is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

Motion Segmentation & Multiple Object Tracking by Correlation Co-Clustering

Margret Keuper[1], Siyu Tang[4],Bjoern Andres[3,5,6], Thomas Brox[2], Bernt Schiele[3]
[1]Data and Web Science Group, University of Mannheim, Germany
[2] Department for Computer Science, University of Freiburg, Germany
[3]Max Planck Institute for Informatics, Saarbruecken, Germany
[4]Max Planck Institute for Intelligent Systems, Tuebingen, Germany
[5]Bosch Center for AI, [6] University of Tuebingen, Germany

Abstract—Models for computer vision are commonly defined either w.r.t. low-level concepts such as pixels that are to be grouped, or w.r.t. high-level concepts such as semantic objects that are to be detected and tracked. Combining bottom-up grouping with top-down detection and tracking, although highly desirable, is a challenging problem. We state this joint problem as a co-clustering problem that is principled and tractable by existing algorithms. We demonstrate the effectiveness of this approach by combining bottom-up motion segmentation by grouping of point trajectories with high-level multiple object tracking by clustering of bounding boxes. We show that solving the joint problem is beneficial at the low-level, in terms of the FBMS59 motion segmentation benchmark, and at the high-level, in terms of the Multiple Object Tracking benchmarks MOT15, MOT16 and the MOT17 challenge, and is state-of-the-art in some metrics.

1 Introduction

Computer vision methods commonly fall into one of two categories. Bottom-up methods are centered around low-level concepts such as pixels that are to be grouped. Top-down methods are centered around high-level concepts such as semantic objects that are to be detected or tracked. These concepts are usually learned from datasets. Combinations of bottom-up and top-down methods are highly desirable, as their advantages are complementary in practice [11], [19], [28], [29], [30].

In this paper, we combine bottom-up motion segmentation with top-down multiple object tracking. Specifically, we combine bottom-up motion segmentation by grouping of point trajectories with top-down multiple object tracking by clustering of bounding boxes. Point trajectories are entities which represent single points over time. Motion segmentation can be achieved as a spatial grouping of point trajectory based on motion cues. Object detections represent sets of points which belong to object instances at one point in time. Object tracking can be achieved by associating detections over time.

Both individual grouping problems have been addressed most successfully by correlation clustering approaches, also referred to as *minimum cost multicuts* [39], [40], [45], [67], [68], [70].

However, point trajectories and bounding boxes form complementary cues to the solution of both problems: Point trajectories, on the one hand, can help to cluster bounding box detections of the same object across partial occlusions, a key challenge of bounding box tracking alone (see Fig. 1). In conventional, purely high-level methods, such occlusions can easily lead to identity switches or lost tracks. However, low-level points on specific, well-structured regions might be easy to track over a long period of time and thus avoid identity switches. If sufficiently many such trajectories can be found on an object of interest, the tracking problem becomes trivial even if the frame-wise object detection fails.

Bounding boxes, on the other hand, can help to group point trajectories in the presence of articulated motion, a key challenge of motion segmentation with point trajectories alone. Ideally, employing such pairwise information between detections may

