Online Discriminative Structured Output SVM Learning for Multi-Target Tracking

Yingkun Xu, Lei Qin, Guorong Li, and Qingming Huang, *Senior Member, IEEE*

***Abstract*—In this letter, we propose an online discriminative learning method for feature combination during multi-target tracking. Previous works utilize offline learned weights for fusion of multiple features, which is not always effective for different tracking contexts. Our work aims to update the weights adaptively in online tracking. We formulate the feature combination problem in data association using structured output SVM, and solve it by online learning algorithm. The constraints of discriminative appearance affinity are integrated to discriminate positive associations from disturbing ones, which makes association more reliable. By comparison with five state-of-the-art methods, our proposed online tracking approach outperforms the other online methods, and is competitive with the global optimal ones.**

***Index Terms*—Multi-target tracking, online learning, structured output SVM.**

I. Introduction

Effective and efficient multi-target tracking is essential for visual surveillance, event detection, etc. However, it is still hard to implement reliable online real-time tracking in complex real-world scenarios. To approach this problem, detection based data association methods [1]–[12] are commonly used for multi-target tracking due to the rapid progress of object detection both in accuracy [13] and speed [14].

From the view of global optimization, object trajectories are always considered as partitions of detection responses. Maximization of posterior probability of the partitions can be solved by minimum-cost flows [1] or by conditional random fields (CRFs) with various constraints, such as motion dependency and occlusion dependency [2], difference between closely tracks [3], motion smoothness [4], and mutual exclusion among detections and trajectories [5]. By contrast with global optimization, online tracking approaches, which handle inputted detections frame by frame, are applicable more directly for time-critical applications. Generally, Bayesian framework is extended to implement the online tracking methods, for example, sparsity-promoted extended Kalman filtering [6], and particle filter method [7]. While these methods are suffered from the combinatorial explosion problem when the number of objects increases, bipartite matching methods provide a good approximation using well-designed affinity metrics. The affinity metrics are always defined as combination of weighted features. In the global optimization, the weights can be learned using HybridBoost [8] or RealBoost [9], [10]. In the online multi-target tracking, the SVM based methods [11], [12] are recently proposed to learn the weights using samples of detection pairs extracted beforehand. However, these offline-trained weights keep constant during tracking, and may not be appropriate for the specific tracking context.

In this letter, we propose an efficient structured output SVM method (SOSVM) to update the weights for affinity combination during online tracking. Unlike the method in [12] which uses SOSVM to learn the weights offline with cross validation dataset, in our method, the weights are learned incrementally and adaptively according to current tracking scenarios. Furthermore, we learn different weight vectors and thresholds for different objects with additional large margin regularization to strengthen the discrimination, whereas the method of [12] uses same weights and threshold for all objects. Fig. 1 shows the difference of tracking framework between [12] and our approach. Another method learning different weight vectors for different tracks is introduced in [10]. Different from our method, in that work, training samples are extracted from track segment (tracklet) pairs, and the weights are learned using RealBoost in a hierarchical manner.

The main contributions of our work lie in following points. First, we propose an adaptive SOSVM learning method for online real-time multi-target tracking. Second, we use different weight vectors for different objects, and large margin constraints are added to enlarge the distinguishability between different trajectories to promote the tracking performance.

II. Proposed Approach

In this section, we first propose an online SOSVM learning approach to update the feature weights adaptively, and then we introduce a large margin based regularization to strengthen the discrimination between different objects. Finally, we give the online tracking procedure with the adaptively learned weights.

A. Online Learning Structured Output SVM for Association

Denoting the set of trajectories generated before frame t as $R_{t-1} = \{r_1^{t-1}, r_2^{t-1}, \ldots, r_m^{t-1}\}$ and detection responses received at frame t as $O_t = \{o_1^t, o_2^t, \ldots, o_n^t\}$, our task is to find the best association between R_{t-1} and O_t as illustrated in Fig. 1(a), (c). Let $a(r_i^{t-1}, o_j^t) \in \mathbb{R}^D$ ($a_{i,j}^t$ in shorthand) denote D-dimensional affinity vector of multi-features between trajectory

Manuscript received November 04, 2013; revised December 16, 2013; accepted December 18, 2013. Date of current version January 08, 2014. This work was supported in part by National Basic Research Program of China (973 Program) under Grant 2012CB316400, in part by National Natural Science Foundation of China under Grants 61025011, 61003165, 61133003, 61332016, 61202234, and 61303153. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jeronimo Arenas-Garcia.

