Tracking The Untrackable: Learning To Track Multiple Cues with Long-Term Dependencies

Amir Sadeghian, Alexandre Alahi, Silvio Savarese
Stanford University
{amirabs,alahi,ssilvio}@cs.stanford.edu

Abstract

We present a multi-cue metric learning framework to tackle the popular yet unsolved Multi-Object Tracking (MOT) problem. One of the key challenges of tracking methods is to effectively compute a similarity score that models multiple cues from the past such as object appearance, motion, or even interactions. This is particularly challenging when objects get occluded or share similar appearance properties with surrounding objects.

To address this challenge, we cast the problem as a metric learning task that jointly reasons on multiple cues across time. Our framework learns to encode long-term temporal dependencies across multiple cues with a hierarchical Recurrent Neural Network. We demonstrate the strength of our approach by tracking multiple objects using their appearance, motion, and interactions. Our method outperforms previous works by a large margin on multiple publicly available datasets including the challenging MOT benchmark.

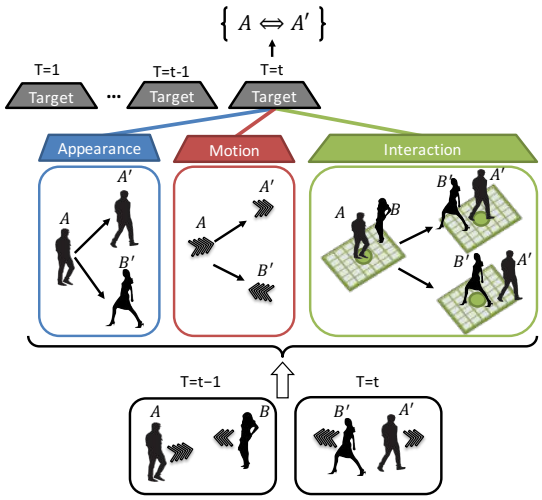

Figure 1. We present a learning framework based on a hierarchical Recurrent Neural Network that jointly encodes multiple cues across time. Our learned representation is used to compute the similarity scores of a "Tracking-by-detection" algorithm. [15]

1. Introduction

Representation learning frameworks are fast becoming an essential instrument in solving perception tasks and have been shown to approach human-level accuracy in classifying images [18, 17]. However, the status quo of the Multi-Object Tracking (MOT) problem is still far from matching human performance [57, 69]. This is mainly due to the difficulty of reasoning with multiple cues across time such as object appearance, motion, and interactions. In this work, we tackle such challenge by jointly learning a metric that takes into account multiple cues in time and space in an end-to-end fashion (see Figure 1).

The objective of the MOT task is to infer trajectories of objects as they move around. It covers a wide range of applications such as sports analysis [40, 47, 70], biology (birds [41], ants [27], fishes [62, 63, 14], cells [42, 36]), robot navigation [11, 12], and autonomous driving vehicles [13, 53]. We follow the "Tracking-by-detection" paradigm whereby detection outputs are to be connected across video frames with a similarity metric. This is often formulated as an optimization problem with respect to a graph [51, 52]. Each detection is represented by a node and edges are weighted with a similarity score.

Over the past decades, researchers have made significant progress in proposing techniques to solve the optimal assignments over graphs efficiently [80, 1, 32, 60] . However, their MOT performances are limited by the similarity scores. Recently, Xiang et al. [69] have shown that learning a representation would improve the similarity score of a tracking framework and outperform previous works by a large margin. Inspired by their result, we argue that the bottleneck of existing tracking methods is the used representations with their corresponding similarity scores. Ideally, with a perfect similarity score, we should not need an optimization technique since greedy matching will be sufficient.

The reality, unfortunately, is different. In crowded envi-

ronments, occlusions, noisy detections –false alarms, missing detections, non-accurate bounding– and appearance variability (same person, different appearance or different people, same appearance) are very common. In traditional MOT approaches [32, 60, 69], the similarity score is handcrafted in the attempt to capture appearance across different time scales, positions, velocities, and the locations of other targets that affect the behavior of the walking target [71]. Existing similarity functions only reason on adjacent temporal frames (pairwise similarity) and combine multiple cues linearly.

We propose to learn a metric that encodes long-term temporal dependencies across multiple cues, i.e., appearance, motion, and interaction without the need to hand specify parameters or weights. Our overall framework is a hierarchical structure of RNNs that has also shown benefits in other applications [25]. In Section 3, we present more details on the inputs of each RNN and how we learn a metric that can be used to compute a similarity score in an end-to-end fashion. Our appearance model is a Siamese Convolutional Neural Network (CNN) that is able to classify a pair of detections as the same or not. Our motion and interaction models leverage two separate Long Short-Term Memory (LSTM) networks that track the motion and interactions of objects for longer tracked period (suited in the presence of long-term occlusions). We then integrate these networks into an end-to-end framework to reason jointly on different learned representations across time. Our framework is generic and scalable to any number of cues and runs online - without the need to see future frames.

In Section 4.1, we present a detailed evaluation of our framework on multiple challenges such as the MOT challenge [34], and Stanford drone dataset [55]. Our method outperforms previous works by a large margin.

2. Related Work

In recent years, tracking has been successfully extended to scenarios with multiple objects [48, 35, 23, 69]. Different from single object tracking approaches which have been constructing a sophisticated appearance model to track single object in different frames, multiple object tracking does not mainly focus on appearance model. Although appearance is an important cue, relying only on appearance can be problematic in MOT scenarios where the scene is highly crowded. To this end, many works have been improving only the appearance model [16, 6], while some have been combining the dynamics and interaction between targets with the target appearance [55, 3, 51, 71, 9, 58, 52].

2.1. Appearance model

An appearance model is closely related to visual representation features of objects. Appearance models vary depending on how precise and rich the visual features they use. Because of efficiency, simple appearance models are widely used in MOT. Many models are based on raw pixel template representation for simplicity [71, 4, 68, 51, 50], while color histogram is the most popular representation for appearance modeling in MOT approaches [9, 35]. Other approaches are using covariance matrix representation, pixel comparison representation, SIFT-like features, or pose features [24, 77, 28, 21, 46]. Recently, deep visual representation of objects have been used for modeling appearance [20, 33, 78]. These high-level features are extracted by deep neural networks mostly convolutional neural networks trained for a specific task. The appearance module of our model shares some characteristics with [20], but differs in two crucial ways: first, we are learning to handle occlusion and solve the re-identification task by a finding a similarity metric between two targets, the network outputs a similarity score indicating how similar two targets are. Second, the overall network architecture is different, particularly, we fuse two small 16 layer VGG [7] networks for faster training and testing with same accuracy. Moreover, we use a different loss function which we will describe in section 3.2.

2.2. Motion model

Object motion model describes how an object moves. Motion cue is crucial for multiple object tracking since knowing the likely position of objects in the future frames will reduce the search space and hence increases the appearance model accuracy. Popular motion models used in multiple object tracking are divided into linear motion models and non-linear motion models. As the name "linear motion" indicates, objects following the linear motion model move with constant velocity across frames. This simple motion model is one of the most popular models in MOT [6, 43, 59, 76, 49]. There are many cases that linear motion models can not deal with long-term occlusions; in this cases, non-linear motion models are proposed to produce a more accurate prediction [72, 73, 10]. We present a new Long Short-Term Memory (LSTM) model which jointly reasons based on the past movements of an object and predicts the future trajectory of that object while it gives accuracy score to the next coordinate of the object based on its previous trajectory. Due to the complexity of human motion patterns, and frequent long period occlusions in busy scenes our model outperforms the simple non-linear motion models, especially in high frame rate scenarios.

