# ONLINE MULTI-OBJECT TRACKING BASED ON HIERARCHICAL ASSOCIATION AND SPARSE REPRESENTATION

*Zijian Lin, Huicheng Zheng, Bo Ke, and Lvran Chen*

School of Data and Computer Science, Sun Yat-sen University
Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China

## ABSTRACT

In recent years, sparse representation has been applied to multi-object tracking and shows promising performance. But existing methods often lead to considerable computation. In this paper, we propose a two-level hierarchical association approach to improve the accuracy and efficiency of online multi-object tracker based on sparse representation. We employ a time-saving affinity measure and a discriminative sparse representation classifier to handle objects with disparate and similar appearances, respectively. We also propose a novel strategy for track termination to protect the reliable tracks containing more detections and restrain the unreliable tracks at the same time. Experimental results demonstrate that the proposed method outperforms state-of-the-art online methods.

***Index Terms*—** Multi-object tracking, sparse representation, hierarchical association

## 1. INTRODUCTION

Multi-object tracking has potential applications in surveillance, intelligent transportation, robotics, etc. In recent years, multi-object tracking has achieved rapid development but it still faces some challenges such as occlusions and similar appearances. Due to the advance of object detection, most of methods adopt the tracking-by-detection strategy and formulate multi-object tracking as a data association problem.

The methods of multi-object tracking can be mainly divided into two categories, online approaches and offline approaches. Online approaches such as [1, 2, 3, 4], which associate existing tracks with detections frame by frame, have high practical utility. On the other hand, offline approaches such as [5, 6] based on global optimization usually show better performance. Recently, some approaches [7, 8] in between, which track online with a temporal delay, are proposed and compatible with both advantages.

Online multi-object tracking focuses on track-to-detection association. Affinity measure between tracks and detections, which is usually defined by appearance similarity, motion models, and so on [3, 9], is an important part of association. Due to the successful application of sparse representation in face recognition and single object tracking, sparse representation has also been applied to multi-object tracking [10, 11, 12, 13, 14]. These methods measure the affinity between a track and a detection by the residual error of object reconstruction over the track.

The affinity measures based on sparse representation show good performance, but spend much time on solving the sparse coefficients. In general scenarios, objects with no other similar objects around can be easily identified by using some simple affinity measures such as the distance between color histograms, so the stronger sparse representation classifier seems not always necessary. The methods mentioned above compute the sparse representations of all detections, and therefore lead to overall high computational complexity.

In the tracking-by-detection based online methods, tracks are allowed to be lost in a short time $\theta_{\Delta t}$ and extended when linked to a detection again. A larger $\theta_{\Delta t}$ makes data association span more frames and easily cause more mismatches, while a smaller $\theta_{\Delta t}$ would cause more frequent track interruptions. $\theta_{\Delta t}$ is manually set to be a fixed value in different scenes in [1, 9], which can not control well the balance.

There are two main contributions in this paper. The first one is a new two-level hierarchical association approach proposed to improve the efficiency of online multi-object tracking based on sparse representation. At the low-level association, an affinity measure with light computation is employed to handle the easily distinguishable detections. At the high level, we employ sparse representation for a discriminative classifier to handle only the confusing detections with similar appearances. Compared to traditional methods such as [12], we use not only spatial constraints but also a weaker affinity measure to handle the easily distinguishable detections in priority, which efficiently reduces the times of solving sparse coefficients and removes more distracters in dictionary to improve the classification accuracy. The second one is a dynam-

This work was supported by National Natural Science Foundation of China (No. 61172141), Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase), Project on the Integration of Industry, Education and Research of Guangdong Province (No. 2013B090500013), Science and Technology Program of Guangzhou (No. 2014J4100092), and Major Projects for the Innovation of Industry and Research of Guangzhou (No. 2014Y2-00213).

ic threshold proposed to determine whether a track should be terminated. A track containing more detections is considered more reliable and allowed to be lost in a longer period. This approach can be applied to other online multi-object trackers without conflict. Experimental results show that the proposed method is superior to other state-of-the-art online trackers.

## 2. PROPOSED METHOD

### 2.1. System overview

We follow the tracking-by-detection strategy. At each frame, objects of interest are detected and then linked to existing tracks by a two-level hierarchical association. In the hierarchical association, the easily identifiable objects are firstly associated to tracks based on a weak affinity measure and spatial constraints, and the confusing objects are associated later based on a stronger classifier. Tracks missing detections continuously for a long time are terminated and the detections failing in association are used to initialize new tracks.