Y. Xu and L. Qin are with the Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lei.qin@vipl.ict.ac.cn).

G. Li and Q. Huang are with University of Chinese Academy of Sciences, Beijing 100049, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2013.2296602

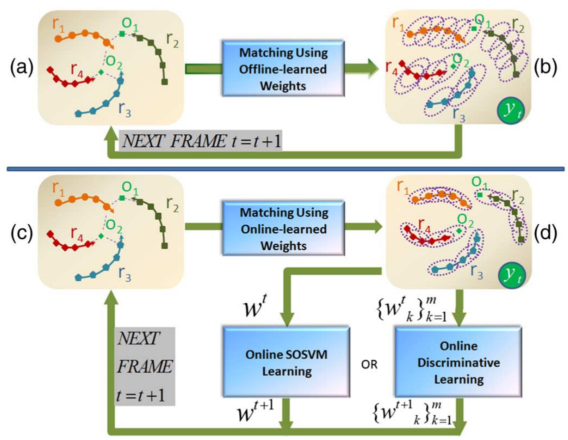

Fig. 1. The framework of online multi-target tracking using offline learned weights or online learned weights for association between detections and on-going trajectories. (a), (c). The candidate associations between detection responses o_1, o_2 and trajectories r_1, r_2, r_3, r_4 at frame t. (b). According to the offline learned constant weights, as shown by same size dashed ellipses, the erroneous association (r_1, o_1) may be obtained. (d). The linking errors can be reduced using online learned weights, which are overall weights w^t or trajectory-specific weights $\{w^t_k\}^m_{k=1}$, and updated using online SOSVM learning or online discriminative learning respectively. y_t is the association result at frame t.

r_i^{t-1} and detection o^t_j, the affinity between R_{t-1} and O_t can be expressed as $\Phi(R_{t-1}, O_t) = [a^t_{1,1}, a^t_{2,1}, \ldots, a^t_{1,n}, \ldots, a^t_{m,n}]^{\mathrm{T}}$. We use a binary vector $y = [y_{1,1}, y_{2,1}, \ldots y_{1,n}, \ldots, y_{m,n}]^{\mathrm{T}}$ to represent the associating result, where each element $y_{i,j} \in \{0, 1\}$ indicates the matching result of trajectory i and detection j. Because each trajectory can be associated at most once, and so does the detection response, there are constraints: $\sum_k y_{k,j} \leq 1$ for all j, and $\sum_k y_{i,k} \leq 1$ for all i. We define the set of all possible associations at frame t as $\mathcal{Y}_t$.

Similar as [12], we can formulate the problem of finding the best association result $y^\star$ as the structured output SVM:

$$y^\star = \arg\max_{y \in \mathcal{Y}_t} \quad \langle w, y^T \Phi(R_{t-1}, O_t)\rangle \tag{1}$$

where w is weight vector to combine D-dimensional affinities. Given the weights w, we can obtain the association result efficiently using Hungarian algorithm as in subsection II-C. Therefore, learning the weight w is the key to find correct matching results for trajectories.

In [12], the weight vector w is learned using standard SOSVM learning algorithm [15], where the samples are collected from tracking dataset of some other scenarios. However, as shown in Fig. 1 (b), it is not optimal to use these offline trained weights for each specific tracking context. Thus, different from [12], we employ online trained weights to promote the associating performance.

Some approaches are proposed for online training SOSVM, such as Larank [16] and stochastic sub-gradient descent method [17]. In our approach, we adopt the passive-aggressive (PA) method [18] to update the weight vector for two reasons. First, this method has a simple closed form solution, thus the learning process is fast. Second, the updating process is designed to tolerate the noisy labels. Therefore, it is a weak supervised online learning method in some sense, and thus suitable for the tracking task.

