This CVPR2015 paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Target Identity-aware Network Flow for Online Multiple Target Tracking

Afshin Dehghan[1] Yicong Tian[1] Philip. H. S. Torr[2] Mubarak Shah[1]

[1]Center for Research in Computer Vision, University of Central Florida
[2]Department of Engineering Science, University of Oxford
adehghan@cs.ucf.edu, ytian@crcv.ucf.edu, philip.torr@eng.ox.ac.uk, shah@crcv.ucf.edu

Abstract

In this paper we show that multiple object tracking (MOT) can be formulated in a framework, where the detection and data-association are performed simultaneously. Our method allows us to overcome the confinements of data association based MOT approaches; where the performance is dependent on the object detection results provided at input level. At the core of our method lies structured learning which learns a model for each target and infers the best location of all targets simultaneously in a video clip. The inference of our structured learning is done through a new Target Identity-aware Network Flow (TINF)*, where each node in the network encodes the probability of each target identity belonging to that node. The proposed Lagrangian relaxation optimization finds the high quality solution to the network. During optimization a soft spatial constraint is enforced between the nodes of the graph which helps reducing the ambiguity caused by nearby targets with similar appearance in crowded scenarios. We show that automatically detecting and tracking targets in a single framework can help resolve the ambiguities due to frequent occlusion and heavy articulation of targets. Our experiments involve challenging yet distinct datasets and show that our method can achieve results better than the state-of-art.*

1. Introduction

Multiple Object Tracking (MOT) is a fundamental problem in computer vision with numerous applications, ranging from surveillance, behavior analysis, to sport video analysis. Most of the recent approaches that aim to solve the MOT problem follow two main steps: Object Detection and Data Association. In the detection phase, a pre-trained object detector is first applied to find some potential object locations in each frame of a video. Once the object candidates are found, in the data association phase the candidates are pruned and tracks between images are formed. In most previous work, these two steps have been considered as two separate problems and the focus of tracking is mostly on designing data association techniques. There are two main classes of data association.

Local Association. These methods are temporally local, which means they consider only a few frames while solving the association problem. The best example of such approaches is bi-partite matching and its extensions [23, 25, 27, 5]. In [5], the association probabilities are computed jointly across all targets to deal with ambiguities in association. Shu et. al in [27] use a greedy approach to combine the responses of part detectors to form a joint likelihood model of multiple cues to associate detections and object hypotheses. Whilst this class of methods are computationally inexpensive, their assumption of using few frames makes them prone to ID-switches and other difficulties in tracking such as long/short term occlusions, pose changes and camera motion.

Global Association. To better deal with above problems, another set of data association techniques have recently received a lot of attentions. In global association methods, the number of frames is increased and sometimes the entire video is processed at once to infer the tracks [37, 13, 35]. Recent approaches have formulated the data association as a network flow problem where a set of tracks are found efficiently by solving min-cost flow.

Different solutions to minimum cost flow for MOT have been proposed recently. In [38] a global optimal solution is found using push-relabel algorithm. Pirsiavash *et al*. in [22] utilize the same graph as [38], and solve the problem using a fast greedy shortest path procedure based on dynamic programming. Berclaz *et al*. in [6] introduce an efficient shortest path algorithm to solve the flow problem. In [9], a new network flow is proposed to incorporate constant velocity motion model in the graph and the solution is found efficiently using Lagrange relaxation. Shitrit *et al*.

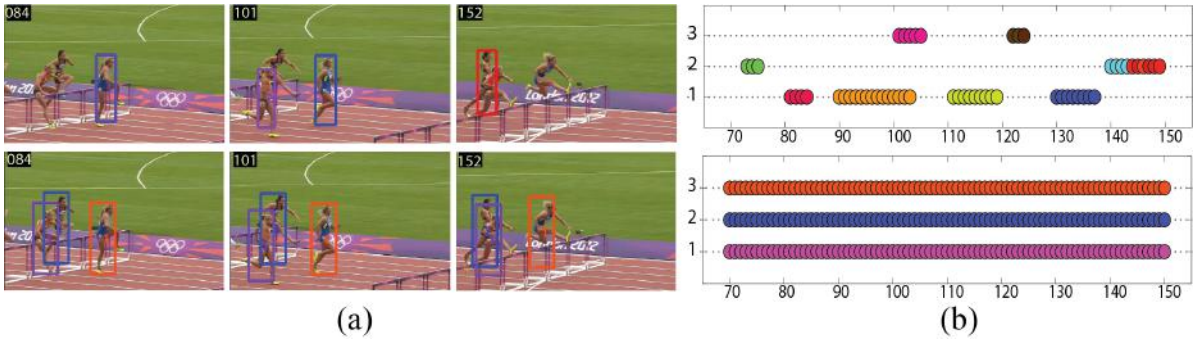

Figure 1: Failure case of data association based trackers. (a) shows the tracking results of our method (bottom row) and the method proposed in [3] (top row). A pre-trained object detector fails when objects go under heavy articulation. This error is propagated to the data association step, which consequently cause failure in tracking. Differently, our method is based on online discriminative learning and solves detection and global data association simultaneously, thus handles articulated targets well. The same observation can be made from (b). Each row represents one of the three identities in the scene. Each circle shows a corresponding match in a frame and the color represents the ID that is assigned to that detection.

in [26] include image appearance cues by solving multiple networks in parallel, each network representing one appearance group.