2.3. Interaction model

Most of the tracking techniques assume that each target has an independent motion model. This simplification can be problematic in crowded scenes. Interaction models capture the interactions and forces between different objects in a scene [19, 22]. There are two typical interaction models

used in literature known as social force models introduced by [19] and the crowd motion pattern models [22]. Social force models are also known as group models. In these models, each object reacts to energy potentials caused by the interactions with other agents through forces (repulsion or attraction), while trying to keep a desired speed and motion direction [55, 3, 51, 71, 9, 58, 52]. Crowd motion pattern models are another type of interaction models used in MOT, inspired by the crowd simulation literature [79, 61]. In general, this kind of models is usually used for over-crowded scenes [84, 44, 30, 31, 56, 54]. The main drawback of most these methods is that they are limited to a few hand-designed force terms, such as collision avoidance or group attraction. Recently, Alahi et al. [3] proposed to use Long Short-Term Memory networks to jointly reason across multiple individuals (referred to as social LSTM). They presented an architecture to forecast the long-term trajectories of all targets. We use a similar LSTM based architecture. However, our data-driven interaction model is trained to solve the re-identification task as opposed to the long-term prediction.

Finally, when reasoning with multiple cues, previous works combine multiple cues in a hand-crafted fashion without adequately modeling long-term dependencies. Whereas, we propose a hierarchical structure of RNNs to cope with such limitations of previous works. We propose to learn a metric that encodes long-term temporal dependencies across multiple cues, i.e., appearance, motion, and interaction automatically in a data-driven fashion.

3. Multi Object Tracking Framework

The task of Multi-Object Tracking (MOT) consists of detecting multiple objects at each time frame and matching their identities in different frames yielding to a set of object trajectories in time. We address the "Tracking-by-detection" problem. As the input, we use the detection results of an object detector. As the output, our method matches their identities in different frames.

We use the Markov Decision Process (MDP) method [69] for our "Tracking-by-detection" framework. Given a new input frame, MDP computes the similarity scores between the already tracked targets and the newly detected objects. The similarity scores $W_{i,j}$ are used to connect the detections d_j and targets t_i in a bipartite graph. Then, the Hungarian algorithm [45] is used to find the optimal assignments. Note that, in our experiments, a greedy assignment performs as good as the Hungarian method. However, we keep the Hungarian method since it is optimal. In this work, we propose a new metric to compute the similarity scores $W_{i,j}$. Our metric is learned from the data, and it is discussed in section 4.

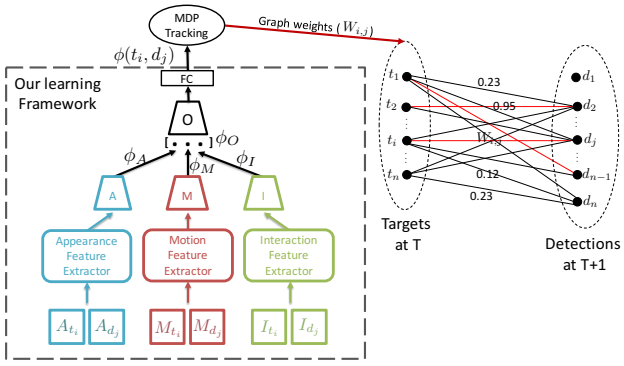

Figure 2. We use a hierarchy of RNNs (the dashed rectangle) to compute the similarity score between targets t_i and detections d_j in the MDP framework [69] when constructing the bipartite graph between the targets and detections. All cues – Appearance (A), Motion (M), and Interaction (I) – have their own RNN. The features represented by these RNNs (ϕ_A, ϕ_M, ϕ_I) are combined through another RNN (referred to it as the target (O) RNN). The squares represent the observed inputs (e.g., image for A_{t_i} or velocities for M_{t_i}).

3.1. Learning Framework

We want to learn a metric that combines multiple cues across time in a principled way. We have identified appearance cues, motion priors, and interactive forces as critical cues of the MOT problem. Recall that, all these cues are not expected to be combined in a linear function towards considering the similarity score. As a result, we address such challenge by using a hierarchical Recurrent Neural Network (RNN) architecture to model all the dependencies.

In our framework, we represent each cue with an RNN. We refer to the RNNs obtained from these cues as appearance (A), motion (M), and interaction (I) RNNs (as illustrated in Figure 2). The features represented by these RNNs (ϕ_A,ϕ_M,ϕ_I) are combined through another RNN which is referred to as target (O) RNN. The target RNN outputs a similarity score from a feature vector, $\phi(t, d)$, which encodes the similarity between a target t and a detection d for data association in an MDP tracking framework.

By using RNNs, more precisely Long Short Term Memory (LSTM) networks, we learn to effectively use the history. The networks have the capacity to encode long-term dependencies in the sequence of observations. Traditionally, similarity scores in a graph-based tracking framework are only computed given the observation from the previous frame, i.e., a pairwise similarity score [80, 1, 32, 60]. Our proposed similarity score is computed by reasoning on the sequence of observations. In Section 4.2, we demonstrate the power of our metric to reason on a sequence of variable length as opposed to a pairwise similarity score.

Our overarching model shown in figure 2 is an LSTM network which we construct over the already pre-trained ap-

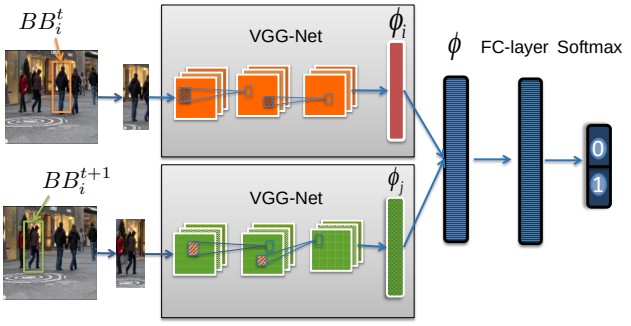

Figure 3. Our appearance model. The inputs are the two bounding boxes BB_i and BB_j we wish to compare. The output is a similarity score. We use a Siamese Convolutional Neural Network (CNN) to represent each image.

pearance, motion, and interaction modules. This LSTM is trained to perform the task of data association – output the score of whether a detection corresponds to a target using a standard Softmax classifier and cross-entropy loss. An important point, is that we train this LSTM with fine-tuning the weights of the individual components of the framework in an end-to-end fashion, which are each in fact pre-trained separately. In the remaining of this section, we describe the learned representation for each cue.

3.2. Appearance

We present the appearance model that we integrate into our framework for multi-object tracking. Recall that, our problem is fundamentally based on addressing the challenge of data association: Given a set of targets T_t at time step t, and a set of candidate detections D_{t+1} at timestep $t+1$, we would like to compute all of the valid pairings that exist between members of T_t and D_{t+1}.