Different from standard SOSVM formulation in [15], given the previous weight w^t, the passive-aggressive algorithm aims to optimize the following objective function:

$$\begin{aligned} w^{t+1} &= \arg\min_{w} \quad \frac{1}{2}\|w - w^t\|^2 + C_1\xi^2 \\ s.t. \ & \max_{y \in \mathcal{Y}_t} \Delta(y, y_t) - \langle w, \delta y_t^T \Phi(R_{t-1}, O_t)\rangle \leq \xi \\ & \delta y_t^T \Phi(R_{t-1}, O_t) \equiv y_t^T \Phi(R_{t-1}, O_t) - y^T \Phi(R_{t-1}, O_t) \end{aligned} \tag{2}$$

where the loss function between augmented inferred association y and true association is defined as $\Delta(y, y_t) = (1 - y)^{\mathrm{T}} y_t$ [12]. Because tracking process has no ground truth labels, we use predicted association y_t inferred as in subsection II-C to approximate the true association label.

Using passive-aggressive solver, w is updated as:

$$w^{t+1} = w^t + \tau^t (y_t^T \Phi(R_{t-1}, O_t) - \tilde{y}^T \Phi(R_{t-1}, O_t)) \tag{3}$$

where:

$$\begin{aligned} \tau^t &= \frac{\Delta(\tilde{y}, y_t) - \langle w^t, y_t^T \Phi(R_{t-1}, O_t) - \tilde{y}^T \Phi(R_{t-1}, O_t)\rangle}{\|y_t^T \Phi(R_{t-1}, O_t) - \tilde{y}^T \Phi(R_{t-1}, O_t)\|^2 + 1/2C_1} \\ \tilde{y} &= \arg\max_{y \in \mathcal{Y}_t} \Delta(y, y_t) + \langle w^t, y^T \Phi(R_{t-1}, O_t)\rangle \end{aligned}$$

After this online updating for weight vector w, the structured loss is decreased as shown in experiments.

B. Discriminative Regularization for Online Association Learning

To promote distinguishability of associations and decrease confusion between similar trajectories, it is necessary to discriminate the affinity combination between similar or closer trajectories. We formulate this problem by weighting the appearance affinities differently for each trajectory. Let $\{w^t_k\}^m_{k=1}$ be weight vectors for trajectories at frame t, we can combine the slack variables for association loss and for large margin between confusing associations into one objective function as:

$$w^{t+1}_{k=1,\ldots,m} = \arg\min_{w_{k=1,\ldots,m}} \sum_{k=1}^{m} \frac{1}{2}\|w_k - w^t_k\|^2 + C_1\xi + C_2 \sum_{k=1}^{m}\sum_{i \neq k} \eta_{k,i} \tag{4}$$

$$\begin{aligned} s.t. \quad & \Delta(\tilde{y}, y_t) - \sum_{k=1}^{m} \langle w_k, \delta a(r_k^{t-1}, o^t_{s(y_t,k)})\rangle \leq \xi, \ \forall \xi \geq 0 \\ & \delta a(r_k^{t-1}, o^t_{s(y_t,k)}) \equiv a(r_k^{t-1}, o^t_{s(y_t,k)}) - a(r_k^{t-1}, o^t_{s(\tilde{y},k)}) \end{aligned} \tag{5}$$

$$\tilde{y} = \arg\max_{y \in \mathcal{Y}_t} \Delta(y, y_t) + \sum_{k=1}^{m} \langle w^t_k, a(r_k^{t-1}, o^t_{s(y,k)})\rangle \tag{6}$$

$$\begin{aligned} & \langle w_k, a(r^t_k, o^t_{s(y_t,k)})\rangle - \langle w_i, a(r^t_i, o^t_{s(y_t,k)})\rangle \geq 2\mu - \eta_{k,i} \\ & \forall i : i \neq k, \ \eta^t_{k,i} \geq 0 \end{aligned} \tag{7}$$

where $s(y_t, k)$ is the index of detection response which is matched to trajectory k under association y_t. For appearance features, the vector $a(r_i^{t-1}, o^t_{s(y_t,k)})$ extracts the affinities between track r_i^{t-1} and its matched detection result $o^t_{s(y_t,k)}$.

