VISUAL TRACKING VIA MULTI-TASK NON-NEGATIVE MATRIX FACTORIZATION

Yong Wang, Xinbin Luo†, Shiqiang Hu†*

*Hisilicon Technologies
†School of Aeronautics and Astronautics, Shanghai Jiao Tong University

ABSTRACT

We propose an online tracking algorithm in which the object tracking is achieved by using subspace learning and non-negative matrix factorization (NMF) under the particle filtering framework. The object appearance is modeled by a non-negative combination of non-negative components learned from examples observed in previous frames. In order to robust tracking an object, group sparsity constraints are included to the non-negativity one. In addition, the Alternating Direction Method of Multipliers (ADMM) algorithm is proposed for efficient model updating. Qualitative and quantitative experiments on a variety of challenging sequences show favorable performance of the proposed algorithm against 9 state-of-the-art methods.

***Index Terms*—** non-negative matrix factorization, Alternating direction method of multipliers, subspace learning

1. INTRODUCTION

Object tracking plays a crucial role in numerous vision applications including human computer interaction, human activity analysis, traffic flow video processing, to name a few [1-10]. Tracking algorithms can be categorized as either generative [2-8] or discriminative [9-11] approaches. Generative tracking algorithms usually construct appearance models with image observations in offline or online settings. The tracking problem is formulated as searching for the region with the highest probability of being generated from the appearance model.

In this paper, a novel target representation is proposed based on the non-negative matrix factorization (NMF) [12]. Linear combinations of a set of non-negative basis are employed to model object appearance. The non-negative basis will efficiently capture the structure information of the target. In order to encode the characteristics in the tracking process, group sparsity is introduced. As shown in our experiments, our method is robust to illumination variation, pose change, background clutter and sever occlusion.

This work was jointly supported by the National Natural Science Foundation of China (No. 61374161) and China Aviation Science Foundation (No. 20142057006).

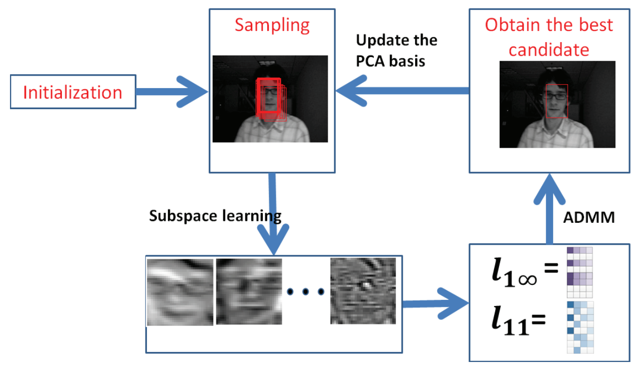

Fig. 1. The proposed tracking algorithm.

To apply our constrained NMF for visual tracking, we propose a tracking framework to capture the appearance property of the target. The workflow is shown in Fig.1. Our tracking algorithm is combined with the particle filtering and subspace learning framework. Specifically, given a new sample y in a new frame, the algorithm iteratively and alternatively updates basis matrix W and the approximation of y with respect to W. the likelihood of each particle is derived from its reconstruction error using the learned basis W. and the maximum of the likelihood is chosen to the target.

Compared with existing approaches, the contributions of this work are three fold. First, we represent the tracked object using PCA bases, taking advantages of the strengths of subspace representation. Second, we propose to use group sparsity NMF for visual tracking. To the best of our knowledge, this is the first time group sparsity NMF has been used for object tracking. The model that takes the inliers and noise into consideration is robust under different tracking scenarios. Third, we develop a novel algorithm based on Alternating direction method of multipliers (ADMM) method. In the experiments, the proposed tracking algorithm demonstrated superior performances in comparison with 9 stat-of-the-art methods.

2. NMF GROUP SPARSITY-BASED TRACKING

In this section, we first briefly introduce particle filtering framework that our tracker is formulated within. And then the subspace learning procedure is presented. Next we give a detailed description of both the principle and algorithm steps of our tracking model, followed by the implementation of the model using the ADMM algorithm.