The idea underlying our appearance model is that we can compute the similarity score between a target and candidate detection based on purely visual cues. More specifically, we can treat this problem as a specific instance of *re-identification*, where the goal is to take pairs of bounding boxes and determine if their content corresponds to the same person. We thus desire our appearance model to recognize the subtle similarities between input pairs, as well as be robust to occlusions and other visual disturbances. To approach this problem, we construct a Siamese Convolutional Neural Network (CNN), whose structure is depicted in Figure 3.

Architecture: Let BB_i^t and BB_j^{t+1} represent the two bounding boxes we wish to compare at time t and $t+1$. We first crop the images containing BB_i^t and BBB_j^{t+1} to contain only the bounding boxes themselves, while also ensuring that we include some amount of the surrounding image context.

The Siamese Convolutional Neural Network (CNN) accepts the raw content within each bounding box and passes it through its layers until it finally produces a 500-dimensional feature vector for each of the two inputs. Let ϕ_i and ϕ_j thus be such feature vectors corresponding to BB_i and BB_j respectively. To compute the similarity between two bounding boxes, we then simply concatenate the two vectors to get a 1000-dimensional vector $\phi = \phi_i || \phi_j$ and pass this as input to a final fully-connected layer. We lastly apply a Softmax classifier, which outputs the probabilities for the positive and negative classes, where positive indicates that the inputs match, and negative indicates otherwise.

Note that we use a network that consists of two 16-layer VGG net. In our case, we begin with the pre-trained weights of this network, but remove the last fully-connected layer and add a fully-connected layer of size 500 so that the network now outputs a 500-dimensional vector.

3.3. Motion

The second cue of our overall framework is the independent motion prior for each target. It can help tracking objects that are occluded or lost, since it provides a heuristic as to where these objects might be located.

One key challenge is to handle the noisy detections. Even when the real motion of a target is linear, the corresponding detections are noisy and lead to a non-linear sequence of coordinates hence velocities – especially if we reason in the image plane. To handle the noise in the sequence, we construct a Long Short-Term Memory (LSTM) network that learns to reason on the sequence of 2D *velocities* (see figure 4).

Architecture: Let the velocity of target i at the j-th timestep be defined as:

$v_j^i = (vx_j^i, vy_j^i) = (x_j^i - x_{j-1}^i, y_j^i - y_{j-1}^i)$,

where (x_j^i, y_j^i) are the 2D coordinates of each target in the image plane (center of the bounding boxes).

Our LSTM accepts as inputs the velocities of a single target for timesteps $1, \ldots, t$ and produces a H-dimensional output ϕ_i. We also pass the velocity vector for timestep $t+1$ (which we wish to determine whether it corresponds to a true trajectory or not) through a fully-connected layer that maps it to H-dimensional vector space which results into ϕ_j (as illustrated in Figure 4). The LSTM output is then concatenated with this vector, and the result is passed to another fully connected layer which brings the $2H$ dimensional vector to the space of k features. Finally, another fully connected layer reduces the dimension to 2 which will be used as the $0/1$ classification problem during the training. Note that when combining with other cues, we keep ϕ_M, the vector before the final fully connected layer.

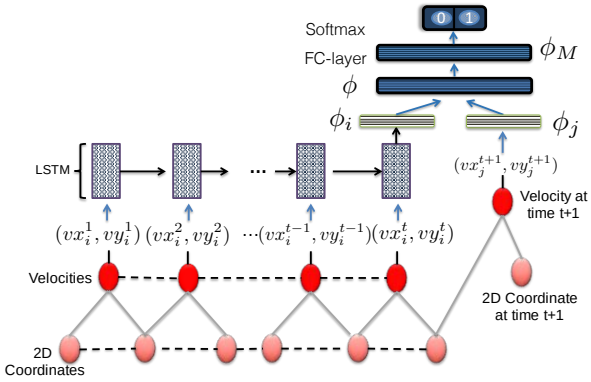

Figure 4. Our motion prior model. The inputs are the 2D coordinates of the target in the image plane converted into velocities. The output is the similarity score of an observed velocity at time $t+1$.

3.4. Interaction model

The motion of a particular target is governed not only by its own previous motion properties, but also by the behavior of nearby targets. We, therefore, decide to incorporate this cue into our overall framework by formulating an Interaction model.

The number of nearby targets can vary. In order to use the same size input, we model the neighborhood of each target as a fixed size occupancy grid. Then, for each target, we use an LSTM network to model the sequence of occupancy grids (see Figure 6).

Architecture: Let $O_i^1, O_i^2, \ldots O_i^t$ represent the 2D occupancy grid for trajectory of target i. The positions of all the neighbors are pooled in this map. The m, n element of the map is simply given by:

$$O_i^t(m,n) = \sum_{j \in \mathcal{N}_i} \mathbb{1}_{mn}[x_j^t - x_i^t, y_j^t - y_i^t] \qquad (1)$$

Where $\mathbb{1}_{mn}[x,y]$ is an indicator function to check if there is a person in (x,y) cell of the grid, and $\mathcal{N}_i$ is the set of neighbors corresponding to person i. The map is further represented as a vector (see Figure 6). Note that all the 2D locations of targets are their equivalent bounding box centers on the image plane.

Our interaction LSTM accepts as input the occupancy grids centered on a specific target for timesteps $1, \ldots, t$ and produces H-dimensional output ϕ_i for each timestep. We also pass the occupancy grid for timestep $t+1$ (which we wish to determine whether it corresponds to a true trajectory or not) through a fully-connected layer that maps it to H-dimensional vector space and obtains ϕ_j. The LSTM output is then concatenated with this vector and the result is passed to another fully connected layer which brings the $2H$ dimensional vector to the space of k features. Finally, another fully connected layer reduces the dimension to 2 which will be used as the $0/1$ classification problem (as illustrated in Figure 5). Similar to motion model, when combining with other cues, we keep ϕ_I, the vector before the final fully connected layer.

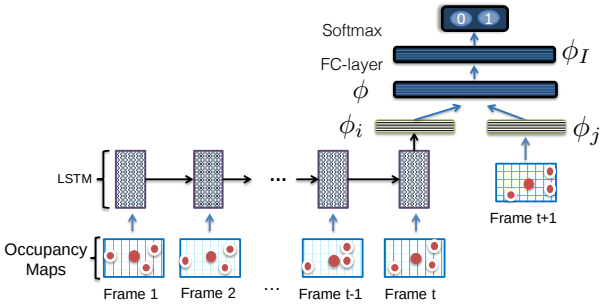

Figure 5. Our Interaction model. The inputs are the occupancy maps across time and the output is the similarity score of an observed velocity at time $t+1$.

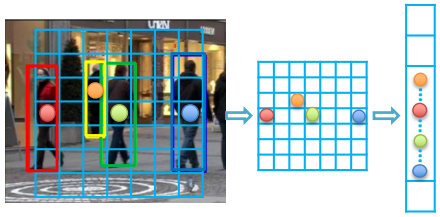

Figure 6. Illustration of the steps involved computing the occupancy map. The location of bounding box center of nearby targets are encoded in a grid –the occupancy map– centered on the target. For implementation purposes, the map is represented as a vector.

4. Experimental results

We have presented our multi-cue metric learning framework to compute the similarity score between a sequence of observations and a new detection. We use our learned metric to tackle the Multi-object Tracking problem. We first present the overall performance of our framework on the MOT challenge [34] and then present more insights on our metric.