The slack variable ξ is used to bound the overall association errors similar as in equation (2). By comparison with basic passive-aggressive algorithm in previous subsection, using different weight vectors for different trajectories is more flexible,

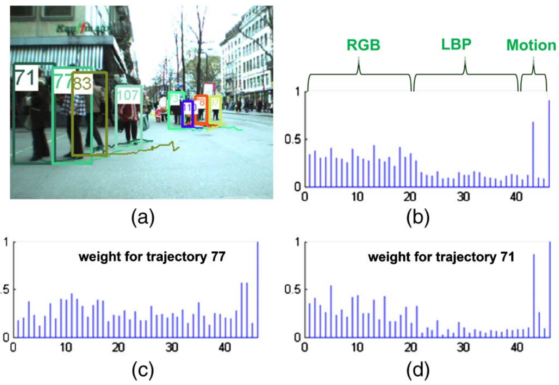

Fig. 2. Illustration of the weights online learned during multi-target tracking. (a) The current tracking scenario in frame 325 of ETHMS bahnnof sequence. (b) The weights learned with online learning method in subsection II-A. (c), (d) The weights of trajectories 77 and 71 learned with discrimination-regularized online method in Section II-B. The weights of trajectory 77 are larger in LBP parts than trajectory 71, which coincides with its distinctive textural features. Whereas the weights in (b) reflect that RGB features are more important than LBP features on the whole.

because the termination of one trajectory does not affect the weight vectors of the other objects.

The slack variable $\eta_{k,i}$ is used to penalize confusing similarity of appearance features between trajectory k and trajectory i. This penalization aims to enlarge margin between positive association and negative association. The hyper-parameter μ should make the regularization more effective than the basic constraint of equation (5). Thus, it is chosen to be somewhat larger than 1. We set its value to 1.2, and this value is appropriate for different datasets after validation.

To simplify the updating of w_k, we adopt one slack method similar as in [15], and combine the constraints (5), (6) and (7) into one slack constraint for each trajectory k. We can obtain the updated w_k using passive-aggressive solver-I [18]:

$$w_k^{t+1} = w_k^t + \kappa^t (C_1 \delta a(r_k^{t-1}, o_{s(y_t,k)}^t) + C_2 \delta a(r_k^t, o_j^t)) \quad (8)$$

where:

$$\begin{aligned}
&\kappa^t = \min\{1, \\
&\quad \frac{\Delta_s(y_t, \tilde{y}, k) - \langle w_k^t, C_1 \delta a(r_k^{t-1}, o_{s(y_t,k)}^t) + C_2 \delta a(r_k^t, o_j^t)\rangle}{\left\| C_1 \delta a(r_k^{t-1}, o_{s(y_t,k)}^t) + C_2 \delta a(r_k^t, o_j^t)\right\|^2}\} \\
&\Delta_s(y_t, \tilde{y}, k) \equiv C_1 I(s(y_t,k) \neq s(\tilde{y},k)) + 2\mu C_2 \\
&\tilde{y} = \arg\max_{y \in \mathcal{Y}_t} \Delta(y, y_t) + \sum_{k=1}^{m} \langle w_k^t, a(r_k^{t-1}, o_{s(y,k)}^t)\rangle \\
&\delta a(r_k^t, o_j^t) \equiv \min_{j \neq s(y_t,k)} a(r_k^t, o_{s(y_t,k)}^t) - a(r_k^t, o_j^t)
\end{aligned} \quad (9)$$

where I is indicator function.

It can be seen that the weight is updated along the direction of right association and better generalization. These two aspects are balanced by the parameter C_1 and C_2. From this method, we can obtain the new affinity weight for association in the next frame. Fig. 2 illustrates an example of weights learned using the online discriminative method and non-discriminative method described in the previous subsection.

