Customized Multi-Person Tracker

Liqian Ma[1]* Siyu Tang[2,3]* Michael J. Black[2] Luc Van Gool[1,4]

[1] KU-Leuven/PSI, TRACE (Toyota Res in Europe)
{liqian.ma, luc.vangool}@esat.kuleuven.be
[2] Max Planck Institute for Intelligent Systems
{stang, black}@tuebingen.mpg.de
[3] University of Tübingen
[4] ETH Zurich

Abstract. This work addresses the task of multi-person tracking in crowded street scenes, where long-term occlusions pose a major challenge. One popular way to address this challenge is to re-identify people before and after occlusions using Convolutional Neural Networks (CNNs). To achieve good performance, CNNs require a large amount of training data, which is not available for multi-person tracking scenarios. Instead of annotating large training sequences, we introduce a *customized multi-person tracker* that automatically adapts its person re-identification CNNs to capture the discriminative appearance patterns in a *test sequence.* We show that a few high-quality training examples that are *automatically* mined from the test sequence can be used to fine-tune pre-trained CNNs, thereby teaching them to recognize the uniqueness of people's appearance in the test sequence. To that end, we introduce a hierarchical correlation clustering (HCC) framework, in which we utilize an existing robust correlation clustering tracking model, but with different graph structures to generate local, reliable tracklets as well as globally associated tracks. We deploy intuitive physical constraints on the local tracklets to generate the high-quality training examples for customizing the person re-identification CNNs. Our customized multi-person tracker achieves state-of-the-art performance on the challenging MOT16 tracking benchmark.

Keywords: Tracking · Person re-identification · Adaptation.

1 Introduction

Tracking multiple people in unconstrained videos is crucial for many vision applications, e.g. autonomous driving, visual surveillance, crowd analytics, etc. Recent approaches for multiple-person tracking tend to use some form of tracking-by-detection [2,21,23,33,34], whereby a state-of-the-art person detector localizes the targets in each frame. Then the task of the tracker is to associate those detections over time. While achieving state-of-the-art tracking results, most approaches still struggle with challenges such as long-term occlusions, appearance changes, detection failures, etc.

* Equal contribution.

One closely related task is person re-identification (re-ID), which associates people across non-overlapping camera views. Major progress in re-ID has been made recently through the use of Convolutional Neural Networks (CNNs), also in the context of tracking [13, 17]. We argue that the full potential of CNNs for re-ID has not yet been explored, due to the lack of proper training data for the target sequences. The illumination conditions, resolution, frame-rate, camera motion and angle can all significantly differ between training and target sequences. Consequently, we would like to leverage the power of CNNs without labeling huge amounts training data for multi-person tracking. Rather than train networks in the traditional way, the key idea is to adapt them to each sequence.

Specifically, we propose an adaptation scheme to automatically customize a sequence-specific multi-person tracker. As with any adaptative tracking scheme, it is critical not to introduce tracking errors into the model, which lead to drift. The key observation is that, once we obtain reliable local tracklets on a test sequence, we can use an intuitive physical constraint that non-overlapping tracklets in the same frame are very likely to contain different people. This allows us to harvest high-quality training examples for adapting a generic re-ID CNN, with a low risk of drift. Our experiments show that this customization approach produces a significant improvement in associating people as well as in the final tracking performance.

Generating reliable local tracklets in crowded sequences is not trivial. Since person detection in such scenes can be quite noisy, resulting in false positive detections and inaccurate localizations. In this work, we build on the recent tracking formulations based on correlation clustering [26,30,31], otherwise known as the minimum cost multicut problem. This leads us to a hierarchical data association approach. At the lower level, we use the similarity measure from robust patch matching [31] to produce reliable local tracks. This allows us to automatically mine high-quality training samples for the re-ID CNNs from the test sequence. At the higher level, the similarity measures generated by the adapted re-ID CNNs provide a robust clustering of local tracks. We call this two-pass tracking scheme *Hierarchical Correlation Clustering (HCC).* The HCC framework, and the adaptation of the re-ID net, operationalize the customization: The HCC produces local tracks, from which the training examples for adapting the re-ID net are mined. The adapted re-ID net then generates much more accurate similarity measures, which help the robust association of local tracks and result in long-lived, persistent final tracks.

Contributions. We make the following contributions: *(i)* we propose an effective adaptation approach to automatically customize a multi-person tracker to previously unseen crowded scenes. The customized tracker adapts itself to the uniqueness of people's appearance in a test sequence; *(ii)* we revisit the idea of formulating tracking as a correlation clustering problem [30, 31]. To facilitate the reliable adaptation, we use a hierarchical tracking approach. We use different graph construction schemes at different levels, yet the optimization problem remains the same; *(iii)* we perform extensive experiments on the adaptation scheme and improve the state-of-the-art for the challenging MOT16 benchmark [22]. HCC is the top-performing method at the time of submission.

1.1 Related Work

Tracking-by-detection. Many multi-person tracking methods build on top of tracking-by-detection [3, 27, 31, 34, 38]. Zhang et al. [38] formulate the data association problem of multi-target tracking as a min-cost flow problem, where the optimal solution is obtained in polynomial time. Pirsiavash et al. [23] also proposes a network flow formulation, which can be solved by a successive shortest path algorithm. Wang et al. [34] extend the network flow formulation to simultaneously track interacting objects. Conceptually different from the network flow based formulation, the tracking task is modeled as a correlation clustering problem in [26, 30, 31] and in this work, where the detections are jointly associated within and across frames.