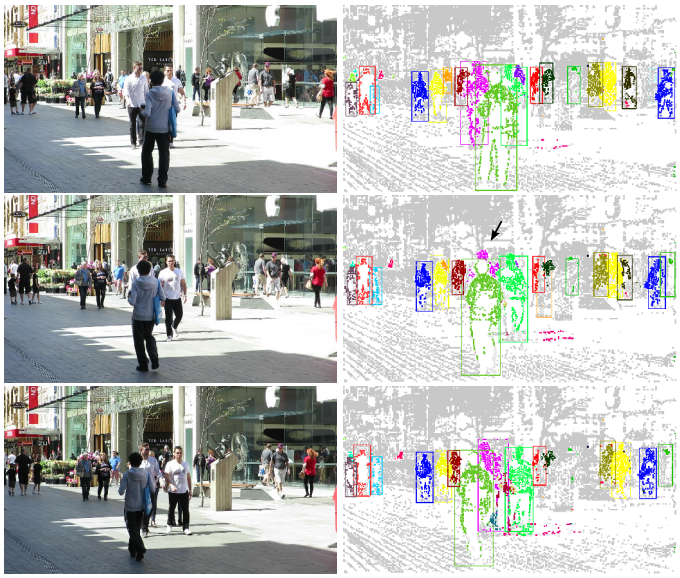


Fig. 1. **Left:** Frames 100, 110, and 120 of the sequence *MOT16-08* [50]. **Right:** Segmentation and tracking result are depicted as color-labeled point trajectories and bounding boxes, respectively. Formulating bottom-up motion segmentation and top-down multiple object tracking as a joint co-clustering problem, combines advantages of both approaches and is tolerant even to strong partial occlusion, indicated by the black arrow. It establishes links between low-level concepts (point trajectories) and high-level concepts (bounding boxes).

M.K. and T.B. acknowledge funding by the ERC Starting Grant VideoLearn. M.K. acknowledges funding by the DFG project KE 2264/1-1

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

replace higher-order terms on trajectories as proposed in [53] or [39]. While it is impossible to tell two rotational or scaling motions apart when only considering pairs of trajectories, pairs of detection bounding boxes contain enough points to distinguish their motion. With sufficiently complex detection models, even articulated motion can be disambiguated.

This motivates the combination of bottom-up motion segmentation by grouping of point trajectories with top-down multiple object tracking by clustering of bounding boxes.

Feature trajectories have been used for multiple object tracking before, for example in [29], [30], [34], [43]. These previous approaches face the challenge to combine possibly contradictive information on the two different levels of granularity. This makes the optimization using, for example, spectral clustering or conditional random fields hard. In contrast to these previous works, we formulate a joint optimization problem that can intrisically handle conflicting information by the means of contraints.We contribute a correlation co-clustering problem whose feasible solutions define

1) a feasible solution w.r.t. the bottom-up motion segmentation problem,
2) a feasible solution w.r.t. the top-down tracking problem, and
3) an association between bottom-up concepts (point trajectories) and top-down concepts (bounding boxes).

This association is depicted in Fig. 1 by colors. The existence of such an association, which we postulate, establishes non-trivial dependencies between the feasible solutions of the bottom-up and top-down problem and, thus, to a consolidation of their respective costs.

This formulation for combining possibly conflicting cues in a clean an flexible way is beneficial at the low-level, as we show in terms of the FBMS59 motion segmentation benchmark [54], where we can report state-of-the-art performance. Particularly strong improvements can be achieved w.r.t. the number of correctly segmented objects. It is equally beneficial at the high-level, as we show in terms of the multiple object tracking benchmarks [44] [50], where it yields state-of-the-art results in some metrics and, in particular, shows the ability to reduce the number of ID switches. It is the winning entry of the MOT17 challenge for multiple object tracking [44], [50], proving that it is easily applicable and results do not dependent on tedious parameter tuning.

2 Related Work

The combination of high-level and low-level cues is an established idea in computer vision research. Its advantages have been demonstrated for image segmentation [11] as well as for motion segmentation in conjunction with tracking [19], [28], [29]. Similar to points trajectories, head detections have been used as additional features for multiple-person tracking for example in [9], [15], [32]. However, our proposed method is substantially different in that we provide a unified graph structure whose partitioning both solves the low level problem, here, the motion segmentation task, and the high-level problem, i.e. the multi target tracking task, at the same time and thus have a dual objective, formulated in a single optimization problem. Closest in spirit to our approach is the approach by Fragkiadaki et al. [30], where detectlets, small tracks of detections, are classified in a graphical model that, at the same time, performs trajectory clustering based on a spectral clustering formulation.

Like our work, Fragkiadaki et al. [30] define a graph whose nodes are point trajectories or (sets of) bounding boxes. Conflicting information on both levels of granularity is handled by a mediation step, i.e., the approach solves a sequence of constrained spectral clustering problems. In contrast, we solve a single correlation clustering problem, where the consolidation between high-level and low-level information is handled intrinsically and directly via constraints. This has clear advantages regarding optimality.

In Milan et al. [49], tracking and video segmentation are also formulated as a joint problem. However, their approach employs conditional random fields instead of correlation clustering, is built upon temporal superpixels [14] instead of point trajectories and strongly relies on unary terms learned on these superpixels.