C. Online Tracking Procedure with Adaptive Weight Vector

When the weights for feature combination are updated, it is important to obtain thresholds to decide whether the candidate associations are positive. Different from [12], we use the average value of the two most similar weighted affinities as the threshold for each trajectory k like the bias in binary classifier:

$$\begin{aligned}
&\theta_k^{t+1} = \frac{1}{2}(\langle w_k^{t+1}, a_{k,s(y_t,k)}^t\rangle + \langle w_i^{t+1}, a_{i,s(y_t,k)}^t\rangle) \\
&\text{where}: \quad i = \arg\max\nolimits_{i \neq k} \langle w_i^{t+1}, a(r_i^{t-1}, o_{s(y_t,k)}^t)\rangle
\end{aligned} \quad (10)$$

This threshold can balance the extendibility of association and its correctness as in experiments. Combing the affinity vector $a_{i,j}^t$ and the threshold θ_i^t, we extend the equation (1) to construct a square affinity matrix Ψ of order $\max(m,n)$ for m trajectories and n detection responses. The element of this matrix is defined as:

$$\psi(i,j) = \begin{cases} \langle w_i^t, a_{i,j}^t\rangle & \text{if } \langle \mathrm{w_i^t, a_{i,j}^t}\rangle \geq \theta_\mathrm{i}^\mathrm{t}, \mathrm{i} \leq \mathrm{m}, \mathrm{j} \leq \mathrm{n} \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

Using Hungarian algorithm, we can obtain the association result y_t for trajectories and detection responses by non-zero entries in Ψ. The associated trajectories are extended to the locations of their matched detections. The un-associated detections are used to create new trajectories. Through the predicted association y_t, the weights are updated as in previous two subsections. We call the tracking method using subsection A as PAMTT, and that using subsection B as DPAMTT.

To remove the false-alarm detections, we classify the trajectories into two types: the candidate trajectories and the confident trajectories. The new trajectories are considered as candidate ones, which will be confident if they are associated continuously more than ΔT_s frames, or will be deleted if they are unassociated for ΔT_f frames. To handle occlusions, we set another threshold ΔT_e. The confident trajectories which are unmatched will be terminated if they are missed continuously longer than ΔT_e, or will be predicted by Kalman filter unless they are out of image boundary. The trajectories which are unterminated and undeleted are used to associate in next frame.

III. Experiments

We evaluate our proposed approach on three public datasets. The first is PETS09 S2.L1 sequence, the second is ETHMS dataset including two sequences bahnnof and sunny day, and the third is TUD Stadtmitte sequence. These sequences contain different challenging problems, such as illumination changes and heavy occlusions. For fair comparison, we use common inputted detection results, ground truth of trajectories, and evaluation tool published in [3], [19]. In our experiments, the features for association affinity are composed by $8 \times 8 \times 8$ RGB histogram, LBP and motional information. Different from [12], the affinities of color and LBP features are computed from 21 randomly selected blocks within the object regions, and the motional affinities are computed using velocity changing, position distance, Bhattacharyya distance of optical flow histogram, and bounding box overlapping ratio. The hyper-parameters C_1, C_2 are set to 0.001, 0.01. The length thresholds ΔT_s, ΔT_f and ΔT_e are 6, 6 and 32 respectively.

To examine the effectiveness of our algorithm, we firstly compare with the baseline method using offline trained weights [12]. We employ the augmented structure loss of SOSVM as the metric to illustrate the advantages of our method. The loss at frame t is defined as:

$$\Delta(\bar{y}_t, y_t) + \langle w_t, y_t^T \Phi(R_{t-1}, O_t) - \bar{y}_t^T \Phi(R_{t-1}, O_t)\rangle$$

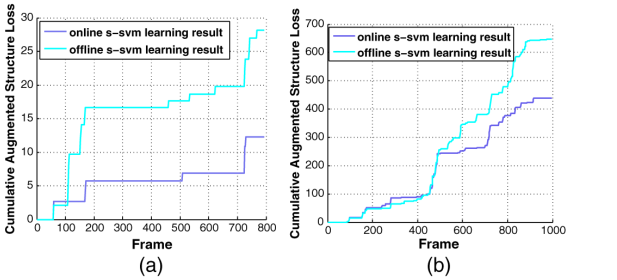

Fig. 3. The cumulative augmented structure loss of association results using online method PAMTT in Section II-A and offline SOSVM method [12].

where y_t is predicted association and $\bar{y}_t$ is real association extracted from tracking ground truth. This loss can be considered as the error measure to evaluate the performance of SOSVM learning [15].