Although data-association based tracking methods have shown to be promising, still there is a major downside to such approaches. The performance is highly reliant on the performance of object detector. If the object detector fires a lot of false alarms, or misses many true detections, the data association fails consequently. In particular, in case of articulated objects the detector often fails when object goes under heavy articulation. This causes failure in tracking. An example is shown in Fig. 1.

Recent approaches have focused on improving the performance of the generic object detector [27, 20, 28] or designing a better data association techniques [37, 13] to improve tracking. Shu *et al*. in [27] proposed an extension to deformable part-based human detector [15] which can handle occlusion up to a scale. Milan *et al*. [20] use superpixels and low-level image information to address the shortcoming of a generic object detector. An alternative method to overcome the drawbacks of object detector when dealing with articulated objects or arbitrary objects (when there does not exist a good pre-trained detector) is online learning of the classifier for objects [2, 17, 32]. Online discriminative learning approaches allow training target specific classifiers for a given sequence using different features including video specific features like color histogram. Moreover, these classifiers can adapt themselves as the appearance of targets change, which is not the case in pre-trained object detector.

Online discriminative learning methods have been used extensively for tracking deformable objects in the context of *single object tracking*. However, its extension to multiple objects remains relatively unexplored and is limited to only few works. The work of Zhang and Maaten [39] is probably the first attempt to apply online discriminative learning in tracking multiple objects. In [39], the spatial constraint among the targets is modeled during tracking. It is shown that the tracker performs well when the structure among the objects remains the same (or changes very slowly). However, this is only applicable to very limited scenarios and it will perform poorly in others, specially when the targets are moving independently.

In this paper we propose a tracking method based on online discriminative learning, which solves detection and global data association simultaneously by integrating a new global data association technique into the inference of a structured learning tracker. Our learning step is inspired by STRUCK [17], which is the state-of-art based on recent studies [29, 36]. We extend STRUCK to track multiple objects simultaneously. Despite other online trackers which are temporally local, our method provides the tracks across a segment of a video. The input to our tracker in every frame, is densely sampled candidate windows instead of sparse detections. This allows our tracker to infer temporal consistency between the frames and correct poor detections (mostly caused by occlusion or severe pose change), thus avoiding error propagation. We propose to do the inference through a new target identity-aware network flow graph which is a variant of multi-commodity flow graph [18].

The network used in our work is different from those in previous works [26, 33]. First, our network includes the target identities by considering more than one node per candidate location, where each node encodes the probability of assigning one of the target identities to that candidate location. Moreover, the network consists of multiple source and sink nodes, where each pair accounts for entry and exit of

one of the targets. Second, the exact solution to the proposed network flow problem opens the door to using powerful structured learning algorithm and we show how the proposed network can be used in an inner loop of structured learning which has not been explored before. Our structured learning framework allows training target specific model which eliminates the need for noisy pre-trained detectors. Third, we show that a high-quality solution to the network can be found through Lagrange relaxation of some of the hard constraints which is more efficient compared to Integer Programming (IP) or Linear Programming (LP) solutions. After relaxing the constraints, at each iteration, the problem reduces to finding the best track for each target individually, where the optimal solution can be found in linear time through dynamic programming. Thus we do not need to prune the graph as in [26, 33].

Additionally, the proposed iterative solution allows us to easily incorporate a soft spatial constraint that penalizes the score of candidate windows from different tracks that highly overlap during optimization. This helps reducing the ambiguity caused by nearby targets with similar appearance in the crowded scenes. Moreover, our spatial constraint replaces the greedy non-maximum suppression step used in most of the object detectors. Our approach, by bringing detection and data association in a single framework, not only enables us to track arbitrary multiple objects (for which there does not exist a good pre-trained detector) but also helps in better dealing with common challenges in multiple object tracking such as pose changes, miss detections and false alarms mostly caused by using a pre-trained object detector. We not only achieve results better than the state of art on sequence which pre-trained detectors perform well, but also we improve state-of-art by a significant margin on sequences for which generic detectors fail.

In summary, our main contributions are: (1) we present a new multiple-object tracking method which combines discriminative learning and global data association, (2) we introduce a new target identity-aware network and efficiently optimize it through Lagrangian relaxation, (3) we show that the proposed iterative optimization is more efficient compared to IP/LP solutions, (4) our soft-spatial constraint replaces the ad-hoc non-maximum suppression step of object detection methods and further improves the results. Finally, (5) we show that our method can achieve results better than state-of-art on challenging sequences.

2. Proposed Approach

Given the initial bounding boxes for the objects entering the scene in the first few frames (from annotation or using an object detector), our method starts by training a model for each of the objects through structured learning (section 3). During learning, the most violated constraints are found by searching for a set of tracks that minimize the cost function of our target identity-aware network flow. Later, the same network is used to find the best tracks in the next temporal span (segment) of a sequence (section 4). The new tracks are later used to update the model through passive aggressive algorithm [11]. An example is show in Figure 2.