The correlation clustering problem [6] is also known as the minimum cost multicut or graph partition problem [20]. Despite its APX-hardness [22], it is used as a mathematical abstraction for a variety of computer vision tasks, including image segmentation [1], [38], [41], [42], [79], multiple object tracking [67], [68] and human body pose estimation [36], [60]. Unlike clustering problems with non-negative costs, the correlation clustering problem does not define a constraint or cost on the number or size of clusters. Instead, these properties are defined by the solutions. Practical algorithms for correlation clustering include local search heuristics [7], [8], [41], [45] for finding feasible solutions, as well as cutting plane algorithms [2], [38], [66] and a column generation algorithm [79] for computing lower bounds. We resort to the local search algorithm [41] for which C++ code is publicly available.

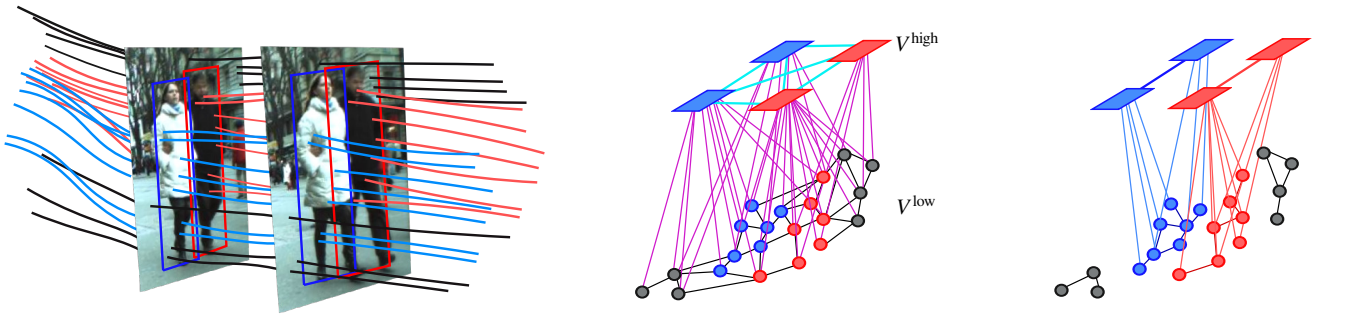
Motion segmentation by grouping of point trajectories is studied in [12], [18], [37], [39], [40], [46], [48], [53], [54], [61], [64]. The approaches of [12], [18], [37], [39], [40], [46], [48], [53], [54], [61], [64] base their segmentations on pairwise affinities while [25], [39], [53], [83] model higher order motions by varying means. In [39], [53] third order terms are employed to explain not only translational motion but also in-plane rotation and scaling. Zografos et al. [83] model even more general 3D motion using group invariants. Elhamifar and Vidal [25] model higher order motion subspaces. The actual grouping in these methods is done using spectral clustering with the exception of Rahmati et al. [61] who employ multi-label graph cuts, Keuper [39] who employ higher-order minimum cost multicuts, and Ji et al. [37] who optimize an unbalanced energy that models the motion segmentation at the same time as the point matching and solve it via the Alternating Direction Method of Multiplier, i.e. they do not rely on any previous method to define point trajectories. Similarly, the approach by Bideau and Learned-Miller [57] works directly on the optical flow between pairs of frames and uses information from the angle field to derive a probabilistic model for object motion.

In Fragkiadaki et al. [29] motion trajectory grouping in a setup similar to [12] is used to perform tracking. Although the grouping in [29] is computed using spectral clustering, repulsive weights can be applied based on the findings of Yu and Shi [80]. Resulsive terms are computed from the segmentation topology. In contrast, we compute both, attractive and repulsive weights, from motion cues and object detections.

In our approach, we build on [40] where the grouping of point trajectories is cast as a correlation clustering problem in terms of pairwise potentials. Algorithms for turning groups of point trajectories into a segmentation on the pixel grid were defined in [51], [52].

Multiple object tracking by linking bounding box detections (*tracking by detection*) was studied, e.g., in [4], [5], [30], [32], [33],

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) High-level bounding boxes and low-level trajectories (b) Proposed graph (c) Feasible solution

Fig. 2. Here, we visualize an exemplary graph G built on a two-frame video sequence showing two walking pedestrians. (a) At a high level, bounding boxes describe feasible detections of humans. At a low level, trajectories describe feasible motions of points. (b) Both are represented here by nodes in a graph. Nodes drawn as rectangles represent bounding boxes, nodes drawn as circles represent point trajectories. (c) An optimal decomposition of the graph defines, firstly, a grouping of point trajectories, secondly, a clustering of bounding boxes, and thirdly, an assignment of point trajectories to bounding boxes.