Hierarchical data association Modeling tracking as a hierarchical data association problem has been proposed in many works [6, 26, 30, 35]. In general, detections are associated from neighboring frames to build a tracklet representation and then longer tracks are merged from the tracklets. In [35], the authors propose a two-stage tracking method. For the first stage, they use bipartite-graph matching to aggregate the local information to obtain local tracklets. For the second stage, the association of the tracklets is formulated as a set cover problem. Hierarchical data association has also been employed in [6], whereby the tracklets are generated by greedy searching using an aggregated local flow descriptor as the pairwise affinity measure. Our work differs in the way the local and the global associations are formulated, namely as one and the same optimization problem, thus substantially simplifying the overall tracking framework.

Learning the appearance model. Dehghan et al. [8] track multiple objects via online learning, solving the detection and data association tasks simultaneously. Several works explicitly model the appearance of targets. Kim et al. [13] introduce an appearance model that combines long-term information and employs the features generated by a generic DNN. Xian et al. [36] formulate multiple people tracking as a Markov decision process, the policies of which are estimated from the training data. Their appearance models follow the temporal evolution of the targets. Leal-Taixé et al. [17] introduce several CNNs to model the similarity between detection pairs.

One closely related work [15] learns discriminative appearance models using simple physical constraints. They use hand-crafted features to model appearance and measure the similarity with the χ_2 distance. Our work differs in two respects: first, we focus on adapting a generic ConvNet to test sequences, which is challenging since ConvNets can easily diverge given a few noisy training examples; second, [15] utilizes the tracking framework proposed by [11], where the low-level, middle-level and high-level data association are solved by a two-threshold strategy, a Hungarian algorithm and an EM algorithm respectively, whereas our local and global associations employ the same optimization method, again substantially simplifying the overall tracking framework. A similar approach is proposed in [5]

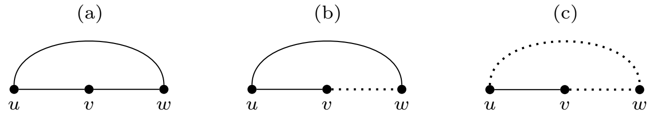


Fig. 1. Examples of feasible/infeasible solutions. The original graph is depicted in (a), where $V = u, v, w$ and $E = uv, vw, uw$. In (b) and (c), a solid line means joining the nodes, a dotted line indicates that the edge is a cut. The decomposition in (b) is infeasible, whereas that in (c) is valid, belonging to the feasible set.

for human pose estimation in a video sequence, where a generic ConvNet pose estimator is adapted to a person-specific pose estimator by propagating labels using dense optical flow and image-based matching. Unlike us, they focus on single-person tracking.

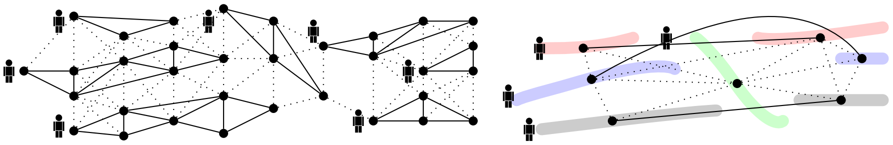
Correlation clustering for tracking. Correlation clustering [9] based tracking formulations have been proposed in [26, 30–32]. Their main advantage is that the detections within and across frames can be jointly clustered, resulting in a robust handling of noisy detections. The use of attractive and repulsive pairwise edge potentials enables a uniform prior distribution for the number of persons, which is determined by the solution of the problem. We extend this idea by using different graph connectivities. Our hierarchical correlation clustering approach, combined with the adaptation scheme, yields better tracking performance.

2 Tracking by Correlation Clustering

Here we introduce the general correlation clustering based tracking formulation that follows [30] and [31]. We describe the model parameters, feasible set and objective function to provide a basic understanding of the formulation and we do not claim novelty with respect to the formulation.

For a target video, let $G = (V, E)$ be an undirected graph whose nodes V are the human detections in a batch of frames or even an entire video. The edges E connect pairs of detections that hypothetically indicate the same target. With respect to the graph, the output of our tracking algorithm is a partition of G, where the node set V is partitioned into different connected components, and each connected component corresponds to one target. The edges $E' \subseteq E$ that straddle distinct components are the cuts of the graph. We define a binary variable x_e for each edge in the graph, where $x_e = 1$ indicates that the edge e is a cut and 0 otherwise. Obtaining the partition of G is equivalent to finding the 01-vector $x \in \{0, 1\}^E$ on the edge set.

Feasible solution. Not all the 01 labelings of the edges lead to valid graph decompositions. As shown in Fig. 1 (b), x_{uv} and x_{uw} join the nodes u, v, w, indicating that all three are in the same cluster. Yet, x_{vw} is a cut, implying v and w should be in different clusters. To avoid such inconsistent edge labeling, we introduce the cycle constraints defined in Eq. 1: for any cycles in the graph



(a) From detections to local tracks by local correlation clustering

(b) From local tracks to global tracks by global correlation clustering

Fig. 2. Illustration of the local and global graph. A dotted line indicates that the edge is a cut. For simplicity, not all the edges are drawn. In (a), the detection graph is partitioned into 7 components, indicating 7 people. In (b), the tracks generated by (a) are associated, resulting in 4 persons. The tracks belonging to the same person are depicted in the same color. (Best viewed in color)