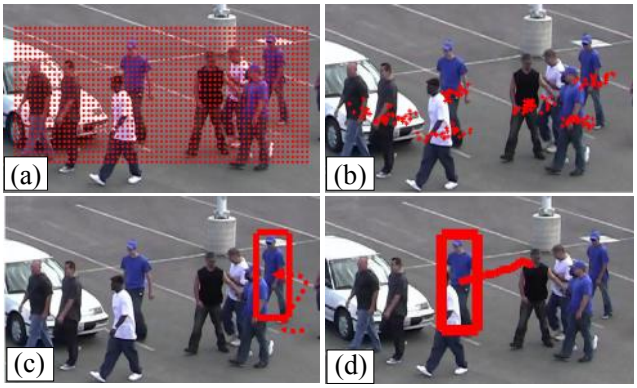

Figure 2: Tracking steps for one person in batch of frames. (a) shows the union of dense candidate windows used in a batch of frames in our method. (b) illustrates the union of human detection results of [15]. (c) shows the most violated constraint found through TINF to update the classifier and in (d) we show the tracking result of our method.

3. Target-specific Model

Given a set of τ training images, $X = \{\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^\tau\} \subset \mathcal{X}$, along with labels $Y = \{\mathbf{y}_1^1, \mathbf{y}_2^1, ..., \mathbf{y}_K^1, ..., \mathbf{y}_{K-1}^\tau, \mathbf{y}_K^\tau\} \subset \mathcal{Y}$, where $\mathbf{y}_k^t$, defines the bounding box location of object k in frame t, the target models are obtained through structured learning [31]. The aim of learning is to find a prediction function $f : \mathcal{X} \mapsto \mathcal{Y}$, which directly predicts the locations of all the objects in a set of frames. The task of structured learning is to learn a prediction function of the form

$$f_w(X) = \arg\max_{Y \in \mathcal{Y}} \sum_{t=1}^{\tau} \sum_{k=1}^{K} \mathbf{w}_k^T \phi(\mathbf{x}^t, \mathbf{y}_k^t), \qquad (1)$$

where $\mathbf{w} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_K\}$ is the concatenation of the models for all the K objects. $\phi(\mathbf{x}^t, \mathbf{y}_k^t)$ is the joint feature map which represents the feature extracted at location $\mathbf{y}_k^t$ in frame t. The optimal parameter vector $\mathbf{w}^*$ is obtained by solving the following optimization problem:

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + C\xi \qquad s.t. \quad \xi \geq 0$$
$$\sum_{t=1}^{\tau} \sum_{k=1}^{K} \mathbf{w}_k^T \left(\phi(\mathbf{x}^t, \mathbf{y}_k^t) - \phi(\mathbf{x}^t, \bar{\mathbf{y}}_k^t)\right) \geq \Delta(Y, \bar{Y}) - \xi$$
$$\forall \bar{Y} \in \mathcal{Y} \setminus Y. \qquad (2)$$

The loss function is defined based on the overlap between groundtruth label Y and prediction $\bar{Y}$

$$\Delta(Y, \bar{Y}) = \frac{1}{\tau} \sum_{t=1}^{\tau} \sum_{k=1}^{K} (1 - (\mathbf{y}_k^t \cap \bar{\mathbf{y}}_k^t)). \quad (3)$$

Due to exponential number of possible combinations of bounding boxes in $\mathcal{Y}$, exhaustive verification of constraint in 2 is not feasible. However [31, 21] showed that high quality solution can be obtained in polynomial time by using only the *most-violated constraints*, i.e a set of bounding boxes that maximize the sum of scores and loss functions. Once the model parameters are learned ($\mathbf{w}$), we use the same inference that we used for finding the *most-violated constraints* to find the best set of tracks for all the K objects in next segment of the video.

4. Track Inference

Given the model parameters, $\mathbf{w}$, and *dense overlapping bounding boxes* in each frame, the goal is to find a sequence of candidate windows, called a track, for each object which maximizes the score in Eq. 1. This maximization requires searching over exponentially many configurations. We propose to formulate the inference as a global data association which helps reducing the search space by enforcing some temporal consistency across the candidates in consecutive frames. Recently, such global data association has been formulated using network flow [38, 22], for which there exists an exact solution. In order to be able to use such networks as inference of our structured learning, the solution to the network needs to maximize the score function in Eq. 1. This requires the nodes in the graph to encode the probability of assigning each of the target identities to them using the learned parameters $\mathbf{w}_k$. This is not possible through traditional network flow methods.

We propose a new network called Identity-Aware network, which is shown in Fig. 3. The black circles represent all possible candidate locations in each frame (densely sampled across the entire frame). Each candidate location is represented with a pair of nodes that are linked through K *observation edges*; one *observation edge* for each identity. This is different from traditional network flow for which there is only one *observation edge* connecting a pair of nodes. Another major difference between our network with traditional network flow is that, our network has K sources and K sinks, each belonging to one object. The rest of the network is similar to that of traditional network flow. Transition edges that connect nodes from different frames, represent a potential move of an object from one location to the other and there is a transition cost associated with that. There is an edge between the start/sink node and every other node in the graph which takes care of persons entering/leaving the scene. (For simplicity we are only showing some of the entry/exit edges).