2.1. Particle Filtering Tracking

In the particle filtering framework, there exist two fundamental steps: prediction and update. Let ss_t denote the state variable of the tracked object and y_t denote its corresponding observation in the t-th frame. Then the posterior probability can be recursively estimated by the following two rules:

$$p(ss_t|y_{1:t-1}) = \int p(ss_t|ss_{t-1})p(ss_{t-1}|y_{1:t-1})dss_{t-1} \quad (1)$$

$$p(ss_t|y_{1:t}) = \frac{p(y_t|ss_t)p(ss_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \quad (2)$$

where $ss_{1:t} = \{ss_1, ss_2, ..., ss_t\}$ stand for all available state vectors up to time t and $y_{1:t} = \{y_1, y_2, ..., y_t\}$ denote their corresponding observations. $p(ss_t|ss_{t-1})$ is a dynamic model that describes the state transition, and $p(y_t|ss_t)$ is an observation model that estimates the likelihood of observing y_t at state ss_t. The posterior $p(ss_t|y_{1:t})$ is approximated by K weighted particles.

2.2. Subspace Learning

At each instance t we model the tracking target and all candidates with k PCA basis vectors (W_t) and an error term (E_t) as:

$$X_t = W_t H_t + E_t \quad (3)$$

where $X_t \in R^{d*N}, X_t = [x_1, x_2, \cdots, x_N]$ is the observation vector, N is the number of observations, $W_t \in R^{d*k}$ denotes a matrix of PCA basis vectors (d represents feature dimension and k the number of PCA basis), $H_t \in R^{k*N}$ denotes the corresponding coding vectors (target coefficients), and $E_t \in R^{d*N}$ represents the error term. The most informative k orthogonal bases of the target subspace are composed to the PCA basis vectors to model the tracking target. We use the affine transformation to model the object motion between two consecutive frames.

2.3. NMF Representation

Recently the $L_{1\infty}$ norm has been proposed for joint regularization. Essentially, this type of regularization aims at learning a set of joint sparse models. The $L_{1\infty}$ norm is a matrix norm that penalizes the sum of maximum absolute values of each row. This regularizer encourages row sparsity: i.e., it encourages entire rows of the matrix to have zero elements. Therefore, we applied group sparsity penalty $L_{1\infty}$ to the row groups of the W_t in equation (3) to capture the shared features among all tasks over all particles. The L_1 loss penalty function is employed to penalize the differences between template and the noise. Thus, equation (3) can be re-written as the following form:

Algorithm1: group sparsity NMF tracking algorithm
Input: Current frame at t.
Dictionary template W_t.
All n particles ss_{t-1}.
1. Generate n particles ss_t within the particle filtering framework.
2. Compute imaging feature for each of the n particles and then form subspace matrix.
3. Obtain group sparse representation W_t and H_t by solving equation (4).
4. Calculate reconstruction error $\Delta r_i = \|x_i - w_i h_i - e_i\|_2$, $i = 1, ..., n$.
5. Calculate $p(y_i|ss_t) = \exp(-\Delta r_i^2)$ for each particle.
6. Select the particle with the highest value of $p(y_i|ss_t)$ as the current tracking result y_t.
Output: Tracked target y_t.
Current state ss_t.

Fig. 2. The proposed group sparsity NMF tracking algorithm.

$$\begin{aligned} & min_{W_t,H_t,E_t} \parallel X_t - W_t H_t - E_t \parallel_F^2 + \lambda_1 \parallel W_t \parallel_{1,\infty} \\ & \quad + \lambda_2 \parallel H_t \parallel_{1,1} + \lambda_3 \parallel E_t \parallel_{1,1}, s.t. W_t \geq 0, E_t \geq 0. \quad (4) \end{aligned}$$

where λ_1, λ_2 and λ_3 are tradeoff parameters controlling reliable construction of the observation, joint sparsity regularization and noise.