Assume that at frame $t$, there are $n$ tracks $\{T_i^{t-1}\}_{i=1}^n$ and $m$ detections $\{d_j^t\}_{j=1}^m$. For simplicity, we omit the time index $t$ in the following sections. In our systems, an object $o$ is represented as $(f, \mathbf{p}, \mathbf{s}, \mathbf{c})$, where $f$ indicates when $o$ occurs; $\mathbf{p} = (x, y)$ and $\mathbf{s} = (w, h)$ are the center position and the size of the bounding box of $o$, respectively; $\mathbf{c}$ is the $l_1$-normed feature vector extracted from image of $o$. Each detection $d_j$, which has the same form as an object, is represented as $(t, \mathbf{p}_j, \mathbf{s}_j, \mathbf{c}_j)$. A track $T_i$ is represented as $(D_i, \mathbf{a}_i)$, where $D_i = \{o_k\}_{k=1}^{N_i}$ is the object sequence of $T_i$ and $N_i$ is the length of $D_i$; $\mathbf{a}_i$ is the feature vector of $T_i$.

### 2.2. Low-level association

In general, an object without other similar objects around is easy to be identified by appearance contrast and spatial constraints in a multi-object tracking task. At the low-level association, we aim to handle these easily identifiable objects efficiently. We search the best matching detection for each track. A track is seen as a candidate for its best matching detection. If a detection has only one candidate track, we link it to the track.

For track $T_i$, the best matching detection $\hat{d}_i$ is decided by

$$\hat{d}_i = \arg\max_{d_j} A(T_i, d_j) \quad s.t. \quad A(T_i, d_j) > \sigma_1, \tag{1}$$

where $A(T_i, d_j)$ is further defined as

$$A(T_i, d_j) = \varphi(T_i, d_j)\phi_v(T_i, d_j)\phi_s(T_i, d_j)\phi_m(T_i, d_j). \tag{2}$$

If there is no detection $d_j$ satisfying $A(T_i, d_j) > \sigma_1$, we consider that track $T_i$ fails in matching.

In Eq. 2, $\varphi(T_i, d_j)$ measures the affinity between $T_i$ and $d_j$. For efficiency and robustness, we define

$$\varphi(T_i, d_j) = \exp(-||\mathbf{a}_i - \mathbf{c}_j||_1). \tag{3}$$

$\phi_v(T_i, d_j)$, $\phi_s(T_i, d_j)$ and $\phi_m(T_i, d_j)$ are designed to cut off the relation between $T_i$ and $d_j$ based on the following considerations: (1) The velocity of an object is limited. (2) The size-changing rate of an object is limited. (3) The motion of an object is smooth in short time. We denote the last object of $T_i$ by $\tilde{o}_i$. Then,

$$\phi_v(T_i, d_j) = \delta\left(\frac{||\tilde{\mathbf{p}}_i - \mathbf{p}_j||_2}{\tilde{w}_i + w_j} < \theta_v(\tilde{f}_i - t) + \varepsilon_v\right), \tag{4}$$

$$\phi_s(T_i, d_j) = \delta\left(\frac{|\tilde{w}_i - w_j|}{\tilde{w}_i + w_j} < \theta_s(\tilde{f}_i - t) + \varepsilon_s\right), \tag{5}$$

where $\tilde{\mathbf{p}}_i$ and $\tilde{w}_i$ are the center position and the width of the bounding box of $\tilde{o}_i$, respectively; $\tilde{f}_i$ is the frame $\tilde{o}_i$ occurred in. $\delta(\cdot)$ is an indicator function which equals to 1 when the argument is true and 0 otherwise. $\theta_v$ and $\theta_s$ are the factors reflecting limited changing rates of position and size, respectively. Different from traditional methods [12, 14], $\varepsilon_v$ and $\varepsilon_s$ are constants to tolerate the error of bounding box.