G, if one of its edges is labeled as a cut ($x_e = 1$), then there must be at least one more cut in the cycle.

$$\begin{aligned} &\forall C \in \text{cycles}(G)\, \forall e \in C : \\ &x_e \leq \sum_{e' \in C \setminus \{e\}} x_{e'} . \end{aligned} \tag{1}$$

Objective function. Based on image observations, we compute pairwise features for every edge in the graph. The edge features are denoted as f. We assume independence of the feature vectors. Given the features f and the feasible set Z, the conditional probability of the labeling vector x of the edges is given by

$$p(x|f, Z) \propto p(Z|x) \prod_{e \in E} p(x_e|f^e) \tag{2}$$

where $p(Z|x)$ gives positive and constant probability to every feasible solution ($x \in Z$) and 0 to all infeasible solutions. Minimizing the negative log-likelihood of Eq. 2, we obtain our objective function:

$$\min_{x \in \{0,1\}^E} \quad \sum_{e \in E} c_e x_e \tag{3}$$

with the costs c_e defined as $\log \frac{1-p_e}{p_e}$ and p_e defined as $\frac{1}{1+\exp(-\langle \theta, f^e \rangle)}$.

Given the features f^e extracted from a training sequence, the model parameter θ is obtained by maximum likelihood estimation. To solve the instances of Eq. 3, we apply a heuristic approach proposed in [12].

3 Customized Multi-person Tracker

Here we present how to customize a multi-person tracker. Operationalizing the customization requires two main components: a hierarchical tracking framework, where we obtain reliable tracklets as well as the final tracks (Sec. 3.1) and an adaptation scheme where we fine-tune a general re-ID net to a sequence-specific model (Sec. 3.2).

3.1 Hierarchical Correlation Clustering

The tracking framework introduced in Sec. 2 is general and allows different kinds of graph connectivity. One case is to only connect the detections that are in the neighboring frames. As the detections are close in time, the similarity measures that are based on the local image patch matching are robust [31]. One can therefore obtain high-quality reliable tracks by the tracking formulation introduced in Sec. 2. The downside is that the tracks break when there are occlusions and/or missing detections.

Another way of constructing the graph is to build a fully-connected graph over the entire sequence. Then the detections before and after occlusions can be clustered for the same person, or separated otherwise. By introducing recent, advanced techniques for re-ID, it becomes possible to compute reasonable similarity measures between the detections that are far apart in time, despite large changes in appearance. However, as shown in our experiments, the domain gap between the training sequences and the test sequence is substantial, meanwhile training data for multi-person tracking is hard to obtain, therefore the direct application of a state-of-the-art person re-ID model does not yield satisfactory results. Moreover, since the feasible set of such a graph is huge, a few mistakes in the similarity measures could lead to a bad decomposition, resulting in dramatic tracking errors. For example, similar looking, but temporally distant, people could be clustered together.

To utilize the advantages of different graph connectivity strategies and operationalize the tracker customization, we decouple the tracking problem into two sub-problems: local data association and global data association. For detections that are temporally close, we employ the robust similarity measure of [31]. We construct the graph in a way that there are edges between detections that are sufficiently close in time, as shown in Fig. 2 (a). By constraining the feasible set of the optimization problem, we obtain reliable tracks of people before and after occlusions. In the experiments, we show that our local tracks already achieve reasonable tracking performance.

Furthermore, given reliable tracklets, we employ intuitive physical constraints to mine positive and negative examples to adapt a generic re-ID net to the test sequence. Without any ground truth information, the adapted re-ID net produces significantly better similarity measures for the test sequence. Such similarity measures facilitate a globally connected graph based on the local tracks (Fig. 2 (b)), which enables the re-ID even between long-term occlusions.

Local Clustering. We introduce a graph $G^{local} = (V, E^{local})$, where the edges E^{local} connect detections that are in the same image or that are close in time. We apply the image patch matching based edge potential proposed in [31]. More specifically, for a pair of images and every detection therein, we employ Deep Matching [24] to generate a set of locally matched keypoints, denoted as M_i, where i is the detection index. Then for pairs of detections, we compute the intersection of the matched points as $\mathrm{M}^I_{ij} = |\mathrm{M}_i \cap \mathrm{M}_j|$ and the union as $\mathrm{M}^U_{ij} = |\mathrm{M}_i \cup \mathrm{M}_j|$. Based on M^I_{ij}, M^U_{ij}, and the detection confidences C_i and C_j, we define the pairwise feature f as $(\mathrm{IOU}_{\mathrm{M}_{ij}}, \min_C, \mathrm{IOU}_{\mathrm{M}_{ij}} \cdot \min_C, {\mathrm{IOU}_{\mathrm{M}_{ij}}}^2, {\min_C}^2)^\top$, where $\min_C$ is the minimum detection confidence between C_i and C_j, and $\mathrm{IOU}_{\mathrm{M}_{ij}}$ is the intersection over union of the matched keypoints, denoted as $\mathrm{IOU}_{\mathrm{M}_{ij}} = \frac{\mathrm{M}_{ij}^I}{\mathrm{M}_{ij}^U}$.

Given the pairwise features f for the training sequences, we estimate the model parameter θ via maximum likelihood estimation. On the test sequences, we compute the cost c_e on the edges (E^{local}) using the corresponding features and the learned parameter θ. By optimizing the objective function (Eq. 3) which is defined on the graph G^{local}, we obtain a decomposition of G^{local}, in other words, the clusters of detections. We obtain our local tracks by estimating a smooth trajectory using the corresponding detections. Implementation details are presented in the experiments.