[33], [35], [58], [71], [81]. Therein, the combinatorial problem of linking detection proposals over time is solved via integer linear programming [65], [73], maximum a posteriori probability (MAP) estimation [58], conditional random fields [43], dominant sets [72], or continuous optimization [5]. To make the optimization in these approaches tractable, non-maximum suppression or pre-grouping of detections into tracklets is very common [4], [30], [33], [35], [71], [76], [77], [81]. Andriluka et al. [4] use a hidden Markov model (HMM) to build tracklets that cover the detections during a small number of frames. Huang et al. [35] propose to use the Hungarian algorithm in a three-level hierarchical association framework to gradually increase the length of the tracklets. Zamir et al. [81] use generalized minimum clique graphs to model the data association problem both for the tracklet generation and the final trajectory generation. Non-maximum suppression is also a crucial component in disjoint path formulations, such as [15], [59], [74]. [15] propose a pairwise overlap cost in their objective function to avoid multiple objects occupying the same spatial location. Similarly [74] propose spatial exclusion constraints to prevent overlapping cuboids in the 3D space.

We build on the prior work from Tang et al. [67], [68], where the combination of bounding boxes is cast as a correlation clustering problem.

3 Correlation Co-Clustering

3.1 Optimization Problem

In this section, we state the low-level grouping of point trajectories and the high-level clustering of bounding boxes in the form of a single correlation co-clustering problem. In this, we build on [41] which states the low-level problem as a correlation clustering problem, and on [67] which states the high-level problem as a correlation clustering problem. Our joint co-clustering problem differs from [41], [67] in that it introduces dependencies between the two sub-problems.

At the low level, we define a graph $G^{\text{low}} = (V^{\text{low}}, E^{\text{low}})$ whose nodes are point trajectories and whose edges connect point trajectories that potentially belong to the same group. Such edges are depicted in Fig. 2b in black. At the high level, we define a graph $G^{\text{high}} = (V^{\text{high}}, E^{\text{high}})$ whose nodes are bounding boxes and whose edges connect bounding boxes that potentially belong to the same object. Such edges are depicted in Fig. 2b in cyan. Between these levels, we define a set E^{lh} of additional edges $\{u, v\} \in E^{lh}$ that connect a low-level point trajectory $u \in V^{\text{low}}$ with a high-level bounding box $v \in V^{\text{high}}$, indicating that both potentially belong to the same object. Such edges are depicted in Fig. 2b in magenta.

For the entire graph $G = (V, E)$ with $V := V^{\text{low}} \cup V^{\text{high}}$ and $E := E^{\text{low}} \cup E^{\text{high}} \cup E^{lh}$ and for any edge $\{u, w\} \in E$, we define a cost $c_{uv} \in \mathbb{R}$ that is positive, i.e. attractive, if u and v are likely to belong to the same object and negative, i.e. repulsive, if v and w are unlikely to belong to the same object. The estimation of these costs from image data is described in detail below.

Also for every edge $\{u, v\} \in E$, we introduce a binary variable $y_{uv} \in \{0, 1\}$ that indicates by $y_{uv} = 0$ that u and v belong to the same object and by $y_{uv} = 1$ that u and v belong to distinct objects. In order to ensure that the 01-labeling $y \in \{0, 1\}^E$ of all edges is consistent and well-defines a decomposition of the graph G into clusters, we impose on y the well-known cycle constraints (2) [20]. Overall, we consider the correlation co-clustering problem (1)–(2)

$$\min_{y \in \{0,1\}^E} \sum_{e^{\text{high}} \in E^{\text{high}}} c_{e^{\text{high}}} y_{e^{\text{high}}} + \sum_{e^{\text{low}} \in E^{\text{low}}} c_{e^{\text{low}}} y_{e^{\text{low}}} + \sum_{e^{\text{hl}} \in E^{\text{lh}}} c_{e^{\text{lh}}} y_{e^{\text{lh}}} \quad (1)$$

$$\text{subject to } \forall C \in \text{cycles}(G)\, \forall e \in C : y_e \leq \sum_{f \in C \setminus \{e\}} y_f \quad (2)$$