$\phi_m(T_i, d_j)$ is the motion cue reflecting whether $T_i$ tends to $d_j$. Motion cue is usually used in affinity function [3, 9], but is not robust in the videos acquired by moving cameras. In order to adapt our tracker to the videos acquired by both static and moving cameras, we use the motion cue as a constraint. When $T_i$ contains only one object, there is no enough information to estimate the velocity of $T_i$, so we define $\phi_m(T_i, d_j) = 1$. Otherwise, we use Kalman filter to predict the dummy object $\overline{o}_i$ of $T_i$ at frame $t$ and

$$\phi_m(T_i, d_j) = \delta\left(\frac{Box(\overline{o}_i) \cap Box(d_j)}{Box(\overline{o}_i) \cup Box(d_j)} > \varepsilon_m\right). \tag{6}$$

In Eq. 6, $Box(o)$ indicates the bounding box of object $o$. If the intersection-over-union (IOU) between the predicted box and the detected box is less than $\varepsilon_m$, we consider that the track $T_i$ does not match $d_j$, which means they likely correspond to different objects.

### 2.3. High-level association

Some objects with similar appearances can be very confusing when moving together or intersecting each other. It is hard to distinguish them correctly by weak affinity measures and spatial constraints. After the low-level association, it is possible that some confusing detections are the best matching detections of more than one track. That is to say, some detections have more than one candidate track. We formulate the high-level association as a classification problem. Sparse representation has been demonstrated to be a very discriminative approach for recognizing confusing objects [12, 13]. Therefore, sparse representation classifier is adopted to discriminate confusing objects with more than one candidate track.

Given a detection $d_j$ and the set of its candidate tracks $\mathcal{T}_j = \{T_{j1}, T_{j2}, ..., T_{jK}\}$, the template dictionary $\mathbf{D}_j$ of $d_j$

consists of the objects of all tracks in $\mathcal{T}_j$ excluding the interpolated ones. But different from LSC in [12], the candidate tracks have been filtered by a motion cue and a weak appearance similarity in our approach so that some distracters have been removed and the elements of template dictionaries are reduced. Before formulating sparse representation, we normalize all the feature vectors to have unit $l_2$ norm. Let $\tilde{\mathbf{c}}_j$ and $\tilde{\mathbf{D}}_j$ be the normalized feature vector and dictionary, respectively. The sparse representation problem is formulated as

$$\boldsymbol{\alpha}_j = \arg\min_{\boldsymbol{\alpha}}(\frac{1}{2}||\tilde{\mathbf{c}}_j - \tilde{\mathbf{D}}_j\boldsymbol{\alpha}||_2^2 + \lambda||\boldsymbol{\alpha}||_1). \quad (7)$$

The label of $d_j$ is decided by

$$l_j = \arg\min_k ||\tilde{\mathbf{c}}_j - \tilde{\mathbf{D}}_j\xi_k(\boldsymbol{\alpha}_j)||_2, \quad (8)$$

where $\xi_k(\boldsymbol{\alpha}_j)$ means setting the coefficients unrelated to label $k$ to zero in $\boldsymbol{\alpha}_j$. We adopt fast DALM algorithm [15] to solve Eq. 7. Finally, we link $d_j$ to the track $T_{jl_j}$.

### 2.4. Track management

After association, we update the tracks according to the following rules. If a track $T_i$ is matched to a detection $d_j$, we add $d_j$ into $D_i$ and update $\mathbf{a}_i = (1-\beta)\mathbf{a}_i + \beta\mathbf{c}_j$, where $\beta \in (0,1)$ is a factor controlling the speed of feature updating. The missing objects at past frames in $D_i$ are estimated by linear interpolation.

Due to missing detections, we allow tracks to get lost in a short time $\theta_{\Delta t}$. If $\theta_{\Delta t}$ is too small, it is difficult to track objects for a long time, so ground-truth tracks could be easily broken into serval fragments with different identities. If $\theta_{\Delta t}$ is too large, a track tends to have a large search area and can be easily linked to an irrelevant detection. When either of the track and the detection is not the object of interest, more false objects could be generated by linear interpolation.

Since it is hard for a fixed $\theta_{\Delta t}$ to be adapted to these problems, we propose a novel dynamic threshold $\theta_{\Delta t}$ for track termination. For the track $T_i$ with $n_i$ detections in $D_i$ (excluding the interpolated objects), we set $\theta_{\Delta t,i} = \min(n_i + \delta_{t1}, \delta_{t2})$, where the parameter $\delta_{t1}$ is proposed to prevent termination of short tracks while $\delta_{t2}$ is an upper limit used to control the number of active tracks like the traditional fixed $\theta_{\Delta t}$. Let $B_i$ indicate the interval between frame $t$ and the frame of the last object in $D_i$. Once a track $T_i$ satisfies $B_i > \theta_{\Delta t,i}$, we terminate it. We consider a track with more detections would have more precise velocity estimation and more stable feature representation, so it is more reliable and less likely to be matched with an incorrect detection. Therefore, we allow tracks that are more reliable to be recovered after a longer interruption.