Because our method is incremental learning method, the errors may be accumulated and become serious. Thus, rather than the loss of each frame, we use the cumulative augmented structure loss over the past frames to illustrate the comparing results more clearly. We show the results of two sequences PETS09 S2.L1 and ETH bahnnof in the Fig. 3(a), (b) respectively. The results illustrate that the PAMTT method has lower error ratio than the offline learning method, and the erroneous matching does not cause consequential errors.

To assess the performance of our proposed method quantitatively, we use VACE evaluation metrics [3], [12]. Because these metrics are somewhat redundant, we select seven representative metrics: recall of correctly matched objects (Recall), precision of correctly matched objects (Precision), harmonic mean of precision and recall (F-score), ratio of mostly tracked trajectories (MT), ratio of partial tracked trajectories (PT), fragments (FM), and ID switches (IDS). These metrics reflect the performance in the aspects of trajectories and objects.

Besides the baseline method [12], we compare with four other state-of-the-art multi-target tracking methods: online learned CRF model [3], online learned motion and appearance model [19], continuous energy minimization [4], and multi-level exclusion based method [5]. These methods tend to obtain the globally optimized association results within temporal sliding windows. To compare with standard SOSVM method [15] further, we implement a simple online SOSVM learning based tracking with samples collected along the tracking process, and we call it BOMTT. To show the effectiveness of regularization (7), we also verify DPAMTT method without constraint (7) by setting C_2 to zero, and we call it SPAMTT.

Table I illustrates the results of comparison. By contrast with the globally optimized data association methods, our methods have competitive results, except some extra drifting problems compared with methods [3], [19], which have lower IDS than ours. It is largely because the scenario information and future observations are used in those methods. However, despite of several frames which are used to confirm whether the new trajectories are false-alarming or not, our methods employ the online tracking framework. Whereas the global optimization methods use the sliding window tracking paradigm, thus there exists long latency.

By contrast with the offline trained SOSVM [12] and BOMTT, the results of our proposed method are more accurate. The F-scores for tracked objects are higher, and the sums of FM and IDS are less in all attributing to well-balanced extendibility and rightness of association. By comparison with PAMTT and SPAMTT, the performance of DPAMTT is promoted with slightly less FM, IDS, and higher precision, which indicates more reliable tracking results. Fig. 4 shows some examples of tracking results using DPAMTT.

