pp:1436–1443, 2009.

[5] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof. PROST: Parallel robust online simple tracking. In CVPR, pp: 723–730, 2010.

[6] J. Kwon and K. M. Lee. Visual tracking decomposition. In CVPR, pp: 269–1276, 2010.

[7] S. Avidan. Ensemble tracking. TPAMI, 29(2), pp: 261, 2007.

[8] H. Grabner and H. Bischof. On-line boosting and vision. In CVPR, pp: 260–267, 2006.

[9] B. Babenko, M.H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In CVPR, pp: 983–990, 2009.

[10] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. In ICCV, pp: 263–270, 2011.

[11] S.Wang, H. Lu, F. Yang, and M.-H. Yang. Superpixel tracking. In ICCV, pp: 1323–1330, 2011.

[12] H. Lu, S. Lu, D. Wang, S. Wang, and H. Leung. Pixel-wise spatial pyramidbased hybrid tracking. TCSVT, 22(9), pp:1365–1376, 2012.

[13] K. Zhang, L. Zhang, and M.-H. Yang. Real-time compressive tracking. In ECCV, pp: 864–877, 2012.

[14] Z. Kalal, J. Matas, and K. Mikolajczyk. P-N learning: Bootstrapping binary classifiers by structural constraints. In CVPR, 2010.

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

[16] X. Mei and H. Ling. Robust visual tracking using L1 minimization. In ICCV, 2009.

[17] C. Bao, Y.Wu, H. Ling, and H. Ji. Real time robust L1 tracker using accelerated proximal gradient approach. In CVPR, pp: 1830–1837, 2012.

[18] D. Wang, H. Lu, and M.-H. Yang. Online object tracking with sparse prototypes. TIP, 22(1), pp: 314–325, 2013.

[19] Dong Wang, Huchuan Lu, Minghsuan Yang, Least Soft-threshold Squares Tracking, CVPR2013.

[20] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via multitask sparse learning. In CVPR, pp: 2042–2049, 2012.

[21] B. Liu, J. Huang, L. Yang, and C. A. Kulikowski. Robust tracking using local sarse appearance model and k-selection. In CVPR, 2011.

[22] Wei Zhong, Huchuan Lu, Minghsuan Yang, Robust Object Tracking via Sparsity-based Collaborative Model, International Conference on Computer Vision and Pattern Recognition, pp:1838-1845, 2012.

[23] Xu Jia, Huchuan Lu, Minghsuan Yang, Visual Tracking via Adaptive Structural Local Sparse Appearance Model, International Conference on Computer Vision and Pattern Recognition, pp: 1822-1829, 2012.

[24] B. Liu, J. Huang, L. Yang, and C. Kulikowsk. Robust tracking using local sparse appearance model and k-selection. In CVPR, 2011.

[25] B. Liu, L. Yang, J. Huang, P. Meer, L. Gong, and C. Kulikowski. Robust and fast collaborative tracking with two stage sparse optimization. In ECCV, 2010.

[26] Q. Shi, A. Eriksson, A. van den Hengel, and C. Shen, Is face recognition really a compressive sensing problem? In CVPR, 2011.

[27] R. Rigamonti, M. A. Brown, and V. Lepetit, Are sparse representations really relevant for image classification? In CVPR, pp:1545-1552, 2011.

[28] L. Zhang, M. Yang, and X. Feng, Sparse representation or collaborative representation: Which helps face recognition? In ICCV, 2011.

[29] G. Tsagkatakis and A. Savakis, "Online distance metric learning for object tracking," IEEE Transactions on Circuits and Systems for Video Technology, pp. 1810–1821, 2011.

[30] N. Jiang, W. Liu, and Y.Wu, "Learning adaptive metric for robust visual tracking," IEEE Transactions on Image Processing, no. 99, pp. 1–1, 2010.

[31] X. Wang, G. Hua, and T. Han, "Discriminative Tracking by Metric Learning," ECCV, pp. 200–214, 2010.

[32] Xi Li, Chunhua Shen, Qinfeng Shi, Anthony R. Dick, Anton van den Hengel: Non-sparse linear representations for visual tracking with online reservoir metric learning. CVPR 2012: 1760-1767

[33] Yang Cong, Junsong Yuan, Yandong Tang: Object tracking via online metric learning. ICIP 2012: 417-420

[34] G. Chechik, V. Sharma, U. Shalit, and S. Bengio, "An online algorithm for large scale image similarity learning," NIPS, vol.21, pp. 306–314, 2009

[35] M. Everingham, L. Gool, C. Williams, J. Winn, A. Zisserman, The Pascal visual object classes challenge, International Journal of Computer Vision 88 (2), pp:303–338, 2010.