Global Clustering. We return to the global clustering step. In order to bridge local tracks split by long-term occlusions, we construct a fully connected graph $G^{global} = (V, E^{global})$. Its node set contains all the local tracks generated from the previous step. Computing reliable pairwise probabilities p_e is key to global clustering. To that end, we employ the person re-ID net to decide whether two local tracks show the same person or not. Furthermore, we propose an adaptation scheme to fine-tune the generic re-ID net to the test sequence without any ground truth labels. We present the details of our re-ID net and the adaptation scheme in the next section. Extensive experiments on learning to re-identify persons in a test sequence are given in Sec. 4

3.2 Adapting a Generic Re-ID Net

We begin by describing the architecture of the re-ID net and then introduce the approach to adapt a generic Re-ID net on a test sequence. The adaptation pipeline is actually divided into three stages: The re-ID net is first trained on the large re-ID dataset Market-1501 [39] with the initial weights that are trained on ImageNet. Then the model is fine tuned on the training sequences of the MOT15 [16] and MOT16 benchmarks [22], which are arguably the two largest tracking datasets exhibiting diverse scenes. The last stage is the re-ID net adaptation, which happens during testing time by finetuning the model parameters with the mined pairwise examples from the test sequence.

Such adaptation could help the re-ID net in two ways: First, it helps to adapt the model parameters to the test scene, which might be captured under significantly different imaging conditions compared to the training sequences. Second, the re-ID net has the opportunity to "see" the people in the test sequence and learn about what makes people look the same or different. The adapted re-ID net shows a significant improvement both for re-ID and tracking, cf. Sec. 4.

Network Architecture. For the person re-ID module, we adopt a Siamese CNN architecture [4], which has been widely used for person/face verification [7, 37]. Fig. 3 shows the architecture of our re-ID net. The convolutional neural network (CNN) is used to extract a 1024-D feature vector x of the person image in each

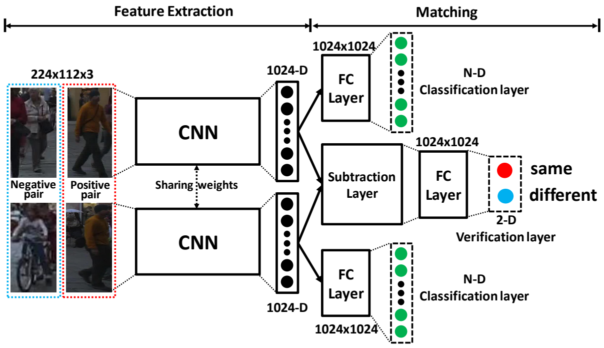


Fig. 3. Re-ID network architecture. The image pairs are passed into two weights sharing CNN branches to extract the 1024-D feature vectors followed by the matching part consisting of a subtraction layer, a 1024-D fully connected layer and a 2-D verification layer. During offline training, the classification layers are added to the re-ID net to use the identification info for network optimization.

branch. Then, the feature vector pair (x_1, x_2) is passed into an element-wise subtraction layer $f(x_1, x_2)$ to obtain the difference vector d. We test 3 different types of non-linearities in the subtraction layer: rectified linear unit (ReLU) $f(x_1, x_2) = \max(x_1 - x_2, 0)$, absolute value non-linearity $f(x_1, x_2) = |x_1 - x_2|$ and square value non-linearity $f(x_1, x_2) = (x_1 - x_2)^2$. The ReLU operation performs the best during the offline finetuning stage, and also costs less computation than square value non-linearity. Then, the difference vector d is passed to a 1024-D fully-connected layer followed by a 2-D output layer with a softmax operation. Inspired by the joint face verification-classification in [28], we add a 1024-D fully-connected layer followed by a N-D classification layer with a softmax operation to classify the identity of each training image. In our experiments, we validate that incorporating the identification information can significantly improve the verification performance.

We use a weight-sharing GoogLeNet [29] in each CNN branch because of its good performance and fewer parameters than other CNN models. In order to make the CNNs more suitable for the pedestrian images, we modify the input layer size from $224 \times 224 \times 3$ to $224 \times 112 \times 3$ and change the kernel size of the last max-pooling layer from 7×7 to 7×4. Since there are two N-D auxiliary classification subnets in each GoogLeNet, we minimize 7 cross entropy losses including. the losses of 6 classification layers and 1 verification layer.

The classification layers are only used in the network training phase. In the testing phase, the verification layer is used to calculate the similarities between the image pairs. This also applies to the adaptation step, as we only optimize the network with the verification loss, since tracklets without temporal overlap may contain the same identities and we can not obtain identification information about the test sequence. Furthermore, to reduce the computational burden, we divide the network into two parts: feature extraction and matching (Fig. 3). First,

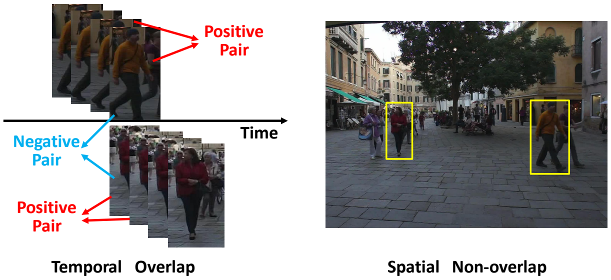


Fig. 4. Training examples mined from test sequence. The two tracklets overlap temporally but not spatially, hence show two identities.

we feed-forward each image through one CNN branch and store its feature vector. Then, the similarity of each feature vector pair is calculated with the matching part, including the subtraction layer, fully connected layer, and verification layer.