4.1. Multi-object tracking

To recall, we use our learned representation in the MDP framework [69]. We have one target LSTM for each target, and the MDP framework tracks the targets using the similarity computed with our learned representation.

Metrics. We report the same metric as the suggested ones in the MOT2D Benchmark challenge [34]: Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), Mostly Track targets (MT), Mostly Lost targets (ML), False Positives (FP), False Negatives (FN), ID

Method	MOTA	MOTP	Rcll	Prcn	MT	ML
MDP [69] + Lin	51.5	74.2	74.1	80.1	44.2%	20.9%
MDP + SF [71]	73.5	77.1	84.4	91.5	58.1%	25.5%
MDP + SF-mc [55]	75.6	78.2	86.1	92.6	60%	23.2%
Ours (MOT)	78.6	79.4	88.2	93.9	69.7%	19.5%
Ours (MOT+Drone)	**82.9**	**80.3**	**92.3**	**95.3**	**85%**	**15.2%**

Table 1. MOT tracking results on Stanford Drone Dataset. Our MDPNN (MOT) has been trained on the MOT training data [] and has not been fine-tuned on the Stanford drone dataset. Whereas, our MDPNN (MOT+Drone) has been fine-tuned on the drone dataset.

Switches (IDS), and finally the number of frameworks processed in one second (Hz) which indicates the speed of a tracking method.

MOT2D Benchmark. We report the quantitative results of our method on the 2DMOT 2015 Benchmark [34] in Table 4. This challenge shares the training and testing set for 11 sequences. We used their publicly shared noisy detections. Our method outperforms previous methods on multiple metrics such as the MOTA, MT, and ML. Our MOTA even outperforms offline methods that have access to the whole set of detections to reason on the data association step. Our IDS is still high since our method seeks to recover back to the right target after an occlusion or drift; hence we have higher MT and lower ML. Indeed, when targets are occluded, our method can wrongly assign them to other detections. But when the targets re-appear, our method re-match them with the correct detections. Such process leads to a high number of switches. Nevertheless, the MT metric remains high.

The impact of our learned metric becomes evident compared with the previously published MDP method. By only switching the metric and keeping the same data association method proposed in [69], we obtain a 20% relative boost in MOTA. The benefits of our metric are further emphasized with the Stanford dataset [55].

Stanford Drone Dataset. As we have mentioned before, one of the main advantages of our model compared to other multi-object tracking methods is the similarity score used for object representation. Our representation is a function of multiple cues across time and seeks to use the right cues at each time. Often, some cues should vote for the similarity score since the others are not discriminant enough or very noisy. To test the power of our method, we also conduct experiments by testing our multi-object tracking experiments on videos that are very different from the MOT challenge [34], i.e., the Stanford Drone Dataset [55]. All targets are small and hence appearance models might be faulty (as illustrated in Figure 10). In table 1, we compare our method with previously reported MDP-based methods. Our method outperforms all the MDP-based methods on all metrics. Even without fine-tuning our metric on the drone dataset, our method outperforms previous works. After fine-tuning, we obtain the best performance as expected. It shows the power of a data-driven method to learn a metric over any input signal.

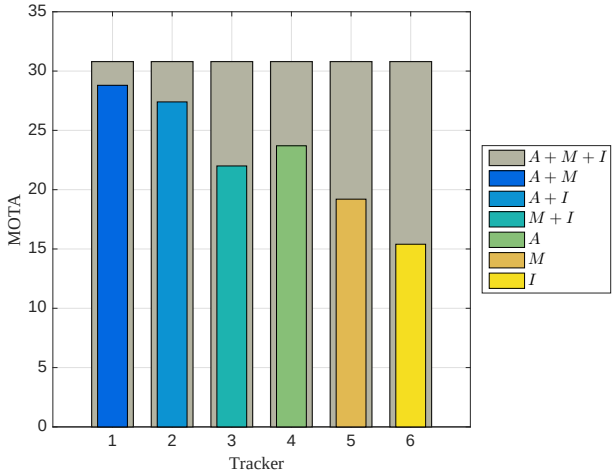

Figure 7. Analysis of our model on the validation set using different set of components (A) Appearance, (M) Motion, and (I) Interaction. We report the MOTA scores.

In the reminder of this section, we analyze the performance of our metric with an ablation study as well as more insights on our appearance on more specific tasks.

4.2. Ablation study

The underlying motivation of our proposed framework is to address the following two challenges (as listed in the introduction): effectively modeling the history of each cue, and effectively combining multiple cues. We now present experiments towards these two goals on the validation set of the MOT2D challenge [34]. We use the same evaluation protocol (training and test splits) as in [69] for our validation set.

Impact of the History: One of the advantages of our metric compared to the previous ones is the capacity to memorize long term dependencies of cues across time, i.e., retaining information from the past. We investigate the impact of changing the sequence length of the LSTMs

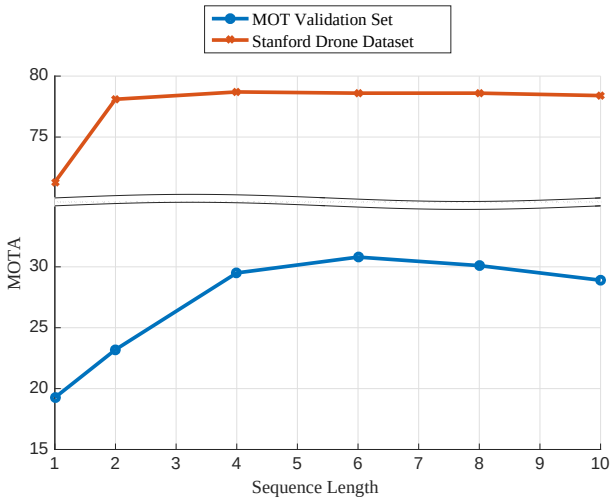

Figure 8. Analysis of the used sequence length (memory) for our model on the validation set for both datasets. We report the MOTA scores.

Tracker	MOTA	MOTP	MT	ML	FP	FN	IDS
A+M+I	**30.8**	73.8	**14**	**51.7**	**2,563**	**13,127**	**98**
A+M	28.8	**73.9**	13.5	52.1	2,776	13,361	134
A+I	27.4	73.9	12	53.4	2,679	13,991	136
M+I	22	73.8	9.8	52.1	2,714	14,954	298
A	23.7	73.7	11.5	55.6	3,359	14,001	138
M	19.2	73.7	8.5	68.4	3,312	15,023	313
I	15.4	73.5	5.6	69.9	3,061	16,250	354

Table 2. Analysis of our model on the validation set using different set of components (A) Appearance, (M) Motion, and (I) Interaction. We report the standard MOT metrics.

on tracking accuracy, where sequence length of an LSTM is the number of unrolled time steps used while training the LSTM. Figure 8 shows the MOTA score of different components for the validation set, under different sequence length for our target LSTM. We can see that increasing the sequence length positively impacts the MOTA. The performance saturates after 3 frames (on the Stanford drone dataset) and after 6 frames on the MOT dataset. These results echo our claim that RNN can effectively model the history of a cue. Note that the MOTA slightly decreases with 10 timesteps on the MOT dataset. It can be explained by the vanishing gradients. Moreover, the difference between the MOT and Stanford dataset can be explained by the difference in the challenges. The drone dataset does not have any occlusion whereas on the MOT has full long-term occlusions. Nevertheless, we can see that modeling the sequence of observations on both datasets positively impacts the similarity score hence tracking performance.