The novel of our tracking method is based on the NMF to implicitly combine holistic and part based methods. The target appearances are modeled as non-negative linear combinations of a set of non-negative basis that implicitly captures structure information. The row group sparsity which reflects the underlying assumption that target appearances across frames lies in the same subspace. Therefore, our representation model inherits the merits of nonnegativity constraint from NMF. In the following subsection we show that the solution to this problem can be obtained by performing a sequence of closed form optimization steps by the ADMM method. The detail of the ADMM algorithm can be found in [13] and [14].

2.4. Resolve Equation (4)

ADMM algorithm has been shown to be robust in machine learning applications and advantageous in resolving optimization problem of the sums of simple convex functions [7]. Resolving equation (4) in Algorithm 1 is essential to efficiently compute matrix B_t and W_t alternatively. We summarize our NMF group sparsity algorithm implemented by ADMM for resolving equation (4) in Fig.3.

3. EXPERIMENT

Performance of the proposed tracker has analyzed on 25 challenging video sequences and compared with seven state-of-the-art tracking works including the Incremental Visual

Algorithm 2: group sparsity algorithm implemented by ADMM.

Input: X, W, (t is omitted for clarity of the algorithm description in the following.)

Initialize H

While stopping criterion is not met do

$$W = \mathrm{argmin}_{W\geq 0}\left(\frac{1}{2}\|X - WH - E\|_F^2 + \langle \mu_1, W - U\rangle + \frac{\rho}{2}\|W - U\|_F^2\right)$$

$$U = \mathrm{argmin}_{U\geq 0}\left(\lambda_1\|U\| + \langle \mu_1, W - U\rangle + \frac{\rho}{2}\|W - U\|_F^2\right)$$

$$H = \mathrm{argmin}_{H\geq 0}\left(\frac{1}{2}\|X - BW - E\|_F^2 + \langle \mu_2, H - V\rangle + \frac{\rho}{2}\|H - V\|_F^2\right)$$

$$V = \mathrm{argmin}_{V\geq 0}\left(\lambda_2\|V\| + \langle \mu_2, H - V\rangle + \frac{\rho}{2}\|H - V\|_F^2\right)$$

$$E = \mathrm{argmin}\left(\frac{1}{2}\|X - BW - E\|_F^2 + \langle \mu_3, E - F\rangle + \frac{\rho}{2}\|E - F\|_F^2\right)$$

$$F = \mathrm{argmin}\left(\lambda_3\|F\| + \langle \mu_3, E - F\rangle + \frac{\rho}{2}\|E - F\|_F^2\right)$$

$$W_+ = \max(W + \rho * \mu_1, 0)$$
$$H_+ = \max(H + \rho * \mu_2, 0)$$
$$\mu_1 = \mu_1 + \rho(W - W_+)$$
$$\mu_2 = \mu_2 + \rho(H - H_+)$$

end while

Output: W_+, H_+.

Fig. 3. Implementation of the proposed NMF group sparsity learning algorithm using ADMM.

Tracking (IVT) [2], L1 tracking (L1T) [3], L1-APG tracking, multi-task tracking (MTT-L01, MTT-L21) [5], Multiple Instance Learning tracking (MIL) [9], compressive tracking (CT) [6], Wacv12 [10], WMIL [11], LSST [16], L2-RLS [15]. The sequences include either a nonrigid object or an object that undergoes significant appearance changes. The tracker was implemented in Matlab and runs at approximately 2 frames per second on an Intel Core i5. The trackers are run 3 times and the average results are reported for each video clip. We would like to emphasize that all the parameters were kept constant for all experiments.

3.1. Quantitative Comparison

The above-mentioned algorithms are evaluated using the center location error as well as the overlapping rate [18]. The average tracking errors are presented in Fig.4 where the best and results are shown with bold red fonts, and the second best ones are shown with blue fonts. The average overlapping errors are presented in Fig.5 where the best and results are shown with bold red fonts, and the second best ones are shown with blue fonts. The proposed tracking algorithm achieves the best or second best results in most sequences in terms of both success rate and center location error. Overall, the proposed tracker performs well against the other state-of-the-art algorithms.