Generic Re-ID Net Training. In order to train the re-ID net more efficiently, the weights trained on ImageNet are used as initialization. Then, we finetune the re-ID net with the training data of the large re-ID dataset Market-1501 [39], which contains 1501 identities captured from 6 cameras. Accordingly, the number of the nodes in the classification layer (Fig. 3) is set to N=1501.

The size of the training data from the tracking benchmarks is often quite small, due to the expensive ground truth annotation. We use the training sequences from the MOT15 and MOT16 benchmark to collect training images, except that the MOT16-02 and MOT16-11 sequences from the MOT16 training set are left for collecting test examples. We have in total 478 identities for finetuning our re-ID model which is pre-trained on the Market-1501 dataset.

Adaptation on Test Sequences. Now we turn to the adaptation step, where we mine training examples from the target video to finetune our re-ID net. As mentioned in Sec. 3, the local clustering step of the tracking approach produces local tracks of people when they are likely to be fully visible. These local tracks are like spatio-temporal "curves" that can start and stop at any frame, depending on the detections and the scene. An illustration of the local tracks is shown in Fig. 2 (b). The spatio-temporal locations of the local tracks enforce intuitive constraints: if two tracks pass through the same frame but do not overlap spatially, they then most probably correspond to different people in the video. An example is shown in Fig. 4. The positive pairs are the detection images that come from the same track, and the negative pairs are from the two tracks that have temporal overlap but no spatial overlap.

To avoid including too many noisy training examples during the finetuning, we discard 20% of the head and tail images of local tracks, as the starting and/or the ending part of the tracks sometimes have inaccurate detection bounding boxes. Furthermore, we compute the average detection score of the detections within a local track. Based on the average score, we learn the probability of the local track to be a true positive on the MOT16 training sequences. During the training example mining, we exclude the local tracks that are more likely to be on the background of the scene by only considering the tracks whose probabilities of being true positives are larger than 0.5. Two tracks without temporal overlap may contain the same identity. We have no information about the identity of the people in the video. Therefore, during the finetuning stage, we only use the verification information, in other words, the classification subnet is not used.

The adapted re-ID net produces the cut/join probabilities between each pair of local tracks. Then we use the probabilities to compute the costs of the objective function (Eq. 3). Similar to the local clustering step, we solve the instances of the optimization problem by the heuristic approach proposed in [12].

4 Experiments

In this section, we present our experiments with the proposed re-ID net (Sec. 4.1), the adaptation approach (Sec. 4.2) and multi-person tracking (Sec. 4.3).

4.1 Re-ID Net Architecture Evaluation

Training/test data collection. On the Market-1501 dataset, we train the re-ID model using its standard training set. In order to evaluate the performance of the re-ID net on the tracking data, we collect training and test examples from the MOT15 and MOT16 training set, which contain 576 identities in total. We randomly select 80% identities (460) from each sequence as the training set and the rest 20% (116) as the test set. The MOT benchmark also provides person detections on every frame. The detections are considered as true positives for a certain identity if their intersection-over-union (IOU) with the ground truth of the identity are larger than 0.5. Then the positive (negative) examples are pairs of detections that are assigned to the same (different) identities.

Metric. The metric used in the following person re-identification experiments is the verification accuracy.

Re-ID Net Implementation Details. On Market-1501, we train the re-ID net for 80k iterations using the initial weights trained on ImageNet. Adam [14] is used as the optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The initial learning rate is set to 1e-4 and the minibatch size is 128. We use random left-right flips for data augmentation.

Verification-Net and Classification-Net. In order to evaluate the effectiveness of the proposed classification subnet of our re-ID net, we compare two network architectures: using only the verification layer (V-Net) and using both verification and classification layers (V+C-Net). In Tab. 1, the first two rows show the verification accuracy of the two re-ID nets trained on Market-1501. The mean accuracy of V+C-Net is 80.7% which is significantly higher than the result

(68.8%) for V-Net. We further finetune both re-ID networks on the MOT training data. In this case, the training and test data are quite similar as they are both from the MOT benchmark. Nevertheless, the V+C-net still outperforms V-net by 3.1%. These experiments suggest that training the classification task and the verification task jointly improves the verification accuracy. Therefore we use the V+C-Net in the following experiments.

Table 1. MOT16 Re-ID Accuracy (%)

Model	Training	PosAcc.	NegAcc.	Mean
V-Net	Market-1501	65.5	72.0	68.8
V+C-Net	Market-1501	90.4	70.9	80.7
V-Net	Market-1501/MOT	100.0	90.2	95.1
V+C-Net	Market-1501/MOT	98.0	98.4	**98.2**

4.2 Adaptation Evaluation

Setup. To evaluate the adaptation scheme of the re-ID net in the context of multi-person tracking, we need to finetune the network on proper tracking test sequences. Here, we choose the MOT16-02 and MOT16-11 sequences as the test set. For the offline trained generic re-ID nets, we provide three different versions: ImageNet model, the one fine tuned on Market-1501, the one further fine tuned on the MOT training set (excluding MOT16-02 and MOT16-11)

For adapting the re-ID net on the test sequence, we employ two kinds of training strategis: only finetuning the fully-connected layer (AdaptationFC), and finetuning the whole network end-to-end (AdaptationE2E). The training examples are mined using the tracks generated by the local correlation clustering (HCC^l). All the adapted models are obtained with 3 epochs to avoid overfitting.