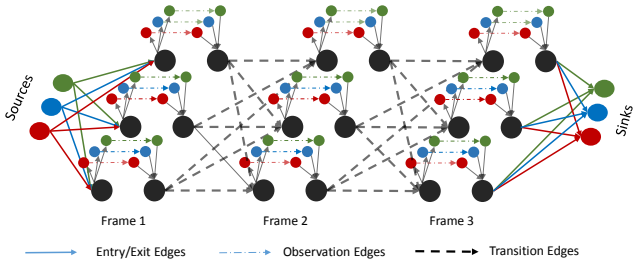

Figure 3: Shows the network used in our inference for three identities. Each identity is shown with a unique color. The flow entering each node can take only one of the three observation edges depending on which source (identity) does it belong to. The constraint in Eq. 8 ensures that one candidate can belong to only one track, so the tracks will not overlap.

The flow is a binary indicator which is 1 when a node is part of a track and 0 otherwise. A unit of flow is pushed through each source and the tracks for all the objects are found by minimizing the cost assigned to the flows. In addition, we will show later that by setting the upper bound of flows passing through *observation edges* of one bounding box, we will ensure that at most one track will claim one candidate location. In the following subsections we will first present formulation of the problem as a Lagrangian relaxation optimization and later we will introduce our spatial constraint which replace the greedy non-maximum suppression in object detectors.

4.1. Target Identity-aware Network Flow

First we need to build our graph $G(V, E)$. For every candidate window in frame t we consider a pair of nodes which are linked through K different *observation edges*, each belonging to one identity. For every node v_p, in frame t and v_q in frame $t+1$, there has to be a transition edge between the two if v_q belongs to the neighborhood of v_p. Neighborhood of the node v_p is defined as

$$v_q^{t+1} \in N_\sigma(v_p^t) \Leftrightarrow \left\| v_p^t - v_q^{t+1} \right\|_2 \leq \sigma,$$

we consider a neighboring area within σ distance of node v_p that connects two candidate windows in two consecutive frames. In addition, we have source/sink edges which connect all the candidate windows to the source and sink nodes.

Different edges in our graph are assigned costs that take into account different characteristics of objects during tracking. Each pair of nodes which represents a candidate window will be assigned K different costs defined by the K target-specific models. Considering $\mathbf{w}_k$ to be the linear weights learned for the k^{th} object, the cost assigned to k^{th}

observation edge representing the candidate location $\mathbf{y}_p^t$ in frame t is computed as follow:

$$c_{ij}^k = -\mathbf{w}_k^T \phi(\mathbf{x}^t, \mathbf{y}_p^t).$$

Transition edges which connect the nodes in consecutive frames are assigned costs which incorporate both appearance and motion direction. The cost of a transition edge (c_{ij}^k) which connects two candidate windows $\mathbf{y}_p^t$ and $\mathbf{y}_q^{t+1}$ in two consecutive frames is computed as:

$$c_{ij}^k = -\alpha K(\phi_c(\mathbf{x}^t, \mathbf{y}_p^t), \phi_c(\mathbf{x}^{t+1}, \mathbf{y}_q^{t+1})) - \beta \frac{V_{pq} V_{ref}^k}{\|V_{pq}\| \left\|V_{ref}^k\right\|}, \tag{4}$$

where $K(\phi_c(\mathbf{x}^t, \mathbf{y}_p^t), \phi_c(\mathbf{x}^{t+1}, \mathbf{y}_q^{t+1}))$ is the histogram intersection between the color histograms extracted from the location $\mathbf{y}_p^t$ and $\mathbf{y}_q^{t+1}$. $\frac{V_{pq} V_{ref}^k}{\|V_{pq}\| \left\|V_{ref}^k\right\|}$ is the cosine similarity between the reference velocity vector V_{ref}^k for the k^{th} object[1] and the velocity vector between the two candidate windows V_{pq}.

Once the graph $G(V, E)$ is constructed, our aim is to find a set of K flows (tracks) by pushing a unit of flow through each source node. The flow $f_{i,j}^k$, is found by minimizing the following cost function:

$$C(f) = \sum_{k=1}^{K} \sum_{(i,j)\in E} c_{ij}^k f_{ij}^k. \tag{5}$$

The flow passing through these edges need to satisfy some constraints to ensure that it can actually represent a track in a real world. The set of constraints that we define in our graph are as follow:

$$\sum_j f_{ij}^k - \sum_j f_{ji}^k = \begin{cases} 1 & \text{if } i = s_k \\ -1 & \text{if } i = t_k \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

$$f_{ij}^k \in \{0, 1\} \qquad \forall (i,j) \in E \ \ and \ \ 1 \le k \le K \tag{7}$$

$$\sum_{k=1}^{K} f_{ij}^k \le 1 \tag{8}$$

The constraint in Eq. 6 is the supply/demand constraint, enforcing the sum of flows arriving at one node to be equal to the sum of flows leaving that node. Constraint in Eq. 8 is the bundle constraint, ensuring that the tracks of different identities will not share a node by setting the upper bound of sum of flows passing through each node to be one.

One can formulate Eq. 5 as an Integer Program (IP). Since IP is NP-Complete, in practice, the problem can be relaxed to Linear Program (LP) in which the solution can be found in polynomial time. However, our experiments show that without pruning steps like the one in [26, 33], which reduces the number of candidate windows, it is intractable to find a solution for a large number of people in a long temporal span (one should note that the input to our tracker is dense candidate windows sampled from the entire frame). Instead, we propose a Lagrange relaxation solution to this problem. We show that after relaxing the hard constraints, the problem in each iteration, reduces to finding the best track for each target separately. The global solution to this can be found in linear time through dynamic programming. Moreover, our iterative optimization allows us to incorporate spatial constraint which further improves the tracking results.