Finally, the detections failing in matching are used to initialize new tracks if they are not severely occluded by other matched detections in the same frame.

## 3. EXPERIMENTS

### 3.1. Experimental settings

In the experiments, we use the following datasets: S2L1 and S2L2 from PETS2009 [16], Town Center [17], and MOT Benchmark 2015 [18]. The first three sequences are acquired by elevated and static cameras. Compared to S2L1 and Town Center, S2L2 has more dense crowd and frequent object intersection. MOT Benchmark 2015 contains 11 test sequences acquired by static or moving cameras in different viewpoints.

We denote the proposed method by HASR (hierarchical association and sparse representation). For the baseline method, denoted by ASR (association based on sparse representation), we replace the association strategy in HASR with a greedy approach, i.e., to compute the affinity measure based on GSCR [12] instead of hierarchical association.

First of all, we test our methods on PETS2009 and Town Center to compare with other sparse representation based methods. Following [12, 14], we use the same detection results, ground-truth and evaluation tool based on CLEARMOT metrics [19]. Then we evaluate the effectiveness of hierarchical association and dynamic $\theta_{\Delta t}$ . Finally, we compare HASR with other state-of-the-art methods on MOT Benchmark.

The image of an object is resized to $96 \times 48$ pixels and then divided into 3 patches with $50\%$ overlap vertically. We compute a 32-dimensional histogram for each channel in the YCrCb and HS color spaces to form a 480-dimensional feature vector for each object and normalize it by $l_1$ norm. The parameters mentioned in Sec. 2 are set as follows: $\theta_v = 4/f_0$, $\varepsilon_v = 0.5, \theta_s = 1/f_0, \varepsilon_s = 0.3, \varepsilon_m = 0.2, \sigma_1 = 0.5, \lambda = 0.1$, $\beta = 0.1$, $\delta_{t1} = 4$, and $\delta_{t2} = 40$. $f_0$ represents the frame rate of videos. The proposed method is implemented in C++ and tested on a PC with an Intel Core i7-3770@3.4GHz CPU.

### 3.2. Results and analysis

The results in Table 1 demonstrate that HASR and ASR are superior to GSCR [12] and TH [14] in terms of MOTA (multi-object tracking accuracy) because some more reasonable constraints and track management help to remove distracters and improve the accuracy of association. In S2L1 and Town Center, TH and GSCR get lower IDs (identity switch) because mismatch likely appears as linking tracks to wrong detections and IDs dose not penalize it but they generate lots of false positives. In S2L2 with dense crowd, mismatch is mainly caused by linking two object with different ground-truth identities, so our methods show advantage in terms of IDs. Compared to ASR, HASR gets higher MOTA and significantly lower IDs in S2L2 thanks to hierarchical association. A weaker appearance similarity measure used at the low-level association does not make determinant effect on the confusing objects, but filters out some distracters probably influencing the sparse coefficients in dictionary, which may help to increase the accuracy of classification.

**Table 1**. Performance comparison between sparse representation based methods. ↑ represents the value higher is better and ↓ represents the value lower is better.

| Data | Method | MOTA↑ | IDs↓ | MOTP↑ | FP↓ | FN↓ |
|---|---|---|---|---|---|---|
| S2L1 | TH[14] | 70.1 | 21 | 71.7 | 543 | 827 |
| | GSCR[12] | 71.3 | **19** | 73.2 | 457 | 852 |
| | ASR | 87.2 | 24 | 75.4 | 110 | **460** |
| | HASR | **87.3** | 24 | **75.5** | **107** | 461 |
| S2L2 | TH[14] | 39.3 | 287 | 69.0 | 1416 | 4536 |
| | GSCR[12] | 43.9 | 194 | 71.1 | 1044 | 4514 |
| | ASR | 57.8 | 171 | 73.6 | 426 | **3743** |
| | HASR | **58.6** | **149** | **74.0** | **296** | 3821 |
| Town Center | TH[14] | 60.7 | 212 | 71.2 | 7295 | **20549** |
| | GSCR[12] | 61.3 | **192** | **71.6** | 3983 | 23476 |
| | ASR | **65.8** | 226 | 70.5 | 3030 | 21184 |
| | HASR | 65.7 | 216 | 70.6 | **2944** | 21338 |