Results. As shown in Tab. 2, of the two generic re-ID models, the one further fine tuned on the MOT data improves the mean accuracy on MOT16-02 from 78.5% to 84.8%. For the adaptation schemes, the end-to-end model (AdaptationE2E) significantly improves the performance on MOT16-02 from 84.8% to 88.3%, where the AdaptationFC decreases the accuracy to 82.9%. These results suggest that it is important to adapt the CNN features to the test scene. We further perform the adaptation on the models that are trained on the Market-1501 and ImageNet. The results are significant: without finetuning the model on the tracking sequence (MOT training set), the adaptation method already produces 88.0% accuracy; even the model trained on ImageNet, which has never seen person re-ID data before, already produces reasonable accuracy.

The same tendency is observed on the MOT16-11 sequence. Adapting the model that is further trained on the MOT set performs the best (93.6%), and is slightly better than the generic model that is trained on Market-1501 (93.3%).

Table 2. Analysis on the adaptation approach

		MOT16-02			MOT16-11		
Model	OfflineTraining	PosAcc.	NegAcc.	Mean	PosAcc.	NegAcc.	Mean
Generic Net	Market-1501	82.6	73.4	78.0	82.7	81.3	82.0
Generic Net	Market-1501/MOT	83.7	85.9	84.8	85.8	90.6	88.2
AdaptationFC	Market-1501/MOT	76.4	89.4	82.9	88.2	88.0	88.1
AdaptationE2E	Market-1501/MOT	83.1	93.4	**88.3**	91.8	95.4	**93.6**
AdaptationE2E	Market-1501	82.6	93.4	88.0	91.7	94.9	93.3
AdaptationE2E	ImageNet	78.4	87.3	82.9	78.9	85.7	82.3

These results clearly validate the effectiveness of the proposed adaptation approach. Even without the annotated tracking data, the re-ID net that is adapted on the test sequence produces a good verification accuracy.

4.3 Tracking Experiments

We perform comparisons with recent tracking work on the challenging MOT16 Benchmark [22]. The benchmark contains a training and a test set, each with 7 sequences that are captured with different camera motion, and different imaging and scene conditions. For the test sequences, the training sequences that are captured under the same framerate and camera motion (moving/static) are provided. The model parameter θ for local clustering are learned from the training sequences that have the same framerate and camera motion (moving/static) via maximum likelihood estimation. To validate the effectiveness of different components of the proposed method, we select MOT16-02 and MOT16-11 as validation set to perform our analysis, in line with the previous section.

Tracking Implementation Details. As the detections provided by the benchmark are very noisy, we could also obtain small clusters on the background. In all the tracking experiments presented below, we remove the clusters whose sizes are smaller than 5. Given the detection cluster of a target, we estimate its tracks by using the code from [21] which estimates a spline curve for each target. We also fill in the missing detections when there are gaps in time due to occlusion or detection failures. However, when the gaps are too big, estimating the position and scale of missing detections of the tracks becomes difficult. Therefore, we fill in the missing detections within a certain temporal distance. In our experiments, for the sequences captured by a moving camera, the missing detections are filled in when the temporal gap is less than fps, where fps is the frame-rate of the sequence. For the sequences captured by a static camera, the missing detections are filled when the temporal gap is less than $2 \times fps$. These hyper-parameters are set according to the performance on the validation set.

Evaluation Metric. Following the MOT16 Benchmark, we apply the standard CLEAR MOT metrics [1]. The most informative metric is the multiple object tracking accuracy (MOTA) which is a combination of false positives (FP), false negatives (FN), and identity switches (IDs). Other important metrics are ID F1 Score (IDF1) [25], mostly tracked (MT) tracks, mostly lost (ML) tracks, fragmentation (FM), recall (Rcll) and precision (Prcn).

Comparison of tracking performance with local and global clustering Setup. In this section, we compare the tracking performance of local (HCC^l) and global (HCC^g) correlation clustering. The HCC^l should generate reliable tracks that are robust to detection noise and abrupt camera motion. Once the target is fully occluded or missed by the detector within a short temporal window, the tracks will be terminated and a new track will start when the target is visible again. Given the local tracks, the global graph is constructed in such a way that all the local tracks are connected to each other, to enable the re-ID of the target within a much longer temporal window, even the whole sequence.
Results. It can be seen from Tab. 3 that on the MOT16-02 sequence, HCC^l achieves a MOTA of 19.5%. With the global clustering step, the MOTA is increased to 20.3% with generic re-ID model, and is further improved to 21.3% with the adapted re-ID model. Intuitively, the HCC^l could produce more ID switches and false negatives, because the underlying graph is constructed in the way that only the detections close in time are connected. With a well trained re-ID net and a reasonable filling in strategy, the HCC^g should reduce the number of ID switches and false negatives. Analyzing the data corroborates our hypotheses: the number of ID switches goes from 62 (HCC^l) to 33 (HCC^g), suggesting the effectiveness of the global correlation clustering step. Similar observation can be made on the MOT16-11 sequence. The overall MOTA increases from 53.3% to 55.1%. The number of ID switches decreases from 24 to 8, indicating that the majority of previously interrupted tracks are re-linked by the global clustering.