4.2. Lagrange Relaxation Solution to TINF

The key idea of Lagrange relaxation is relaxing the hard constraints and moving them into the objective function in order to generate a simpler approximation. We start by relaxing the bundle constraints in Equation. 8, where we introduce the non-negative Lagrange multiplier λ_{ij}. λ is a vector of Lagrange multipliers that has the same dimension as the number of edges in the graph. After relaxing the bundle constraint the new objective function becomes:

$$C(f) = \sum_{k=1}^{K} \sum_{(i,j)\in E} c_{ij}^k f_{ij}^k + \sum_{(i,j)\in E} \lambda_{ij} (\sum_{k=1}^{K} f_{ij}^k - 1), \tag{9}$$

We can further simplify this and write it as follow:

$$C(f) = \sum_{k=1}^{K} \sum_{(i,j)\in E} (c_{ij}^k + \lambda_{ij}) f_{ij}^k - \sum_{(i,j)\in E} \lambda_{ij}, \tag{10}$$

Subject to:

$$\sum_j f_{ij}^k - \sum_j f_{ji}^k = \begin{cases} 1 & \text{if } i = s_k \\ -1 & \text{if } i = t_k \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$f_{ij}^k \in \{0, 1\} \qquad \forall (i,j) \in E \ \ and \ \ k \in K \tag{12}$$

The second term in Eq. 10 is a constant for any given choice of Lagrange multipliers, therefore we can ignore it. The new objective function has a cost of $c_{ij}^k + \lambda_{ij}$ associated with every flow variable f_{ij}^k. Since none of the constraints in this problem contains the flow variables for more than one of the identities, we can decompose the problem into separate **minimum cost flow** problem for each identity. Since only one unit of flow is pushed through each source, the solution to minimum cost flow can be found optimally through dynamic programming in $O(N)$. Thus the

[1] Average velocity vector for the k^{th} identity in previous batch

Figure 4: In top row the tracks of two pedestrians get confused due to their appearance similarity. This issue is fixed when the spatial constraint is enforced (bottom row).

complexity of our optimization in each iteration is $O(KN)$, where K is the number of targets and N is the number of frames in the temporal span. Consequently, to apply the sub-gradient optimization to this problem, we alternate between the following two steps:

- For a fixed value of Lagrange multipliers we would solve the minimum cost flow for each identity separately considering the cost coefficients $c_{ij}^k + \lambda_{ij}$.
- Update the Lagrange multipliers according to Eq. 13.

$$\lambda_{ij}^{q+1} = \left[\lambda_{ij}^q + \theta^q(\sum_{k=1}^{K} f_{ij}^k - 1)\right]^+, \tag{13}$$

where λ^q is the Lagrange multipliers at iteration q, θ^q is the step size defining how far we would like to move from current solution and $[\alpha]^+ = max(0, \alpha)$.

4.3. Spatial Constraint

One major difference between our tracking algorithm and other data association based trackers is that, the input to our tracker is dense candidate windows instead of human detection output. When pedestrians with similar appearance and motion are walking next to each other, it is very likely to have ID-Switches in tracking results. Also when a pedestrian becomes partially occluded, the track for that person tend to pick candidates that highly overlap with other nearby pedestrians (see Fig. 4). This issue is addressed by non-maximum suppression in human detection [15] or by using other techniques like the one in [39], where the objects are forced to keep the spatial configurations between consecutive frames. Instead we introduce a soft-spatial constraint which penalizes the tracks that highly overlap. Our spatial constraint can be easily integrated into our iterative optimization. Similar to our Lagrange multipliers, we introduce a new set of variables that penalizes the cost of *observation edges* that highly overlap. Now the cost associated to each *observation edge* becomes $c_{ij}^k + \lambda_{ij} + \rho_{ij}$. ρ is a vector which has the same size as the number of *observation edges* in the graph. It is initialized with a zero vector in the first iteration and is updated according to Eq. 14.

$$\rho_{ij}^{q+1} = \left[\rho_{ij}^q + \theta^q[(y_i^t \cap y_j^t) - 0.5]^+ \exp^{((y_i^t \cap y_j^t)-0.5)/2}\right]^+, \tag{14}$$

where $y_i^t \cap y_j^t$ is the overlap between neighboring bounding boxes in the same frame. ρ_{ij} penalizes the observation node which is associated with the cost c_{ij}. One should note that the spatial constraint only penalizes the bounding boxes that overlap more than 50% and the penalty increases exponentially as the overlap increases. After adding the spatial constraint the cost of the nodes are updated at each iteration according to the following:

$$c_{ij}^{q+1,k} = c_{ij}^k + \lambda_{ij}^{q+1} + \rho_{ij}^{q+1}. \tag{15}$$

We observed that penalizing both nodes that highly overlap, sometimes lead to inaccurate bounding boxes for one of the tracks. Therefor, we only penalize the observation nodes of the track that have lower score according to the score function in Eq. 1. The algorithm of our Lagrangian relaxation solution, including the spatial constraint, is shown in Algorithm 1.

Algorithm 1: Lagrangian Relaxation Solution to TINF.