TABLE I
COMPARISON WITH STATE-OF-THE-ART METHODS

Datasets	Methods	Recall	Precision	F-score	MT	PT	FM	IDS
PETS	Kim[12]	97.2%	93.7%	0.95	94.7%	5.3%	19	4
09-	Yang[19]	91.8%	99.0%	0.95	89.5%	10.5%	9	0
S2-L1	Milan[5]	—	—	—	94.7%	5.3%	15	22
	Milan[4]	92.4%	98.4%	0.95	91.3%	4.3%	6	11
	BOMTT	95.8%	94.2%	0.95	94.7%	5.3%	7	5
	PAMTT	97.2%	95.6%	0.96	94.7%	5.3%	5	3
	SPAMTT	96.0%	97.3%	0.97	94.7%	5.3%	6	5
	DPAMTT	96.1%	97.5%	0.97	94.7%	5.3%	3	2
ETHMS	Kim[12]	78.4%	84.1%	0.81	62.7%	29.6%	72	5
	Yang[3]	79.0%	90.4%	0.84	68.0%	29.0%	19	11
	Milan[5]	77.3%	87.2%	0.82	66.4%	25.4%	69	57
	BOMTT	80.8%	83.4%	0.82	64.9%	26.6%	45	52
	PAMTT	81.8%	84.1%	0.83	68.1%	24.5%	32	37
	SPAMTT	77.6%	86.7%	0.82	66.0%	26.6%	31	28
	DPAMTT	79.3%	87.6%	0.83	67.0%	26.6%	27	21
TUD	Kim[12]	80.0%	96.7%	0.88	80.0%	20.0%	11	0
Stadt-	Yang[3]	87.0%	96.7%	0.92	70.0%	30.0%	1	0
mitte	Milan[5]	—	—	—	40.0%	60.0%	13	15
	Milan[4]	84.7%	86.7%	0.86	77.8%	22.2%	3	4
	BOMTT	88.1%	94.1%	0.91	80.0%	20.0%	7	3
	PAMTT	91.1%	95.7%	0.93	80.0%	20.0%	5	3
	SPAMTT	87.6%	96.0%	0.92	80.0%	20.0%	5	4
	DPAMTT	90.6%	96.3%	0.93	80.0%	20.0%	4	2

(a)
(b)
(c)


Fig. 4. Tracking examples for (a) PETS09, (b) ETHZ bahnnof and (c) TUD.

The platform of our experiments is Intel E7300 of 2.66 GHz with 2 GB ROM. We implement the code except object detection using C++ with the average running speed of 28FPS.

IV. CONCLUSION

In this letter, we develop an online discriminative SOSVM learning method for feature combination during multi-target tracking. The experimental results show the advantages of the method over the offline learned feature weighting, and the final results are competitive with those of global optimization methods. In future work, we tend to consider more primal image information to promote the tracking performance, and reduce the possible erroneous associations, which are caused by lack of observations in later frames.

REFERENCES

[1] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2008.

[2] B. Yang, C. Huang, and R. Nevatia, "Learning affinities and dependencies for multi-target tracking using a crf model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2011.

[3] B. Yang and R. Nevatia, "An online learned crf model for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2012.

[4] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multi-target tracking," *IEEE Trans. Patt. Anal. Mach. Intell.*, 2013.

[5] A. Milan, K. Schindler, and S. Roth, "Detection- and trajectory-level exclusion in multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2013.

[6] E. Masazade, M. Fardad, and P. K. Varshney, "Sparsity-promoting extended kalman filtering for target tracking in wireless sensor networks," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 845–848, 2012.

[7] M. R. Morelande, M. K. Christopher, and K. Keith, "A bayesian approach to multiple target detection and tracking," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 1589–1604, 2007.

[8] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: Hybridboosted multi-target tracker for crowded scene," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2009.

[9] C. H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2010.

[10] C. H. Kuo and R. Nevatia, "How does person identity recognition help multi-person tracking?," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2011.

[11] X. Yan, X. Wu, I. A. Kakadiaris, and S. K. Shah, "To track or to detect? An ensemble framework for optimal selection," in *ECCV*, 2012.

[12] S. Kim, S. Kwak, J. Feyereisl, and B. Han, "Online multi-target tracking by large margin structured learning," in *ACCV* , 2012.

[13] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, 2012.

[14] R. Benenson, M. Mathias, R. Timofte, and L. V. Gool, "Pedestrian detection at 100 frames per second," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2012.

[15] T. Joachims, T. Finley, and C. N. J. Yu, "Cutting-plane training of structural SVMs," *Mach. Learn.*, vol. 77, no. 1, pp. 27–59, 2009.

[16] A. Bordes, L. Bottou, P. Gallinari, and J. Weston, "Solving multiclass support vector machines with LaRank," in *Proc. 24th Int. Conf. Machine Learning*, 2007.

[17] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal estimated subgradient solver for SVM," in *Proc. Int. Conf. Machine Learning*, 2007.

[18] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *J. Mach. Learn. Research*, pp. 551–585, 2006.

[19] B. Yang and R. Nevatia, "Multi-target tracking by online learning of non-linear motion patterns and robust appearance models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2012.