Comparison of tracking performance with the generic and adapted re-ID net Setup. In this section, we compare the tracking performance of the generic re-ID model ($HCC^g_{generic}$) and the adapted re-ID model ($HCC^g_{adaptation}$). For both of them, the local tracks are identical. For $HCC^g_{generic}$, the similarities between the local tracks are computed with the generic re-ID net; for $HCC^g_{adaptation}$, the similarities between the local tracks are computed with the adapted one. As shown in Tab. 2, the accuracy of the similarity measure indicates the superior performance of the adapted model. It stands to reason that such superiority should be translated into a better performance of the tracking task.
Results. It can be seen from Tab. 3 that for both the MOT16-02 and MOT16-11 sequences, the adapted model produces a better MOTA (20.3% to 21.3%, 54.2% to 55.1% respectively). Besides, with the adapted model, we obtain better MOTP and Prcn, which suggests that the overall tracking performance is superior.

Results on the MOT16 Benchmark. To compare with the state-of-the-art multi-target tracking models, we evaluate our tracking model on the MOT16 test set. The evaluation is performed according to the benchmark and the results

Table 3. Comparison of tracking performance: 1. local clustering vs global clustering; 2. generic vs. adapted re-ID net

Method	MOT16-02				MOT16-11			
	FP	FN	IDs	MOTA	FP	FN	IDs	MOTA
HCC^{l}	507	13784	62	19.5	408	3861	24	53.2
$HCC^{g}_{generic}$	1045	13120	43	20.3	429	3763	7	54.2
$HCC^{g}_{adaptation}$	867	13131	33	21.3	352	3762	8	55.1

Table 4. Comparison on the MOT16 test set. Best in bold, second best in blue

Method	MOTA	IDF1	MT	ML	FP	FN	IDs	Frag	Hz	Detector
MOTDT [19]	47.6	50.9	15.2%	38.3%	9253	**85431**	792	1858	**20.6**	Public
NLLMPa [18]	47.6	47.3	17.0%	40.4%	5844	89093	629	768	8.3	Public
FWT [10]	47.8	44.3	**19.1%**	**38.2%**	8886	85487	852	1534	0.6	Public
GCRA [20]	48.2	48.6	12.9%	41.1%	**5104**	88586	821	1117	2.8	Public
LMP [32]	48.8	**51.3**	18.2%	40.1%	6654	86245	481	595	0.5	Public
HCC	**49.3**	50.7	17.8%	39.9%	5333	86795	**391**	**535**	0.8	Public

are publicly available [5]. In Tab. 4, we compare with all the published works. Generally speaking, for the 9 metrics that are considered by the benchmark, our model achieves the best performance on MOTA, IDs, Frag, and the second best performance on FP. Such results suggest the advantages and effectiveness of the proposed tracking approach.

5 Conclusion

In this paper, we address the challenging problem of tracking multiple people in crowded scenes, where long-term occlusion is arguably the major challenge. To that end, we propose an adaptation scheme that explores the modeling capability of deep neural networks, by mining training examples from the target sequence. The adapted neural network produces reliable similarity measures, which facilitate person re-ID after long-term occlusion. Furthermore, we utilize an overall rigorous formulation [30,31] to hierarchically link and associate people. The combination of the tracking formulation and the adaptation scheme results in an effective multi-person tracking approach that demonstrates a new state-of-the-art.

Acknowledgments This research was supported in part by Toyota Motors Europe and the German Research Foundation (DFG CRC 1233). **Disclosure** MJB has received research gift funds from Intel, Nvidia, Adobe, Facebook, and Amazon. While MJB is a part-time employee of Amazon, his research was performed solely at, and funded solely by, MPI.