3.2. Qualitative Comparison

Large pose variations with occlusions. In the basketball sequence, the player undergoes large pose variation and heavy occlusions. When the player is partially occluded by other similar players, the IVT, MIL, L1 and L1-APG methods do

	CT	IVT	L1-APG	L1	L2-RLS	MIL	MTT-L01	MTT-L02	WMIL	Ours
Basketball	16.3142	63.4504	63.3235	40.3348	33.4781	39.1368	44.8406	28.9425	14.47.7	8.211
Car11	3.8078	2.0304	1.9286	33.3288	2.7366	7.6508	2.2616	4.1108	96.9464	2.657
Caviar	68.5046	85.6776	24.8298	18.7307	144.9574	69.7605	65.2356	103.1512	88.6514	5.2573
Caviar1	16.8755	33.3381	48.4095	4.4955	1.3008	87.2633	53.4084	101.8416	29.531	1.4928
Caviar2	63.1670	13.3922	5.8703	3.4694	16.3851	22.6452	4.8253	10.4497	62.0663	3.0071
Cup	43.6092	1.6257	2.4408	2.7351	2.6194	40.9867	64.6413	159.8587	10.1017	1.6286
DavidIndoor	16.5161	67.5226	31.2135	221.8834	20.8459	23.1548	88.3262	19.5485	23.577	12.5193
Faceocc2	24.5455	63.7754	12.4189	153.9712	11.5001	21.4552	8.1904	29.6458	30.9168	9.819
Human	3.4695	359.7149	1.921	310.9163	393.4827	5.8683	3.2527	92.2673	16.3524	3.5852
Juice	6.7165	69.1284	0.9835	110.9964	6.1545	42.3063	3.4975	4.3433	10.4687	1.3031
Shirt	12.2288	111.6935	21.7862	299.1189	87.4274	22.4791	72.3955	205.4118	26.456	6.9631
Singer	19.2334	47.1161	5.098	386.1092	27.0821	23.1601	51.9938	132.2349	18.0611	4.1445
Ucsdpeds	5.3104	11.4702	1.7455	101.5581	62.797	10.9856	1.2932	4.6251	12.555	1.9478
Davidoutdoor	17.0244	259.9404	88.5216	458.5109	251.2946	70.7758	68.5676	482.0626	106.9907	5.3921
Fish	12.3659	33.1363	19.052	256.9916	38.1686	32.8666	39.8556	73.2485	52.5336	8.5455
Head_motion	15.2259	27.0431	9.437	8.3989	8.2647	9.8749	8.2459	9.4153	89.1344	8.0022
Mhyang	31.5107	51.5048	3.6666	251.1091	9.7712	53.9352	4.4252	19.2103	43.3887	3.7361
Ucup_on_table	13.8521	18.3712	1.5787	1.9819	2.5663	14.5659	1.8291	3.2201	17.5912	1.721
Uperson	10.4055	71.7153	68.6291	139.2425	2.8737	14.1563	12.9408	450.6377	90.8026	4.094
Uperson_partially_occluded	4.2774	2.3043	2.6337	2.6405	2.8452	45.4719	2.4118	2.9807	50.8289	2.5028
Chasing	12.8025	38.6676	4.9867	16.4593	5.8737	27.0968	6.7806	10.0162	9.3662	5.6147
Wball	7.0224	54.723	67.4685	154.9613	25.8186	23.4256	64.6334	38.8347	14.6955	6.9762
Wsurfing	10.6356	76.054	1.5356	1.7362	2.0272	5.5925	1.3556	4.5537	14.5308	4.0732
Xped1	51.3988	66.844	59.738	364.7516	10.3577	12.0172	64.2581	294.7471	45.1044	9.0738
Ycampus	32.3187	51.2778	2.8255	91.7948	7.2465	18.8763	24.3624	93.2914	56.1295	1.7744

Fig. 4. The average tracking errors. The error is measured using the Euclidian distance of two center points from the ground truth. The last row is the average error for each tracker over all the test sequences.