Impact of multiple cues. We investigate the contribution of different cues in our framework by measuring the performance in terms of MOTA on the validation set. Figure 7 presents the results of our ablation study. The appearance cue is the most important one. Each cues helps to increase the performance. It is worth to point out our proposed interaction cue positively impacts the overall performance. Our proposed target LSTM (in charge of combining all the other RNNs) effectively reason on all the cues to increase the performance. Table 2 reports more details on the impact of each cue on the various tracking metrics.

Method	Rank 1	Rank 5	Rank 10
FPNN [37]	19.9	49.3	64.7
BoW [85]	23	45	55.7
BoW + HS [85]	24.3	–	–
ConvNet [2]	45	75.3	95
LX [38]	46.3	78.9	88.6
MLAPG [39]	51.2	83.6	92.1
SS-SVM [82]	51.2	80.8	89.6
SI-CI [67]	52.2	**84.3**	92.3
DNS [81]	54.7	84.8	94.8
SLSTM [64]	**57.3**	80.1	88.3
Ours	55.9	81.7	**95.1**

Table 3. Rank-1 re-identification rates for automatically labeled pedestrian bounding boxes.

Figure 9. Qualitative results on the CUHK03 dataset [37]. The images with green border are correctly labeled as same and images with red border are correctly labeled as different people.

4.3. Appearance model: Re-identification

For completeness, we report the performance of our appearance cue on other tasks. We train our network on positive and negative samples extracted from two MOT2D and CUHK03 datasets [34, 37]. MOT2D dataset contains 11 scenes with more than $5,000$ frames that contain around $40,000$ objects. We extracted more than 500k of positive and negative samples. And second CUHK03 dataset that contains 13164 images of 1360 pedestrians and contains 150k pairs. For positive pairs, we use instances of the same

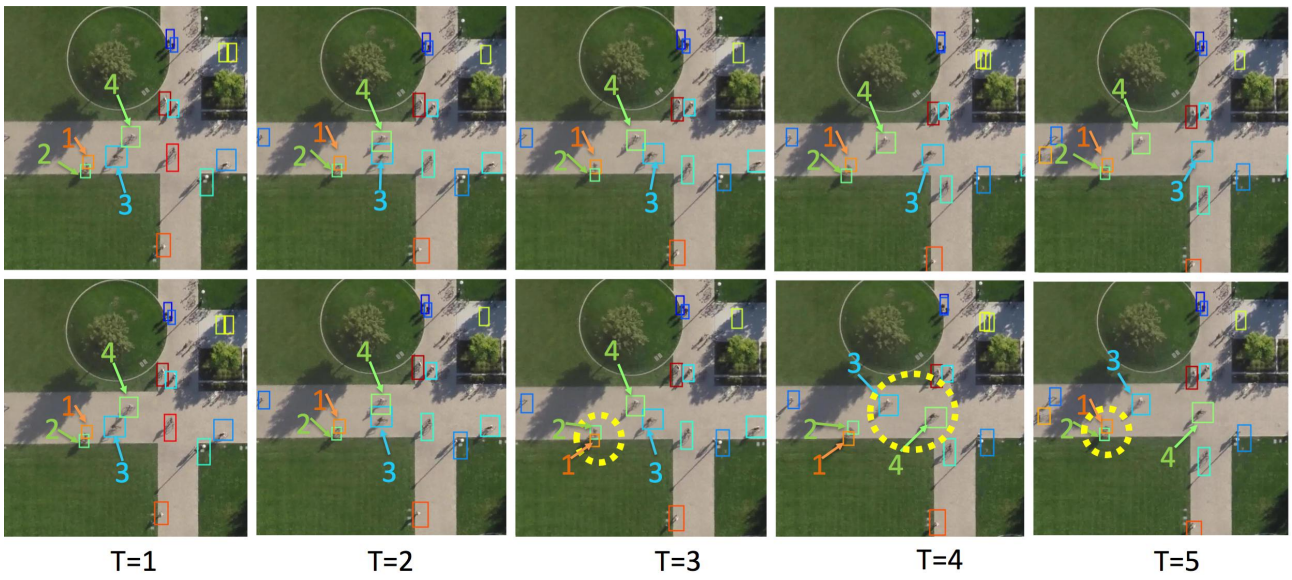

Figure 10. Qualitative results on the Stanford Drone dataset [55]. The first row presents the tracking results of our MDPNN method whereas the second row presents the results of MDP+SF-mc [55]. The dashed circles illustrate ID switches in previous method.

Tracker	Tracking Mode	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓	IDS↓	Frag ↓	Hz ↑
SiameseCNN [33]	Offline	29.0	71.2	8.5%	48.4%	**5,160**	37,798	639	1,316	**52.8**
CNNTCM [66]	Offline	29.6	71.8	11.2%	44.0%	7,786	34,733	712	943	1.7
MHT_DAM [29]	Offline	32.4	71.8	16.0%	43.8%	9,064	32,060	**435**	826	0.7
NOMT [8]	Offline	33.7	71.9	12.2%	44.0%	7,762	32,547	442	**823**	11.5
TSMLCDEnew [65]	Offline	34.3	71.7	14.0%	39.4%	7,869	31,908	618	959	6.5
JointMC [26]	Offline	35.6	71.9	**23.2**%	39.3%	10,580	**28,508**	457	969	0.6
TC_ODAL [5]	**Online**	15.1	70.5	3.20%	55.80%	12,970	38,538	637	1,716	1.7
RMOT [75]	**Online**	18.6	69.6	5.30%	53.30%	12,473	36,835	684	**1,282**	**7.9**
SCEA [73]	**Online**	29.1	71.1	8.9%	47.3%	**6,060**	36,912	604	1,182	6.8
MDP [69]	**Online**	30.3	71.3	13.00%	38.40%	9,717	32,422	680	1,500	1.1
TDAM [74]	**Online**	33.0	**72.8**	13.3%	39.1%	10,064	**30,617**	**464**	1,506	5.9
Ours	**Online**	**36.6**	71.4	**13.3**%	**36.5**%	6,419	31,811	700	1,458	1.0

Table 4. Tracking performance on the test set of the 2D MOT 2015 Benchmark with public detections.

target that occur in different frames. For negative examples, we use pairs of different targets that may span across all frames. Network hyperparameters are chosen by cross validation. The mini-batch size of 64, learning rate of 0.001, sequentially decreased every 2 epochs by a factor 10 (for 20 epochs).

We evaluate our appearance model on CUHK03 reidentification benchmark [83]. Table 3 presents our results for Rank 1, Rank 5, and Rank 10 accuracies. Our method achieves 55.9 percent of accuracy for Rank 1 which is competitive against the state-of-the-art method (57.3%). When measuring the re-identification rate for Rank 10, our appearance model outperforms previous methods. This is a crucial indicator for showing that our appearance model can extract meaningful feature representation for re-identification task. Figure 9 shows the qualitative results of our model on CUHK03 dataset.