Input: candidate windows in T frames
model parameters for each identity ($\mathbf{w}_k$)
Output: Tracking result for K identities

- build the TINF graph
$G(V, E)$
- Initialize the lagrange multipliers and spatial constraint multipliers
$\lambda = 0,\ \rho = 0,\ \theta = 1,\ q = 1$
while *do not converge* **do**
-Solve the minimum cost flow for each identity ($\mathbf{f}^k$)
-Update Lagrange multiplies;
$\lambda_{ij}^{q+1} = \left[\lambda_{ij}^q + \theta^q(\sum_k^K f_{ij}^k - 1)\right]^+$
-Update spatial constraint multipliers;
$\rho_{ij}^{q+1} =$
$\left[\rho_{ij}^q + \theta^q[(y_i^t \cap y_j^t) - 0.5]^+ \exp^{((y_i^t \cap y_j^t)-0.5)/2}\right]^+$
-Update edge costs
$c_{ij}^{q+1,k} = c_{ij}^k + \lambda_{ij}^{q+1} + \rho_{ij}^{q+1}$
-Update step size
$\theta^{q+1} = \frac{1}{q}$
$q = q + 1$
end

5. Experimental Results

In our evaluation, we focus on tracking humans, due to its importance. But our method can be used for tracking

any object. We conducted two sets of experiments. First we compare our method with the state of the art trackers on publicly available sequences. For those sequences where the object detection performs well, excellent results are already reported. However, we show that, using our method, one can further improve the performance. Second, we evaluated our method on two new sequences where targets experience heavy articulation and we show that we can significantly improve the performance of data-association based trackers as well as online trackers. *Parking Lot 1* [27], *Parking Lot 2* [28], *TUD Crossing* [3] and PET [16] are the four publicly available sequences used in our experiments and the two new sequences are called *Running* and *Dancing*.

Setup. To initialize the target, similar to [39, 10] we used manual annotation. We annotated four initial bounding boxes for each object entering the scene. We also report results where targets are initialized automatically using a pre-trained object detector. For manual annotation the target is initialized only once and there is no re-initialization of targets. We use histogram-of-oriented gradient [12] and color histogram [14] as our features. We found the combination of both features to be important. HOG captures the edge information of target and is helpful in detecting target from the background, while color histogram is a video specific features and helps in distinguishing different targets from each other. The sequence is divided into segments of 20 frames each. At the end of each temporal span we check if a track is valid or not by comparing its score with a predefined threshold. If the track is valid then it is used to update the model.

Comparison. We quantitatively and qualitatively compare our method with two main sets of trackers: data-association based trackers and online trackers. On sequences for which no other tracking results are reported, we compare our method with three data-association based trackers for which we have access to their code, CET [3], DCT [4] and GOG [22]. We used Deformable Part based model [15] as our human detector. The input to the data-association methods is the DPM output with different thresholds ranging from -1 to 0. We agree that these trackers have parameters to tune to achieve the best performance for each sequence. However, we stayed with the default parameter suggested by the authors and the only parameter we changed was the human detector threshold. The numbers reported are for a threshold that gave us the best performance. In addition to these three trackers, we quantitatively compared our results with other trackers which have used the same sequences in their experiments. For online discriminative learning-based trackers we selected STRUCK [17] as well as structure preserve multi-object tracking (SPOT) approach [39]. For STRUCK, we train one

	Method	MOTA	MOTP	MT	ML	IDS
Running	CET	0.463	0.508	0.67	0	0
	DCT	0.376	0.504	0	0	0
	GOG	0.03	0.6945	0	1	0
	SPOT	0.661	0.662	0.67	0	0
	STRUCK	0.799	0.643	1	0	0
	Ours	**0.987**	**0.665**	**1**	**0**	**0**
Dancing	CET	0.366	0.62	0.57	0	64
	DCT	0.363	0.636	0	0.14	81
	GOG	0.249	0.64	0	0.14	96
	SPOT	0.554	0.659	0.43	0	16
	STRUCK	0.691	0.671	0.71	0.14	9
	Ours	**0.899**	**0.659**	**0.86**	**0**	**1**
Parking Lot 2	CET	0.717	0.558	0.6	0	59
	DCT	0.736	0.565	0.8	0	48
	GOG	0.4827	0.598	0.2	0.1	96
	Ours	**0.893**	**0.663**	**1**	**0**	**0**
Parking Lot 1	LPD	0.893	0.777	NR	NR	NR
	GMCP	0.9043	0.741	NR	NR	NR
	H2T	0.884	0.819	0.78	0	21
	Ours	**0.907**	**0.693**	**0.86**	**0**	**3**
TUD Crossing	PF	0.843	0.71	NR	NR	2
	GMCP	0.9163	0.756	NR	NR	0
	Ours	**0.929**	**0.692**	**1**	**0**	**0**
PET 2009	LDA	0.9	0.75	0.89	NR	6
	DLP	0.91	0.7	NR	NR	5
	GMCP	0.903	0.6902	NR	NR	8
	Ours	**0.904**	**0.6312**	**0.95**	**0**	**3**

Table 1: Quantitative comparison of our method with competitive approaches of LPD [30], LDA [24], DLP [19], H2T [34], GMCP [37], PF [8],CET [3], DCT [4], GOG [22], STRUCK [17] and SPOT [39].

structured SVM per target given the annotation of humans in the first frame. For SPOT, the manual annotation is used to initialize the tracking. Whenever a new object enters the scene we re-initialize the tree to get the track for the new target. In SPOT the spatial relationship between the targets are modeled during tracking. This model is updated according to a weight ,γ, every frame. The weight was set originally to 0.05 and we found the weight to be important in final results. The reported results for SPOT are based on the best value that we found for γ.