Specifically, the cycle contraints (2) impose, for all cycles in G, that, if one edge in the cycle is cut, so is at least one other. Thus, intuitively, if any path between to nodes is cut, there can not be a connection between these nodes via another path in G. Thus, the feasible solutions to the optimization problem from Eq.(1)–(2) are exactly all *partitionings* of the graph G. Given any sequence of images, we construct an instance of this problem by defining the graph $G = (V, E)$ and costs $c \in \mathbb{R}^E$. In the ideal case, each partition describes either the entire background or exactly one object throughout the whole video at two levels of granularity: the tracked bounding boxes of this object and the point trajectories of all points on the object. On the one hand, if an object is only detected in few video frames and missed in others, the connection between these detections can still be established in the graph via point trajectories. On the other hand, false detections usually do not move consistently with point trajectories and therefore tend to end up as isolated nodes. Thus, they can easily be removed in a postprocessing step. A proposed solution to the Correlation Co-

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

Clustering problem on the graph in Fig. 2 (b) is shown in Fig. 2 (c). It contains four clusters: one for each pedestrian tracked over time, and two background clusters in which no detections are contained.

Below, we first describe the definition of the low-level subgraph $G^{\text{low}} = (V^{\text{low}}, E^{\text{low}})$ whose nodes are point trajectories, then the definition of the high-level subgraph $G^{\text{high}} = (V^{\text{high}}, E^{\text{high}})$ whose nodes are bounding boxes, and finally the definition of inter-level edges E^{lh} that connect low-level point trajectories with high-level bounding boxes.

3.2 Low-Level Graph of Point Trajectories

At the low level, we define the graph $G^{\text{low}} = (V^{\text{low}}, E^{\text{low}})$ whose nodes are point trajectories and whose edges connect point trajectories that potentially belong to the same group. In addition, we define, for every edge $e^{\text{low}} := \{u, v\} \in E^{\text{low}}$, a cost $c_{e^{\text{low}}} \in \mathbb{R}$ to be payed for any feasible solution that assigns the point trajectories u and v to distinct groups.

A point trajectory $u \in V^{\text{low}}$ is a spatio-temporal curve that describes the long-term motion of its starting point. We compute point trajectories from the image sequence by the algorithm of [54]. For this, we track by large displacement optical flow [13] all points sampled for the first image at a certain sampling rate for which the image has sufficient structure. A point trajectory is ended if the consistency between forward and backward optical flow is large, indicating that the point is occluded or lost. Whenever the trajectory density is lower than intended and the current image has sufficient structure, we start a new trajectories in order to maintain the desired sampling rate. For edges $e^{\text{low}} \in E^{\text{low}}$, we define the costs $c_{e^{\text{low}}} \in \mathbb{R}$ exactly as Keuper et al. [40]. That is, we compute the maximum motion difference $d^{\text{m}}(u, v)$ between the trajectories u and v connected by e^{low} during their shared time interval, as proposed by Ochs, Malik and Brox [54] as

$$d^{\text{m}}(u, v) = \max_t \frac{\|\partial_t u - \partial_t v\|}{\text{var}_t}, \tag{3}$$

where $\partial_t u$ and $\partial_t v$ are the partial derivatives of trajectories u and v with respect to the time dimension and var_t is the variation of the optical flow in this frame. Intuitively, the normalization by var_t accounts for the fact that a small motion difference between two trajectories is more important in a frame with hardly any motion than in a frame with generally strong, possibly higher order motion (compare [54] for more details). In addition, we compute a color distance $d^{\text{c}}(u, v)$ and a spatial distance $d^{\text{sp}}(u, v)$ between each pair of trajectories that share at least one image, and spatial distances also for trajectories without temporal overlap. We combine these distances non-linearly according to $c_{uv} := \max\{\theta_0 + \theta_1 d^{\text{m}} + \theta_2 d^{\text{c}} + \theta_3 d^{\text{sp}},\ \theta_4 + \theta_1 d^{\text{m}}\}$. Ideally, the parameters $\theta \in \mathbb{R}^5$ would be learned from training data. In reality, training data for motion segmentation is scarce. Thus, we set θ as defined and validated on training data in [40].