5. Conclusions

We have presented a metric learning framework that encodes dependencies across multiple cues over a temporal window. Our learned multi-cue metric is used to compute the similarity scores in a tracking framework. We showed that by switching the existing state-of-the-art metric with our proposed one, the tracking performance (measured as MOTA) increases by 20 %. Consequently, our method ranks first in the MOT challenge and Stanford drone dataset. As future work, since our framework is generic to any collective behavior, we plan to use it to track challenging targets such as ants behavior. Their appearance and dynamics are quite challenging and different from humans. It will be exciting to learn a metric for such collective behavior and help researchers in biology to get more insights in their field.

References

[1] A. Adam, E. Rivlin, and I. Shimshoni. Robust fragments-based tracking using the integral histogram. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 798–805. IEEE, 2006.

[2] E. Ahmed, M. Jones, and T. K. Marks. An improved deep learning architecture for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3908–3916, 2015.

[3] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social lstm: Human trajectory prediction in crowded spaces.

[4] S. Ali and M. Shah. Floor fields for tracking in high density crowd scenes. In *Computer Vision–ECCV 2008*, pages 1–14. Springer, 2008.

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

[6] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1515–1522. IEEE, 2009.

[7] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[8] W. Choi. Near-online multi-target tracking with aggregated local flow descriptor. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3029–3037, 2015.

[9] W. Choi and S. Savarese. Multiple target tracking in world coordinate with single, minimally calibrated camera. In *Computer Vision–ECCV 2010*, pages 553–567. Springer, 2010.

[10] C. Dicle, O. Camps, and M. Sznaier. Tracking multiple targets with similar appearance.

[11] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.

[12] A. Ess, B. Leibe, K. Schindler, and L. Van Gool. A mobile vision system for robust multi-person tracking. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[13] A. Ess, K. Schindler, B. Leibe, and L. Van Gool. Improved multi-person tracking with active occlusion handling. In *ICRA Workshop on People Detection and Tracking*, volume 2. Citeseer, 2009.

[14] E. Fontaine, A. H. Barr, and J. W. Burdick. Model-based tracking of multiple worms and fish. In *ICCV Workshop on Dynamical Vision*. Citeseer, 2007.

[15] Freepik. Black silhouettes of mans and woman walking. Designed by Freepik.com.

[16] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via on-line boosting. In *BMVC*, volume 1, page 6, 2006.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[18] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

[19] D. Helbing and P. Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[20] D. Held, S. Thrun, and S. Savarese. Learning to track at 100 FPS with deep regression networks. *CoRR*, abs/1604.01802, 2016.

[21] S. Hong and B. Han. Visual tracking by sampling tree-structured graphical models. In *European Conference on Computer Vision*, pages 1–16. Springer, 2014.

[22] M. Hu, S. Ali, and M. Shah. Detecting global motion patterns in complex videos. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–5. IEEE, 2008.

[23] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *Computer Vision–ECCV 2008*, pages 788–801. Springer, 2008.

[24] H. Izadinia, V. Ramakrishna, K. M. Kitani, and D. Huber. Multi-pose multi-target tracking for activity understanding. In *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*, pages 385–390. IEEE, 2013.

[25] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. *arXiv preprint arXiv:1511.05298*, 2015.

[26] M. Keuper, S. Tang, Y. Zhongjie, B. Andres, T. Brox, and B. Schiele. A multi-cut formulation for joint segmentation and tracking of multiple objects. *arXiv preprint arXiv:1607.06317*, 2016.

[27] Z. Khan, T. Balch, and F. Dellaert. An mcmc-based particle filter for tracking multiple interacting targets. In *Computer Vision-ECCV 2004*, pages 279–290. Springer, 2004.

[28] B. Y. S. Khanloo, F. Stefanus, M. Ranjbar, Z.-N. Li, N. Saunier, T. Sayed, and G. Mori. A large margin framework for single camera offline tracking with hybrid cues. *Computer Vision and Image Understanding*, 116(6):676–689, 2012.

[29] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg. Multiple hypothesis tracking revisited. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4696–4704, 2015.

[30] L. Kratz and K. Nishino. Tracking with local spatio-temporal motion patterns in extremely crowded scenes. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 693–700. IEEE, 2010.

[31] L. Kratz and K. Nishino. Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes. *IEEE transactions on pattern analysis and machine intelligence*, 34(5):987–1002, 2012.

[32] C.-H. Kuo and R. Nevatia. How does person identity recognition help multi-person tracking? In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1217–1224. IEEE, 2011.

[33] L. Leal-Taixé, C. Canton-Ferrer, and K. Schindler. Learning by tracking: Siamese cnn for robust target association. *arXiv preprint arXiv:1604.07866*, 2016.

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

[35] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(10):1683–1698, 2008.

[36] K. Li, E. D. Miller, M. Chen, T. Kanade, L. E. Weiss, and P. G. Campbell. Cell population tracking and lineage construction with spatiotemporal context. *Medical image analysis*, 12(5):546–566, 2008.

[37] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 152–159, 2014.

[38] S. Liao, Y. Hu, X. Zhu, and S. Z. Li. Person re-identification by local maximal occurrence representation and metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2197–2206, 2015.

[39] S. Liao and S. Z. Li. Efficient psd constrained asymmetric metric learning for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3685–3693, 2015.

[40] C.-W. Lu, C.-Y. Lin, C.-Y. Hsu, M.-F. Weng, L.-W. Kang, and H.-Y. M. Liao. Identification and tracking of players in sport videos. In *Proceedings of the Fifth International Conference on Internet Multimedia Computing and Service*, pages 113–116. ACM, 2013.

[41] W. Luo, T.-K. Kim, B. Stenger, X. Zhao, and R. Cipolla. Bi-label propagation for generic multiple object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1290–1297, 2014.

[42] E. Meijering, O. Dzyubachyk, I. Smal, and W. A. van Cappellen. Tracking in cell and developmental biology. In *Seminars in cell & developmental biology*, volume 20, pages 894–902. Elsevier, 2009.

[43] A. Milan, S. Roth, and K. Schindler. Continuous energy minimization for multitarget tracking. *IEEE transactions on pattern analysis and machine intelligence*, 36(1):58–72, 2014.

[44] P. Mordohai and G. Medioni. Dimensionality estimation, manifold learning and function approximation using tensor voting. *Journal of Machine Learning Research*, 11(Jan):411–450, 2010.

[45] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38, 1957.

[46] H. Nam and B. Han. Learning multi-domain convolutional neural networks for visual tracking. *arXiv preprint arXiv:1510.07945*, 2015.

[47] P. Nillius, J. Sullivan, and S. Carlsson. Multi-target tracking-linking identities using bayesian network inference. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2187–2194. IEEE, 2006.

[48] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *Computer Vision-ECCV 2004*, pages 28–39. Springer, 2004.

[49] S. Oron, A. Bar-Hille, and S. Avidan. Extended lucas-kanade tracking. In *European Conference on Computer Vision*, pages 142–156. Springer, 2014.

[50] S. Oron, A. Bar-Hillel, and S. Avidan. Real-time tracking-with-detection for coping with viewpoint change. *Machine Vision and Applications*, 26(4):507–518, 2015.

[51] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 261–268. IEEE, 2009.