	CT	IVT	L1-APG	L1	L2-RLS	MIL	MTT-L01	MTT-L02	WMIL	Ours
Basketball	0.2634	0.0152	0.2662	0.0083	0.0938	0.2579	0.2497	0.0276	0.5234	0.5917
Car11	0.7201	0.6768	0.9211	0.5623	0.8295	0.3919	0.9135	0.7684	0.0025	0.8804
Caviar	0.158	0.146	0.22	0.452	0.024	0.162	0.156	0.15	0.138	0.994
Caviar1	0.3927	0.3089	0.3037	0.9215	1	0.0079	0.3037	0.2984	0.0288	1
Caviar2	0.27	0.302	0.914	0.992	0.342	0.036	0.982	0.424	0.012	0.994
Cup	0.4587	1	0.9967	1	1	0.4455	0.4752	0.1617	0.7294	1
DavidIndoor	0.2359	0.2273	0.2087	0.1991	0.2273	0.0606	0.2857	0.3701	0.2143	0.671
Faceocc2	0.5767	0.3877	0.4147	0.4515	0.7067	0.5607	0.9607	0.8613	0.5546	0.8859
Human	0.5049	0.0243	0.9078	0.2451	0.00218	0.4029	0.9976	0.2451	0.267	0.9927
Juice	0.4703	0.349	1	0.5	0.8861	0.0074	1	0.9926	0.4579	1
Shirt	0.7939	0.0126	0.6036	0.0063	0.0053	0.7056	0.0053	0.0053	0.2818	0.9832
Singer	0.2906	0.3447	0.4359	0.2393	0.0256	0.2222	0.3476	0.2821	0.2593	1
Ucsdpeds	0.5594	0.0383	1	0.0536	0.023	0.0575	0.8352	0.4751	0.0038	0.9962
Davidoutdoor	0.881	0.0198	0.3611	0.0159	0.0159	0.3968	0.3968	0.0397	0.3254	0.996
Fish	0.9139	0.2164	0.0735	0.0483	0.0651	0.1639	0.042	0.271	0.0357	0.5903
Head_motion	0.9489	0.6711	0.7545	0.9851	0.9728	1	0.983	0.9919	0.0694	1
Mhyang	0.002	0.294	0.9913	0.2416	0.7517	0.002	1	0.8483	0	0.9664
Ucup_on_table	0.1216	0.1863	1	1	1	0.1392	1	0.998	0	1
Uperson	0.7434	0.1837	0.4952	0.5111	0.9894	0.453	0.6315	0.0876	0.5322	0.9916
Uperson_partially_occluded	0.9082	0.9508	0.9934	0.9705	0.9377	0.0033	0.9541	0.9541	0.0033	0.9508
Chasing	0.1783	0.0667	0.64	0.7383	0.6717	0.005	0.705	0.6483	0.74	0.8283
Wball	0.799	0.0449	0.1196	0.2193	0.3206	0.0166	0.1096	0.1296	0.5249	0.799
Wsurfing	0.9326	0.0426	1	0.9965	0.9965	0.9397	1	0.578	0.3972	1
Xped1	0.5897	0.4316	0.2692	0.0085	0.5897	0.9615	0.5385	0.0983	0.2521	0.9615
Ycampus	0.6374	0.1374	1	0.1758	1	0.533	0.2747	0.1319	0.2747	1

Fig. 5. Average overlap rate. The best three results are shown in red, blue, and green fonts.