3.3 High-Level Graph of Bounding Boxes

At the high level, we construct a graph $G^{\text{high}} = (V^{\text{high}}, E^{\text{high}})$ whose nodes are bounding boxes and whose edges connect bounding boxes that potentially belong to the same object. In addition, we define, for every edge $e^{\text{high}} := \{u, v\} \in E^{\text{high}}$, a costs $c_{e^{\text{high}}} \in \mathbb{R}$ to be payed for any feasible solution that assigns the bounding boxes u and v to distinct objects.

For the two experiments we conduct and describe in Section 4, the one with the FBMS59 motion segmentation benchmark and the other with the MOT tracking benchmark, the construction of the graph and edge costs is different. For example, we define a faster R-CNN [62] bounding box object detector for the FBMS59 motion segmentation benchmark while we adhere to bounding boxes that are given for the MOT tracking benchmark, as required to evaluate on this benchmark. In both cases, the underlying object model allows to produce a tentative frame-wise object segmentation or template T_v of the detected object $v \in V^{\text{high}}$. Such a segmentation template can provide far more information than the bounding box alone. Potentially, a template indicates uncertainties and enables to find regions within each bounding box, where points most likely belong to the detected object.

Further commonalities between the two constructions are described here. Differences are described in detail in Section 4.

We consider between every pair of bounding boxes their intersection over union (IoU). As the plain bounding box IoU is less informative for larger temporal distance, we additionally compute the distance proposed by Tang et al. [68] based on Deep Matching [75]. For every pair of frames t_a and t_b and every detection u in t_a, Deep Matching generates a set of matched keypoints M_{u,t_b} inside the detection. For every pair of detections u in t_a and v in t_b with $t_a \neq t_b$, we can compute the intersection as $\text{MI}_{uv} = |\text{M}_{u,t_b} \cap \text{M}_{v,ta}|$ and the union as $\text{MU}_{uv} = |\text{M}_{u,t_b} \cup \text{M}_{v,t_a}|$. Then, the Deep Matching based IoU can be computed as

$$\text{IoU}^{\text{DM}}_{uv} = \frac{\text{MI}_{uv}}{\text{MU}_{uv}} \tag{4}$$

IoU^{DM} can be understood as a robust IoU measure. It is especially needed when bounding boxes in non-neighboring frames are to be compared. In these cases, the traditional IoU does not provide a reliable signal because objects or the camera might have moved significantly. Compare [68] for a thorough analysis.

If the IoU between two bounding boxes is zero, we need to measure their spatial difference. To this end, we consider, for every bounding box u, its spatio-temporal center $r_u = (x_u, y_u, t_u)^\top$ and size $(w_u, h_u)^\top$. For every edge $\{u, v\} \in E^{\text{high}}$ between bounding boxes u and v, we compute the normalized distance between u and v

$$d^{\text{sp}}(u, v) = 2 \left\| \begin{pmatrix} (x_u - x_v)/(w_u + w_v) \\ (y_u - y_v)/(h_u + h_v) \end{pmatrix} \right\| , \tag{5}$$

where $\|.\|$ denotes the ℓ_2-norm and the factor 2 accounts for the normalization of the distance between the bounding box centers by the average of their widths and heights. Intuitively, small, non-overlapping bounding boxes whose centers are far away from each other are less likely to belong tho the same objects than large bounding boxes at the same distance.

Both d^{sp} (5) and IoU are used for computing the edge weights c_{uv} for $\{u, v\} \in E^{\text{high}}$. However, the exact computation depends on the task and dataset, where different information is available. For the multiple object tracking task, all detected objects are pedestrians and can thus share a common template T while the object categorie is unknown for the motion segmentation task. On the MOT datasets, detections are provided after non-maximum suppression and thus might be missing in some frames. Thus, robust longer distance connections might be necessary. In contrast, on motion segmentation, we ran our own detector and thus have access to overlapping and low-scoring detections. We will discuss these details in our experiments.