[52] S. Pellegrini, A. Ess, and L. Van Gool. Improving data association by joint modeling of pedestrian trajectories and groupings. In *European Conference on Computer Vision*, pages 452–465. Springer, 2010.

[53] A. Petrovskaya and S. Thrun. Model based vehicle detection and tracking for autonomous urban driving. *Autonomous Robots*, 26(2-3):123–139, 2009.

[54] E. Ristani and C. Tomasi. Tracking multiple people online and in real time. In *Asian Conference on Computer Vision*, pages 444–459. Springer, 2014.

[55] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *European Conference on Computer Vision*, pages 549–565. Springer, 2016.

[56] M. Rodriguez, S. Ali, and T. Kanade. Tracking in unstructured crowded scenes. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1389–1396. IEEE, 2009.

[57] R. Sanchez-Matilla, F. Poiesi, and A. Cavallaro. Online multi-target tracking with strong and weak detections.

[58] P. Scovanner and M. F. Tappen. Learning pedestrian dynamics from the real world. In *ICCV*, volume 9, pages 381–388, 2009.

[59] K. Shafique, M. W. Lee, and N. Haering. A rank constrained continuous formulation of multi-frame multi-target tracking problem. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[60] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah. Part-based multiple-person tracking with partial occlusion handling. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1815–1821. IEEE, 2012.

[61] F. Solera, S. Calderara, E. Ristani, C. Tomasi, and R. Cucchiara. Tracking social groups within and across cameras. *IEEE Transactions on Circuits and Systems for Video Technology*.

[62] C. Spampinato, Y.-H. Chen-Burger, G. Nadarajan, and R. B. Fisher. Detecting, tracking and counting fish in low quality unconstrained underwater videos. *VISAPP (2)*, 2008:514–519, 2008.

[63] C. Spampinato, S. Palazzo, D. Giordano, I. Kavasidis, F.-P. Lin, and Y.-T. Lin. Covariance based fish tracking in real-life underwater environment. In *VISAPP (2)*, pages 409–414, 2012.

[64] R. R. Varior, B. Shuai, J. Lu, D. Xu, and G. Wang. A siamese long short-term memory architecture for human re-identification. In *European Conference on Computer Vision*, pages 135–153. Springer, 2016.

[65] B. Wang, G. Wang, K. L. Chan, and L. Wang. Tracklet association by online target-specific metric learning and coherent dynamics estimation. 2016.

[66] B. Wang, L. Wang, B. Shuai, Z. Zuo, T. Liu, K. Luk Chan, and G. Wang. Joint learning of convolutional neural networks and temporally constrained metrics for tracklet association. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8, 2016.

[67] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang. Joint learning of single-image and cross-image representations for person re-identification.

[68] Z. Wu, A. Thangali, S. Sclaroff, and M. Betke. Coupling detection and data association for multiple object tracking. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1948–1955. IEEE, 2012.

[69] Y. Xiang, A. Alahi, and S. Savarese. Learning to track: Online multi-object tracking by decision making. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4705–4713, 2015.

[70] J. Xing, H. Ai, L. Liu, and S. Lao. Multiple player tracking in sports video: A dual-mode two-way bayesian inference approach with progressive observation modeling. *Image Processing, IEEE Transactions on*, 20(6):1652–1667, 2011.

[71] K. Yamaguchi, A. C. Berg, L. E. Ortiz, and T. L. Berg. Who are you with and where are you going? In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1345–1352. IEEE, 2011.

[72] B. Yang and R. Nevatia. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1918–1925. IEEE, 2012.

[73] B. Yang and R. Nevatia. An online learned crf model for multi-target tracking. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2034–2041. IEEE, 2012.

[74] M. Yang and Y. Jia. Temporal dynamic appearance modeling for online multi-person tracking. *Computer Vision and Image Understanding*, 2016.

[75] J. H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon. Bayesian multi-object tracking using motion context from multiple objects. In *2015 IEEE Winter Conference on Applications of Computer Vision*, pages 33–40. IEEE, 2015.

[76] Q. Yu, G. Medioni, and I. Cohen. Multiple target tracking using spatio-temporal markov chain monte carlo data association. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[77] A. R. Zamir, A. Dehghan, and M. Shah. Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs. In *Computer Vision–ECCV 2012*, pages 343–356. Springer, 2012.

[78] M. Zhai, M. J. Roshtkhari, and G. Mori. Deep learning of appearance models for online object tracking. *arXiv preprint arXiv:1607.02568*, 2016.

[79] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu. Crowd analysis: a survey. *Machine Vision and Applications*, 19(5-6):345–357, 2008.

[80] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[81] L. Zhang, T. Xiang, and S. Gong. Learning a discriminative null space for person re-identification. *arXiv preprint arXiv:1603.02139*, 2016.

[82] Y. Zhang, B. Li, H. Lu, A. Irie, and X. Ruan. Sample-specific svm learning for person re-identification.

[83] W. L. R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification.

[84] X. Zhao, D. Gong, and G. Medioni. Tracking using motion patterns for very crowded scenes. In *Computer Vision–ECCV 2012*, pages 315–328. Springer, 2012.

[85] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1116–1124, 2015.

Tracking The Untrackable: Learning To Track Multiple Cues with Long-Term Dependencies (Supplementary material)

Amir Sadeghian, Alexandre Alahi, Silvio Savarese
Stanford University
{amirabs,alahi,ssilvio}@cs.stanford.edu

1. Preliminary remarks

In this supplementary material, we provide:

- Our results on MOT16 benchmark.
- More qualitative results on our proposed tracking method.
- A video to present an overview of our work.

2. MOT16 benchmark

We report the quantitative results on the MOT16 benchmark [10] in Table 2. The conclusions are similar to the MOT15 results (already in the paper). This challenge contains 14 video sequences (7 training, 7 test) in unconstrained environments filmed with both static and moving cameras. We used their publicly shared noisy detections. The metrics used to evaluate the multiple object tracking performance as suggested by the MOT Benchmark is shown in Table 1. Our method outperforms previous methods on multiple metrics such as the MOTA, MOTP, MT, ML, and FP. Our IDS is still high since our method seeks to recover back to the right target after an occlusion or drift; hence we have higher MT and lower ML. Indeed, when targets are occluded, our method can wrongly assign them to other detections. But when the targets re-appear, our method re-match them with the correct detections. Such process leads to a high number of switches. Nevertheless, the MT metric remains high.

Table 3 and 4 present detailed tracking evaluation of our framework on the sequences in the test set of the 2DMOT15 and MOT16 benchmark, respectively.

3. Qualitative results

In this section, we present more qualitative results on the performance of our tracking method. Figure 1 and 2 show sampled tracking results on sequences in the test set of 2DMOT2015 and MOT16 benchmark, respectively.

References

[1] Y. Ban, S. Ba, X. Alameda-Pineda, and R. Horaud. Tracking multiple persons based on a variational bayesian model. In *European Conference on Computer Vision*, pages 52–67. Springer, 2016.

[2] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008(1):1–10, 2008.

[3] W. Choi. Near-online multi-target tracking with aggregated local flow descriptor. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3029–3037, 2015.

[4] 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*, pages 774–790. Springer, 2016.

[5] H. Kieritz, S. Becker, W. Hübner, and M. Arens. Online multi-person tracking using integral channel features. In *Advanced Video and Signal Based Surveillance (AVSS), 2016 13th IEEE International Conference on*, pages 122–130. IEEE, 2016.