**Table 2**. Comparison of the computational efficiency.

| Dataset | ASR | | HASR | | Speed |
|---|---|---|---|---|---|
| | TSSC | Speed(fps) | TSSC | Speed(fps) | up |
| S2L1 | 3989 | 9.7 | 135 | 69.2 | 7.1x |
| S2L2 | 5412 | 5.0 | 565 | 23.0 | 4.6x |
| Town Center | 46194 | 4.2 | 1454 | 25.3 | 6.0x |

In terms of efficiency, as shown in Table 2, HASR greatly decrease the times of solving the sparse coefficients (TSSC). In the scenes with sparse crowd such as S2L1, there are less similar objects getting together, so the TSSC decline more obviously. On the other hand, the weaker affinity measure has low time complexity $O(n)$, where $n$ is the dimension of feature vector. Therefore, the overall efficiency is improved considerably, so that the tracker can run in real time on a PC.

We next compare the performance of dynamic $\theta_{\Delta t}$ and fixed $\theta_{\Delta t}$ experimentally. As shown in Fig. 1, the best fixed $\theta_{\Delta t}$ is quite different although S2L1 and S2L2 correspond to the same scene. It seems that $\theta_{\Delta t} = 20$ is a balanced choice in terms of MOTA. On the other hand, dynamic $\theta_{\Delta t}$ is adaptive and maintains a satisfactory MOTA throughout different sequences. Furthermore, the IDs of dynamic $\theta_{\Delta t}$ is lower than that of a fixed $\theta_{\Delta t}$. The results show that dynamic $\theta_{\Delta t}$ is an effective strategy for track termination because it considers the reliability of tracks.

**Table 3**. Performance comparison with state-of-the-art methods on MOT Benchmark 2015. The symbol ⋆ means the method is online and the symbol † means the method is based on sparse representation.

| Method | MOTA ↑ | MOTP ↑ | MT (%)↑ | ML (%)↓ | IDs ↓ | Frag ↓ | Speed (fps)↑ |
|---|---|---|---|---|---|---|---|
| TC_ODAL⋆[3] | 15.1 | 70.5 | 3.2 | 55.8 | 637 | 1716 | 1.7 |
| GSCR⋆†[12] | 15.8 | 69.4 | 1.8 | 61.0 | 514 | 1010 | 28.1 |
| MDP⋆[1] | 30.3 | 71.3 | 13.0 | **38.4** | 680 | 1500 | 1.1 |
| NOMT[7] | 33.7 | 71.9 | 12.2 | 44.0 | 442 | 823 | 11.5 |
| SCEA⋆[2] | 29.1 | 71.1 | 8.9 | 47.3 | 604 | 1182 | 6.8 |
| LINF1†[13] | 24.5 | 71.3 | 5.5 | 64.6 | **298** | **744** | 7.5 |
| TSMLCDE[5] | **34.3** | **71.7** | 14.0 | 39.4 | 618 | 959 | 6.5 |
| RNN_LSTM⋆[4] | 19.0 | 71.0 | 5.5 | 45.6 | 1490 | 2081 | 165.2 |
| HASR⋆†(Ours) | 30.5 | 71.0 | **14.6** | 41.1 | 612 | 1585 | 34.3 |

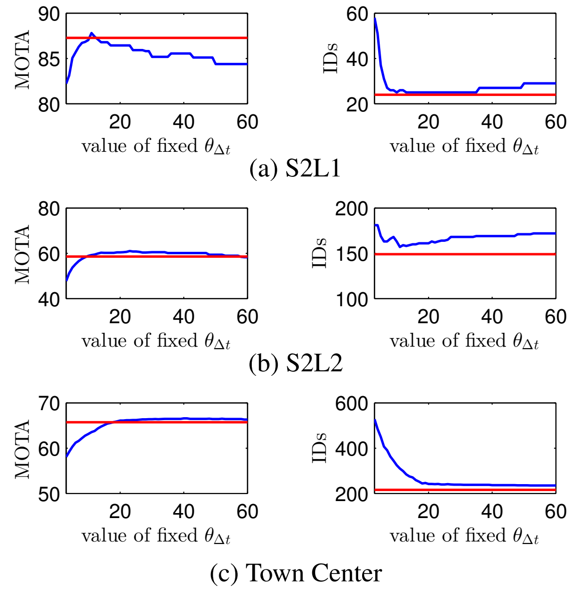


**Fig. 1**. MOTA and IDs performances of HASR with dynamic $\theta_{\Delta t}$ (red lines) or fixed $\theta_{\Delta t}$ with various values (blue curves).