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

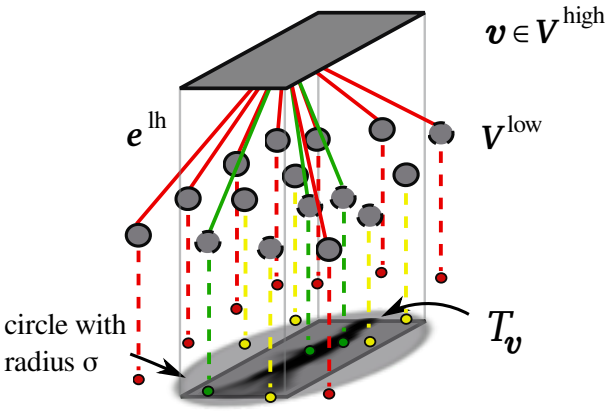


Fig. 3. Edges e^{lh} between high and low level nodes. For every detection v, the template T_v is evaluated at the spatial location of every trajectory $u \in V^{\text{low}}$. An edge with an attractive cost c_e^{lh} is introduced if u intersects with T_v in a location of high object probability (green edges). If u misses the template T_v and the distance $d^{\text{sp2}}(u, v)$ to the center of T_v is larger than a threshold σ (indicated by the gray circle), an edge with repulsive edge cost is introduced (red). If u intersects with T_v in a location of low object probability and the distance is smaller than σ, no edge is introduced.

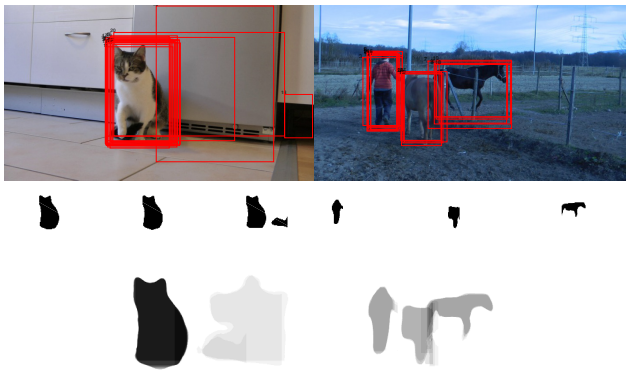


Fig. 4. Examples of the faster R-CNN object detections on images from FBMS59 sequences [54]. The first row shows the best 20 detections. The second row shows three exemplary templates T generated with DeepLab [17], [55] on these detections.

3.4 Inter-Level Edges

For every image t, every bounding box v detected in this image and every point trajectory u intersecting this image, we consider the size (w_v, h_v) and center $(x_v, y_v)^\top$ of the bounding box. We compare the center of the bounding box with the point $(x_u, y_u)^\top$ in which the trajectory intersects with the image by the metric

$$d^{\text{sp2}}(u, v) = 2 \left\| \begin{pmatrix} (x_u - x_v)/w_v \\ (y_u - y_v)/h_v \end{pmatrix} \right\| , \tag{6}$$

where the factor 2 corrects for the fact that we divide the distance between point trajectory and bounding box center by the full width and height. Thus, the normalized distance d^{sp2} is 1 along an ellipse with shape parameters $w_v/2$ and $h_v/2$. For $d^{\text{sp2}} > \sqrt{2}$, the bounding box is fully contained within the ellipse. As the probability that a bounding box $v \in V^{\text{high}}$ and a point trajectory $u \in V^{\text{low}}$ relate to the same object visible in the image depends more specifically on the relative location of both, we encode by $T_v(x, y) \in (0, 1)$ the probability that the point (x, y) in the image plane is covered by the shape of the object represented by the bounding box v. See Fig. 3 for an illustration. For every detection v, the template T_v is evaluated at the spatial location of every trajectory $u \in V^{\text{low}}$. An edge with an attractive cost c_e^{lh} is introduced if u intersects with T_v in a location of high object probability. If u misses the template T_v and the distance $d^{\text{sp2}}(u, v)$ to the center of T_v is larger than a threshold σ, an edge with repulsive edge cost is introduced. If u intersects with T_v in a location of low object probability and the distance is smaller than σ, no edge is introduced.