For quantitative analysis we utilized two sets of metrics. CLEAR MOT metrics [7] as well as Trajectory Based Metrics (TBM) [38]. CLEAR metrics (MOTA-MOTP) look at the entire video as a whole while TBM consider the behavior of each track separately. Each of these metrics captures different characteristics of a tracker and it is important to look at both of them while comparing different tracking algorithms to better capture strength and weakness of each tracker. [2]

Initialization. For initialization, besides manual anno-

[2]For more information please visit: http://crcv.ucf.edu/projects/TINF/

	MOTA	MOTP	MT	ML	IDS
Running	0.972	0.681	1	0	0
Running-SP	0.987	0.665	1	0	0
Dancing	0.88	0.649	0.86	0	2
Dancing-SP	0.899	0.659	0.86	0	1
PL1	0.88	0.629	0.79	0	4
PL1-SP	0.907	0.693	0.86	0	3
PL2	0.822	0.656	0.9	0	2
PL2-SP	0.893	0.663	1	0	0
TUD	0.866	0.698	0.92	0	1
TUD-SP	0.929	0.692	1	0	0

Table 2: This table shows the performance of our method with and without spatial constraint. The improvement from spatial constraint is evident from this evaluation.

Method	MOTA	MOTP	MT	ML	IDS
PL1-Auto	0.905	0.652	0.857	0	5
PL1-Manual	0.907	0.693	0.8571	0	3
TUD-Auto	0.908	0.688	0.9167	0.083	0
TUD-Manual	0.929	0.692	1	0	0
PL2-Auto	0.834	0.632	0.7	0	5
PL2-Manual	0.893	0.663	1	0	0

Table 3: This table shows the performance of our method with automatic and manual initialization of the targets. For automatic initialization of targets a pre-trained human detector is used [15].

tation, we use human detection to automatically initialize the targets. During each segment a new track is initialized if there are at least four confident detections in consecutive frames that highly overlap and are not associated to any other tracks. We tested automatic initialization of targets on publicly available sequences where human detection performs reasonably well. As can be seen in Table. 3, the performance of our method doesn't change much when using automatic initialization. The main difference is that some of the tracks in some sequences will start late compared to manual annotation which cause a small drop in MOTA due to the added false negatives.

Effect of Spatial Constraint. In order to clearly see the effect of our spatial constraint, we ran our method on different sequences with and without the spatial constraint. As can be seen in Table. 2, when spatial constraint is added, the performance increases, specially for sequences which involve interaction between objects.

Run Time and Convergence. In order to compare the complexity of the proposed Lagrangian relaxation method with the one of IP and LP, we implemented the IP and LP version of our method as well. We used CPLEX [1] as the optimization toolbox.The performance of IP and LP is within $1-2\%$ performance of our Lagrange relaxation formulation when no spatial constraint is used. The runtime for a selected segment of PL2 sequence with different number of targets is shown in the top row in Fig. 5. Note that the curves are shown with logarithmic coordinates. As can be observed, the proposed optimization is a lot more efficient compared to the IP and LP solutions. Finally, the bottom row in Fig. 5 shows the number of iterations that the Lagrangian optimization takes to converge in PL2 sequence. In Fig. 5 the horizental axes shows the segment number in PL2 sequence.

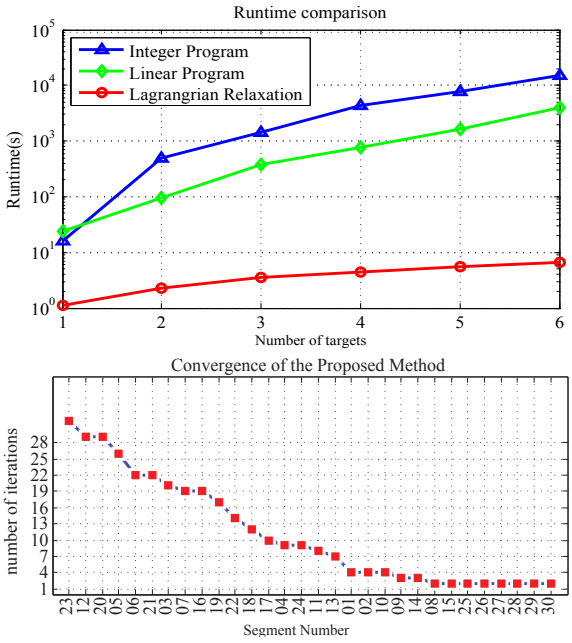

Figure 5: The top figure shows the run time comparison of the proposed Lagrangian solution vs IP and LP. The bottom figure shows the convergence of the proposed method on PL2 sequence.

6. Conclusion

In this paper we introduce a new tracker which brings in discriminative learning and global data association method in a unified framework. At the core of our framework lies a structured learning which learns a model for each target. The inference is formulated as global data association problem which is solved through a proposed target identity-aware network flow. Our experiments show that the proposed method outperforms traditional online trackers in difficult scenarios. Our work is one of the very few attempts that aims to solve tracking multiple objects by solving detection and tracking simultaneously. We hope that our results encourage other researcher to discover this direction more.