[5] https://motchallenge.net/results/MOT16/

References

1. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: The CLEAR MOT metrics. Image and Video Processing **2008**(1), 1–10 (2008) 12
2. Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., Van Gool, L.: Online multiperson tracking-by-detection from a single, uncalibrated camera. TPAMI **33**(9), 1820–1833 (2011) 1
3. Brendel, W., Amer, M., Todorovic, S.: Multiobject tracking as maximum weight independent set. In: CVPR. pp. 1273–1280. IEEE (2011) 3
4. Bromley, J., Bentz, J.W., Bottou, L., Guyon, I., LeCun, Y., Moore, C., Säckinger, E., Shah, R.: Signature verification using a "siamese" time delay neural network. IJPRAI **7**(4), 669–688 (1993) 7
5. Charles, J., Pfister, T., Magee, D., Hogg, D., Zisserman, A.: Personalizing human video pose estimation. In: CVPR. pp. 3063–3072. IEEE, Las Vegas (2016) 3
6. Choi, W.: Near-online multi-target tracking with aggregated local flow descriptor. In: ICCV. pp. 3029–3037. IEEE, Santiago (2015) 3
7. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: CVPR. pp. 539–546. IEEE, San Diego (2005) 7
8. Dehghan, A., Tian, Y., Torr, P.H.S., Shah, M.: Target identity-aware network flow for online multiple target tracking. In: CVPR. pp. 1146–1154. IEEE, Boston (2015) 3
9. Grötschel, M., Wakabayashi, Y.: A cutting plane algorithm for a clustering problem. Mathematical Programming **45**(1), 59–96 (1989) 4
10. Henschel, R., Leal-Taixé, L., Cremers, D., Rosenhahn, B.: Fusion of head and full-body detectors for multi-object tracking. In: CVPRW. pp. 1541–1550. IEEE, Salt Lake City (2018) 14
11. Huang, C., Wu, B., Nevatia, R.: Robust object tracking by hierarchical association of detection responses. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV. pp. 788–801. Springer, Marseille (2008) 3
12. Keuper, M., Levinkov, E., Bonneel, N., Lavoue, G., Brox, T., Andres, B.: Efficient decomposition of image and mesh graphs by lifted multicuts. In: ICCV. pp. 1751–1759. Santiago (2015) 5, 10
13. Kim, C., Li, F., Ciptadi, A., Rehg, J.M.: Multiple hypothesis tracking revisited. In: ICCV. pp. 4696–4704. IEEE, Santiago (2015) 2, 3
14. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. In: ICLR. pp. 13–23. San Diego (2015) 10
15. Kuo, C.H., Huang, C., Nevatia, R.: Multi-target tracking by on-line learned discriminative appearance models. In: CVPR. pp. 685–692. IEEE, San Francisco (2010) 3
16. Leal-Taixé, L., Milan, A., Reid, I., Roth, S., Schindler, K.: MOTChallenge 2015: Towards a benchmark for multi-target tracking. arXiv:1504.01942 7
17. Leal-Taixé, L., Canton-Ferrer, C., Schindler, K.: Learning by tracking: Siamese CNN for robust target association. In: CVPRW. pp. 418–425. IEEE, Las Vegas (2016) 2, 3
18. Levinkov, E., Uhrig, J., Tang, S., Omran, M., Insafutdinov, E., Kirillov, A., Rother, C., Brox, T., Schiele, B., Andres, B.: Joint graph decomposition & node labeling: Problem, algorithms, applications. In: CVPR. pp. 1904–1912. IEEE, Honolulu (2017) 14

19. Long, C., Haizhou, A., Zijie, Z., Chong, S.: Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In: ICME. pp. 1–6. IEEE, San Diego (2018) 14
20. Ma, C., Yang, C., Yang, F., Zhuang, Y., Zhang, Z., Jia, H., Xie, X.: Trajectory factory: Tracklet cleaving and re-connection by deep siamese bi-gru for multiple object tracking. In: ICME. pp. 1–6. IEEE, San Diego (2018) 14
21. Milan, A., Roth, S., Schindler, K.: Continuous energy minimization for multitarget tracking. TPAMI **36**(1), 58–72 (2014) 1, 12
22. Milan, A., Leal-Taixé, L., Reid, I.D., Roth, S., Schindler, K.: MOT16: A benchmark for multi-object tracking. arXiv:1603.00831 (2016) 3, 7, 12
23. Pirsiavash, H., Ramanan, D., Fowlkes, C.C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: CVPR. pp. 1201–1208. IEEE, Colorado Springs (2011) 1, 3
24. Revaud, J., Weinzaepfel, P., Harchaoui, Z., Schmid, C.: Deepmatching: Hierarchical deformable dense matching. IJCV **120**(3), 300–323 (2016) 6
25. Ristani, E., Solera, F., Zou, R.S., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: ECCVW. pp. 17–35. Springer, Amsterdam (2016) 13
26. Ristani, E., Tomasi, C.: Tracking multiple people online and in real time. In: ACCV. pp. 444–459. Springer, Singapore (2014) 2, 3, 4
27. Shitrit, H.B., Berclaz, J., Fleuret, F., Fua, P.: Tracking multiple people under global appearance constraints. In: ICCV. pp. 137–144. IEEE, Spain (2011) 3
28. Sun, Y., Chen, Y., Wang, X., Tang, X.: Deep learning face representation by joint identification-verification. In: NIPS. pp. 1988–1996. Montreal (2014) 8
29. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR. pp. 1–9. IEEE, Boston (2015) 8
30. Tang, S., Andres, B., Andriluka, M., Schiele, B.: Subgraph decomposition for multi-target tracking. In: CVPR. pp. 5033–5041. IEEE, Boston (2015) 2, 3, 4, 14
31. Tang, S., Andres, B., Andriluka, M., Schiele, B.: Multi-person tracking by multicuts and deep matching. In: ECCVW. pp. 100–111. Springer, Amsterdam (2016) 2, 3, 4, 6, 14
32. Tang, S., Andriluka, M., Andres, B., Schiele, B.: Multiple people tracking by lifted multicut and person re-identification. In: CVPR. pp. 3701–3710. IEEE, Honolulu (2017) 4, 14
33. Tang, S., Andriluka, M., Schiele, B.: Detection and tracking of occluded people. IJCV **110**(1), 58–69 (2014) 1
34. Wang, X., Turetken, E., Fleuret, F., Fua, P.: Tracking interacting objects using intertwined flows. TPAMI **38**(11), 2312–2326 (2016) 1, 3
35. Wu, B., Nevatia, R.: Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors. IJCV **75**(2), 247–266 (2007) 3
36. Xiang, Y., Alahi, A., Savarese, S.: Learning to track: Online multi-object tracking by decision making. In: ICCV. pp. 4705–4713. IEEE, Santiago (2015) 3
37. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Deep metric learning for person re-identification. In: ICPR. pp. 34–39. IEEE, Stockholm (2014) 7
38. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: CVPR. IEEE (2008) 3
39. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. In: ICCV. pp. 1116–1124. IEEE, Santiago (2015) 7, 9