Table 3 presents the results of the proposed method and some state-of-the-art methods on MOT Benchmark 2015. The processing speeds are tested under different hardware configuration and reported individually. The offline or near-online methods such as TSMLCDE [5] and NOMT [7] show better performances because they use global or future information for association. In the field of online methods, HASR is compatible with accuracy and efficiency. MDP [1] also gets high MOTA as ours but poor efficiency. RNN_LSTM [4] has a great efficiency because it does not use any visual information of detections and gets low MOTA. Furthermore, HASR needn't any extra labeled trajectories for training compared to MDP and RNN_LSTM. By comparison, the proposed method is competitive with state-of-the-art methods.

## 4. CONCLUSION

In this paper, we proposed a novel two-level hierarchical association for online multi-object tracking based on sparse representation. We showed that handling the easily identifiable objects by an affinity measure with light computation in priority can improve the efficiency significantly and the accuracy slightly. In addition, we proposed a novel dynamic threshold to determine whether a track should be terminated, which is more adaptive compared to a traditional fixed choice. The evaluations on challenging datasets show that the proposed method achieves state-of-the-art performance.

## 5. REFERENCES


[1] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *International Conference on Computer Vision*. IEEE, 2015, pp. 4705–4713.

[2] J. H. Yoon, C. R. Lee, M. H. Yang, and K. J. Yoon, "Online multi-object tracking via structural constraint event aggregation," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2016, pp. 1392–1400.

[3] S. H. Bae and K. J. Yoon, "Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 1218–1225.

[4] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," in *AAAI*, 2017.

[5] B. Wang, G. Wang, K. L. Chan, and L. Wang, "Tracklet association by online target-specific metric learning and coherent dynamics estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 589–602, 2017.

[6] B. Yang and R. Nevatia, "An online learned CRF model for multi-target tracking," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2034–2041.

[7] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *International Conference on Computer Vision*. IEEE, 2015, pp. 3029–3037.

[8] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *International Conference on Computer Vision*. IEEE, 2015, pp. 4696–4704.

[9] H. Jiang, J. Wang, Y. Gong, N. Rong, Z. Chai, and N. Zheng, "Online multi-target tracking with unified handling of complex scenarios," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3464–3477, 2015.

[10] M. A. Naiel, M. O. Ahmad, M. N. S. Swamy, Y. Wu, and M. H. Yang, "Online multi-person tracking via robust collaborative model," in *International Conference on Image Processing*. IEEE, 2014, pp. 431–435.

[11] W. Lu, C. Bai, K. Kapalma, and J. Ronsin, "Multi-object tracking using sparse representation," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 2312–2316.

[12] L. Fagot-Bouquet, R. Audigier, Y. Dhome, and F. Lerasle, "Online multi-person tracking based on global sparse collaborative representations," in *International Conference on Image Processing*. IEEE, 2015, pp. 2414–2418.

[13] L. Fagot-Bouquet, R. Audigier, Y. Dhome, and F. Lerasle, "Improving multi-frame data association with sparse representations for robust near-online multi-object tracking," in *European Conference on Computer Vision*. Springer, 2016, pp. 774–790.

[14] J. Zhang, L. L. Presti, and S. Sclaroff, "Online multi-person tracking by tracker hierarchy," in *International Conference on Advanced Video and Signal-Based Surveillance*. IEEE, 2012, pp. 379–385.

[15] A. Y. Yang, Z. Zhou, A. G. Balasubramanian, S. S. Sastry, and Y. Ma, "Fast $l_1$-minimization algorithms for robust face recognition," *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3234–3246, 2013.

[16] J. Ferryman and A. Shahrokni, "PETS2009: Dataset and challenge," in *International Workshop on Performance Evaluation of Tracking and Surveillance*. IEEE, 2009, pp. 1–6.

[17] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 3457–3464.

[18] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "MOT Challenge 2015: Towards a benchmark for multi-target tracking," *arXiv:1504.01942 [cs]*, Apr. 2015.

[19] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang, "Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 319–336, 2009.