References

[1] Ibm ilog cplex optimizer, www.ibm.com/software/integration/optimization/cplex-optimizer. 8

[2] Z. K. andKrystian Mikolajczyk and J. Matas. Tracking-Learning-Detection. In *PAMI*, 2010. 2

[3] A. Andriyenko and K. Schindler. Multi-target Tracking by Continuous Energy Minimization. In *CVPR*, 2011. 2, 7

[4] A. Andriyenko, K. Schindler, and S. Roth. Discrete-Continuous Optimization for Multi-Target Tracking. In *CVPR*, 2012. 7

[5] Y. Bar-Shalom, T. Fortmann, and M. Scheffe. Joint probabilistic data association for multiple targets in clutter. In *Information Sciences and Systems*, 1980. 1

[6] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua. Multiple Object Tracking Using K-Shortest Paths Optimization. In *PAMI*, 2011. 1

[7] K. Bernardin and R. Stiefelhagen. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 7

[8] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Robust tracking-by-detection using a detector confidence particle Filter. In *ICCV*, 2009. 7

[9] A. Butt and R. Collins. Multi-target Tracking by Lagrangian Relaxation to Min-Cost Network Flow. In *ICCV*, 2013. 1

[10] S. Chen, A. Fern, and S. Todorovic. Online multi-person tracking-by-detection from a single, uncalibrated camera. In *CVPR*, 2014. 7

[11] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, , and Y. Singer. Online passive-aggressive algorithms. In *Journal of Machine Learning Research*, 2006. 3

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 7

[13] A. Dehghan, S. Modiri, and M. Shah. GMMCP-Tracker:Globally Optimal Generalized Maximum Multi Clique Problem for Multiple Object Tracking. In *CVPR*, 2015. 1, 2

[14] J. Domke and Y. Aloimonos. Deformation and viewpoint invariant color histograms. In *BMVC*, 2006. 7

[15] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. In *PAMI*, 2010. 2, 3, 6, 7, 8

[16] J. Ferryman and A. Shahrokni. Dataset and challenge. *International Workshop on Performance Evaluation of Tracking and Surveillance*, 2009. 7

[17] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. In *ICCV*, 2011. 2, 7

[18] G. Karakostas. Faster approximation schemes for fractional multicommodity flow problems. In *ACM-SIAM*, 2002. 2

[19] A. K. K.C. and C. D. Vleeschouwer. Discriminative Label Propagation for Multi-Object Tracking with Sporadic Appearance Features. In *ICCV*, 2013. 7

[20] A. Milan, L. Leal-Taix, K. Schindler, and I. Reid. Joint tracking and segmentation of multiple targets. In *CVPR*, 2015. 2

[21] C. nam Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, 2009. 4

[22] H. Pirsiavash, D. Ramanan, and C. Fowlkes. Globally-Optimal Greedy Algorithms for Tracking a Variable Number of Objects. In *CVPR*, 2011. 1, 4, 7

[23] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automated Control*, 1996. 1

[24] A. V. Segal and I. Reid. Latent Data Association: Bayesian Model Selection for Multi-target Tracking. In *ICCV*, 2013. 7

[25] K. Shafique and M. Shah. A noniterative greedy algorithm formultiframe point correspondence. In *PAMI*, 2005. 1

[26] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Multi-commodity network flow for tracking multiple people. In *PAMI*, 2013. 2, 3, 5

[27] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah. Part-based Multiple-Person Tracking with Partial Occlusion Handling. In *CVPR*, 2012. 1, 2, 7

[28] G. Shu, A. Dehghan, and M. Shah. Improving an Object Detector and Extracting Regions using Superpixels. In *CVPR*, 2013. 2, 7

[29] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual Tracking: an Experimental Survey. In *PAMI*, 2013. 2

[30] S. Tang, M. Andriluka, A. Milan, K. Schindler, S. Roth, and B. Schiele. Learning people detectors for tracking in crowded scenes. In *ICCV*, 2013. 7

[31] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. In *JMLR*, 2005. 3, 4

[32] S. Wang, H. Lu, F. Yang, and M. Yang. Superpixel tracking. In *CVPR*, 2011. 2

[33] X. Wang, E. Turetken, F. Fleuret, and P. Fua. Tracking interacting objects optimally using integer programming. In *ECCV*, 2014. 2, 3, 5

[34] L. Wen, W. Li, J. Yan, Z. Lei, D. Yi, and S. Z. Li. Multiple target tracking based on undirected hierarchical relation hypergraph. In *CVPR*, 2014. 7

[35] B. Wu and R. Nevatia. Detection and Tracking of Multiple, Partially Occluded Humans by Bayesian Combination of Edgelet based Part Detectors. volume 75, pages 247–266, 2007. 1

[36] Y. Wu, J. Lim, and M. H. Yang. Online Object Tracking: A Benchmark. In *CVPR*, 2013. 2

[37] A. R. Zamir, A. Dehghan, and M. Shah. GMCP-Tracker: Global Multi-object Tracking Using Generalized Minimum Clique Graphs. In *ECCV*, 2012. 1, 2, 7

[38] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008. 1, 4, 7

[39] L. Zhang and L. van der Maaten. Structure Preserving Object Tracking. In *CVPR*, 2013. 2, 6, 7