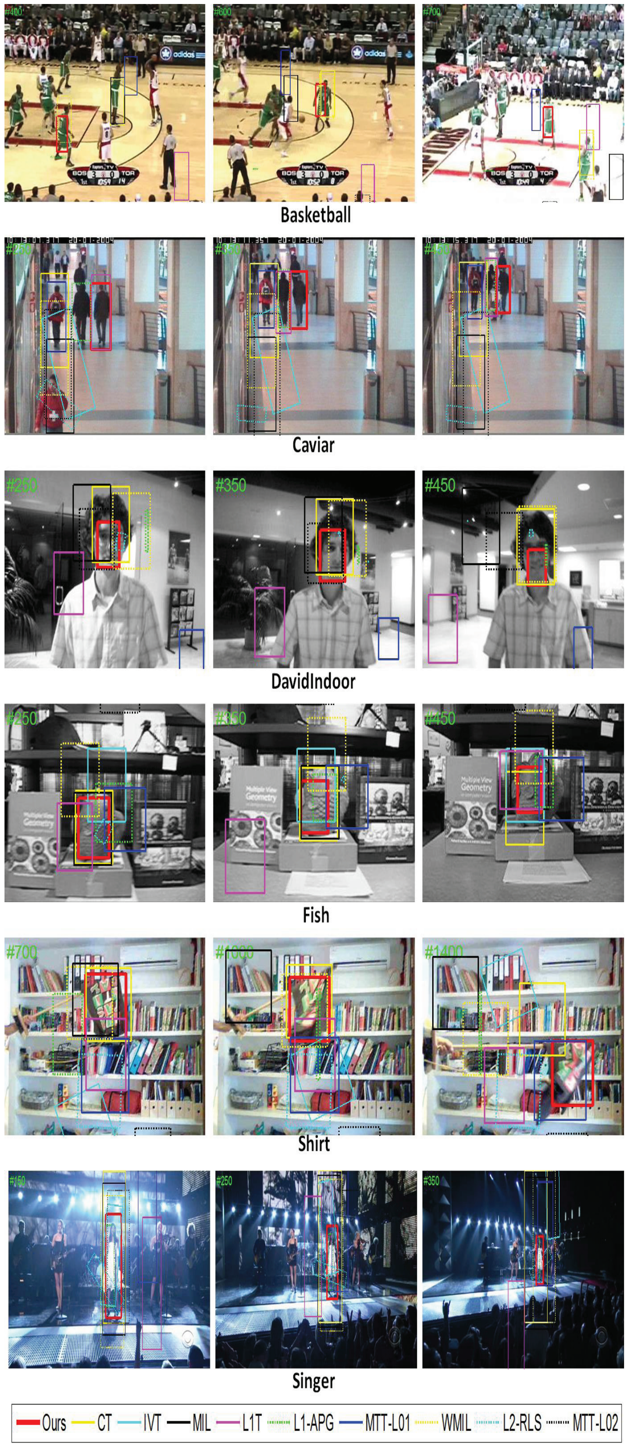

Fig. 6. Shows screenshots of some tracking results.

not perform well. Although, the WMIL track the object center location well, it cannot estimate the size of the objects. Our algorithm tracks the target objects reliably through the sequence.

Occlusions. The target in Caviar sequence undergoes heavy occlusions. In addition, the scale of the object in the caviar sequence changes significantly. The L2-RLS, MIL and MTT methods do not perform well when large scale change occurs. Due to significant scale changes in the caviar sequence, the CT shows limited tracking performance. When heavy occlusions occur, the WMIL and IVT methods start to drift away from the target object. On the other hand, our algorithm tracks the target objects well.

Illumination and pose variations: The objects in Singer and DavidIndoor sequences undergo large appearance changes due to illumination and pose variations. In the Singer sequence, the L1 methods do not perform well. The IVT, MIL, and L2-RLS approaches do not track the object reliably when illumination and pose variations occur together. In addition, the MTT-L01 and MTT-L02 methods do not perform well when scale and large illumination changes occur simultaneously. The WMIL does not deal with large scale changes well. Different from other tracking methods, our algorithm tracks the object favorably for various appearance changes.

Appearance changes: The target object in Shirt sequence undergoes various appearance changes including motion blurs, background clutter, and pose variations. When the target undergoes motion blurs, the L1 and MTT-L02 methods do not perform well. When background clutter occurs, the MTT-L01 and MIL methods drift away from the target objects. On the other hand, the IVT and WMIL methods fail to track the objects well when motion blurs occur. The CT and L1-APG methods do not perform well when large pose changes occur. In contrast, our algorithm performs well which can be attributed to use subspace learning and representation model to handle appearance changes.