[6] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg. Multiple hypothesis tracking revisited. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4696–4704, 2015.

[7] N. Le, A. Heili, and J.-M. Odobez. Long-term time-sensitive costs for crf-based tracking by detection. In *European Conference on Computer Vision*, pages 43–51. Springer, 2016.

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

[9] Y. Li, C. Huang, and R. Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2953–2960. IEEE, 2009.

[10] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

MOTA	Multiple Object Tracking Accuracy. This measure combines three error sources: false positives, missed targets and identity switches
MOTP	Multiple Object Tracking Precision. The misalignment between the annotated and the predicted bounding boxes
GT	The total number of ground truth trajectories.
MT	Mostly tracked targets. Percentage of ground truth trajectories that are covered by tracking output for at least 80% of their respective life span
ML	Mostly lost targets. Percentage of ground truth trajectories that are covered by tracking output less than 20% of their respective life span
FP	The total number of false positives.
FN	The total number of false negatives (missed targets).
IDS	The total number of identity switches.
Frag	The total number of times a trajectory is fragmented (i.e. interrupted during tracking).

Table 1. Evaluation metrics used for multi-object tracking. [2, 9]

[11] R. Sanchez-Matilla, F. Poiesi, and A. Cavallaro. Online multi-target tracking with strong and weak detections. In *European Conference on Computer Vision*, pages 84–99. Springer, 2016.

[12] B. Yang and R. Nevatia. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1918–1925. IEEE, 2012.

Tracker	Tracking Mode	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓	IDS↓	Frag ↓	Hz ↑
LTTSC-CRF [7]	Offline	37.6	75.9	9.60%	55.20%	11,969	101,343	481	1,012	0.6
LINF1 [4]	Offline	41	74.8	11.60%	51.30%	7,896	99,224	430	963	1.1
MHT_DAM [6]	Offline	42.9	76.6	13.60%	46.90%	5,668	97,919	499	659	0.8
JMC [12]	Offline	46.3	75.7	15.50%	**39.70%**	6,373	90,914	657	1,114	0.8
NOMT [3]	Offine	**46.4**	**76.6**	**18.30%**	41.40%	9,753	**87,565**	**359**	**504**	2.6
OVBT [1]	**Online**	38.4	75.4	7.50%	47.30%	11,517	99,463	1,321	2,140	0.3
EAMTT_pub [11]	**Online**	38.8	75.1	7.90%	49.10%	8,114	102,452	965	1,657	**11.8**
oICF [5]	**Online**	43.2	74.3	11.30%	48.50%	6,651	**96,515**	**381**	**1,404**	0.4
Ours	**Online**	**43.8**	**75.5**	**12.40%**	**40.70%**	**3,501**	98,193	723	2,036	1

Table 2. Tracking performance on the test set of the MOT16 Benchmark with public detections.

Figure 1. Qualitative results on the 2dMOT 2015 benchmark [8].

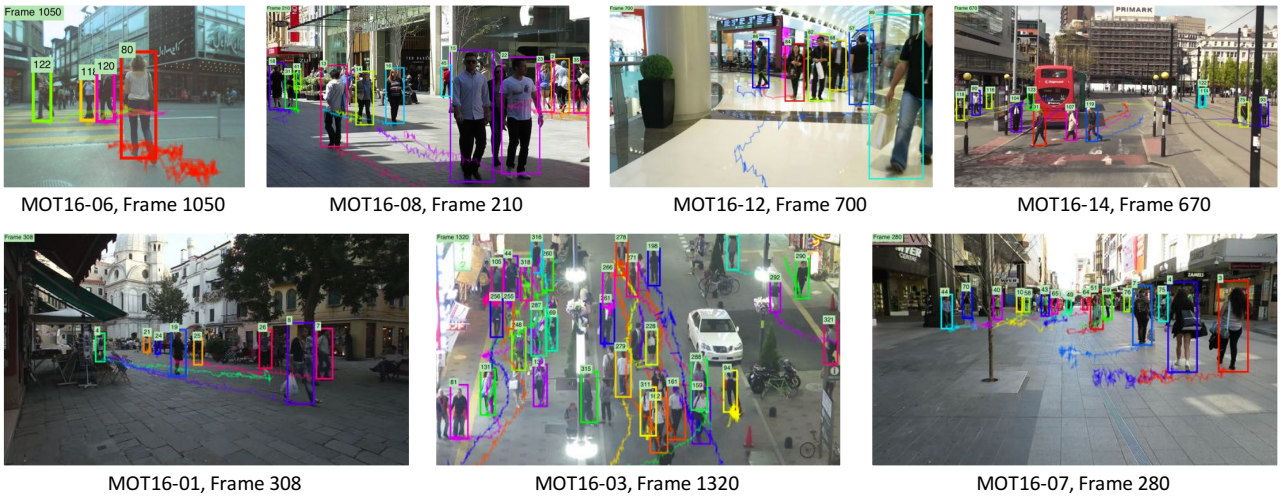

Figure 2. Qualitative results on the MOT16 benchmark [10].

Sequence	MOTA	MOTP	FAF	GT	MT	ML	FP	FN	ID Sw	Frag
TUD-Crossing	69.4	74.2	0	13	46.20%	7.70%	10	310	17	28
PETS09-S2L2	45.3	70.3	1	42	9.50%	9.50%	415	4,636	218	326
ETH-Jelmoli	45.1	73.7	0.6	45	17.80%	37.80%	261	1,114	18	61
ETH-Linthescher	28.8	74.8	0.1	197	8.10%	62.90%	109	6,207	40	102
ETH-Crossing	27.8	73.5	0.1	26	7.70%	57.70%	11	713	0	16
AVG-TownCentre	35.9	69.8	2.2	226	19.00%	29.60%	978	3,498	108	252
ADL-Rundle-1	27.3	71.4	5.3	32	12.50%	12.50%	2,671	4,007	87	206
Total	**36.6**	**71.4**	**1.1**	**581**	**13.30%**	**36.50%**	**6,419**	**31,811**	**700**	**1,458**

Table 3. Individual tracking performance on test set of the 2DMOT 2015 Benchmark.

Sequence	MOTA	MOTP	FAF	GT	MT	ML	FP	FN	ID Sw	Frag
MOT16-01	35.5	72.1	0.3	23	17.40%	39.10%	154	3,944	25	99
MOT16-03	49	75.5	1	148	16.20%	20.90%	1,548	51,429	315	1,171
MOT16-06	49.2	74	0.2	221	14.00%	43.90%	213	5,555	94	156
MOT16-07	43	74.6	0.6	54	11.10%	35.20%	318	8,882	102	209
MOT16-08	31.7	79.6	0.6	63	9.50%	39.70%	391	10,941	94	144
MOT16-12	37.2	77.3	0.6	86	16.30%	43.00%	531	4,647	35	60
MOT16-14	28.6	74.9	0.5	164	5.50%	55.50%	346	12,795	58	197
Total	**43.8**	**75.5**	**0.6**	**759**	**12.40%**	**40.70%**	**3,501**	**98,193**	**723**	**2,036**

Table 4. Individual tracking performance on test set of the MOT16 Benchmark.