Illumination and motion blur: The target objects undergo drastic illumination changes and motion blurs in Fish sequence. Most of trackers do not perform well. While our algorithm tracks the objects well due to the use of subspace model to update the template.

4. CONCLUSION

In summary, based on the subspace learning and NMF group sparsity constraint, we developed a robust tracking method which improved tracking accuracy. The accuracy improvement is achieved via a new object representation model for finding the sparse representation of the target. And it is solved by ADMM numerical solver. Numerous experimental results and evaluations demonstrate the proposed tracker performs favorably against existing state-of-the-art algorithms in the literature.

5. REFERENCES

[1] Yilmaz, A., Javed, O., Shah, M., ”Object tracking: A survey,” ACM Comput. Surv. 38(4),13C32, 2006.

[2] Ross, D., Lim, J., Lin, R.S., Yang, M.H., ”Incremental learning for robust visual tracking,” International Journal of Computer Vision, (IJCV) 77(1), 125C141, 2008.

[3] Mei, X., Ling, H., ”Robust visual tracking and vehicle classification via sparse representation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(11), 2259C2272, 2011.

[4] Zhang T, Ghanem B, Liu S, et al, ”Low-rank sparse learning for robust visual tracking,” ECCV 2012. Springer Berlin Heidelberg: 470-484, 2012.

[5] Zhang, T., Ghanem, B., Liu, S., Ahuja, N., ”Robust visual tracking via multi-task sparse learning,” In IEEE conference on computer vision and pattern recognition (pp. 1C8).

[6] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, ”Real-Time Compressive Tracking,” Proceedings of European Conference on Computer Vision (ECCV 2012), vol. 3, pp. 864-877, Florence, Italy, October, 2012

[7] C. Bao, Y. Wu, H. Ling, and H. Ji , ”Real Time Robust L1 Tracker Using Accelerated Proximal Gradient Approach,” IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Rhode Island, 2012.

[8] Y. Wu, B. Shen, and H. Ling, ”Visual Tracking via Online Non-negative Matrix Factorization,” IEEE Trans. on Circuits and Systems for Video Technology (T-CSVT), in press

[9] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie, ”Robust Object Tracking with Online Multiple Instance Learning,” IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 33, no. 8, pp. 1619-1632, 2011.

[10] Qing Wang, Feng Chen, Wenli Xu, Ming-Hsuan Yang, ”Online Discriminative Object Tracking with Local Sparse Representation,” IEEE Workshop on the Applications of Computer Vision, 425-432, 2012.

[11] Zhang K, Song H, ”Real-time visual tracking via online weighted multiple instance learning,” Pattern Recognition, 46(1): 397-411, 2013.

[12] D.D. Lee and H.S. Seung, ”Learning the parts of objects by non-negative matrix factorization,” Nature, 401(6755):788C791, 1999.

[13] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J, ”Distributed optimization and statistical learning via the alternating direction method of multipliers,” Found. Trends Mach. Learn., 3(1):1C122, 2011.

[14] Yong Wang, Shiqiang Hu, and Shandong Wu, ”Visual tracking based on group sparsity learning,” Machine Vision and Applications, pp: 1-13, 2014.

[15] Ziyang Xiao, Huchuan Lu, Dong Wang, ”L2-RLS-Based Object Tracking,” IEEE Trans. Circuits Syst. Video Techn. 24(8): 1301-1309, 2014.

[16] Dong Wang, Huchuan Lu, Ming-Hsuan Yang, ”Least Soft-Threshold Squares Tracking,” CVPR, 2371-2378, 2013

[17] X. Chen, W. Pan, J. Kwok, and J. Carbonell, ”Accelerated gradient method for multi-task sparse learning problem,” In IEEE international conference on data mining, pp. 746C751, 2009.

[18] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, ”The PASCAL Visual Object Classes Challenge,” (VOC2010) Results, 2010.