Specifically, we define a probability $p_{uv} \in [0, 1]$ of the bounding box $v \in V^{\text{high}}$ and the point trajectory $u \in V^{\text{low}}$ belonging to distinct objects as

$$p_{uv} := \begin{cases} 1 - T_v(x_u, y_u) & \text{if } T_v(x_u, y_u) > \frac{1}{2} \\ 1 & \text{if } d^{\text{sp2}}(u, v) > \sigma \\ \frac{1}{2} & \text{otherwise} \end{cases} . \tag{7}$$

The parameter $\sigma \in \mathbb{R}^+$ depends on the application. It has to be chosen sufficiently large such that it does not conflict with the first case in (7). Intuitively, its choice depends, on the one hand, on the localization accuracy of bounding boxes, on the other hand on the density of objects that need to be distinguished. A small σ allows the insertion of repulsive terms to trajectories on nearby objects. However, if the localization is inaccurate, small σ values can lead to oversegmentation.

W.r.t. the probability p_{uv}, we define the cost $c_{uv} := \text{logit}(p_{uv}) = \log \frac{p_{uv}}{1-p_{uv}}$.

3.5 Search for Feasible Solution

In order to find feasible solutions of low cost to the instances of the NP-hard correlation co-clustering problem that we construct from image data as described above, we employ the efficient primal feasible local search heuristic of [41].

4 Experiments

In this section, we apply the proposed correlation co-clustering problem to the task of motion segmentation and multiple object tracking and show the following evaluations:

- We show results for the FBMS59 [54] motion segmentation dataset containing sequences with various object categories and motion patterns (Sec. 4.1).
- We show results for the 2D MOT 2015 benchmark [44], the MOT 2016 benchmark [50] and the MOT 2017 benchmark [44], [50] for multiple object tracking (Sec. 4.2).
- We compare our segmentations on two of these sequences to the previous approach to joint segmentation and tracking by Milan et al. [49] (Sec. 4.3).
- We report results for the tracking performance of our model on three standard multiple object tracking sequences of [3], [81]. The evaluation on these sequences allows a comparison to Fragkiadaki et al. [30] and Tang et al. [67] (Sec. 4.4).

4.1 Motion Segmentation

The FBMS59 [54] motion segmentation dataset consists of 59 sequences split into a training set of 29 and a test set of 30 sequences. The videos are of varying length (19 to about 500 frames) and show diverse types of moving objects such as cars, persons and different types of animals. The results are evaluated in

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

terms of segmentation precision and recall, the aggregate f-measure and the number of segmented objects with f-measure ≥ 0.75 for different levels of trajectory sampling rates as well as for densified segmentations using the variational method from Ochs et al. [52]. Among these measures, the f-measure is the most representative since it reflects the trade-off between precision and recall.

4.1.1 Implementation Details

To apply the correlation co-clustering problem to this data, the very first question is how to obtain reliable detections in a video sequence without knowing the category of the object of interest. To this end, we use detections from the Faster R-CNN [62] detector, trained on the PASCAL VOC 2012 dataset.

Faster R-CNN is an object detector that integrates a region proposal network with the Fast R-CNN [31] network. In our experiments, we compute detections using the code and model published with their paper. We only use the most confident detections, i.e., those with detection scores above a threshold of 0.97, on a scale between 0 and 1. This yields a sparse set of detections with high precision but potentially low recall.

From these detections, we generate segmentation proposals using DeepLab [17], [55]. These tentative segmentations serve as templates for the computation of pairwise costs between detections and trajectories. Examples of detections and corresponding templates per frame are shown in Fig. 4. These examples show the localization quality of the detections.

Since occlusion does not play a significant role in this dataset, we compute pairwise terms between detections only within the same frame and in directly neighboring frames. This way, we can use the standard intersection over union (IoU) definition computed directly on the templates. From the IoU and the pairwise distance d^{sp} from (5), we compute the pseudo cut probability between two bounding boxes $u, v \in V^{\text{high}}$ as

$$p_{uv} = \begin{cases} \frac{\exp(-q)}{1+\exp(-q)} & \text{if IoU}(u,v) > 0.7 \\ \frac{1}{1+\exp(-q')} & \text{if } d^{\text{sp}}(u,v) > 1.2 \\ \frac{1}{2} & \text{otherwise} \end{cases} \tag{8}$$

Here, $q := -20 \cdot (0.7 - \text{IoU}(u,v))$ and $q' := -5 \cdot (1.2 - d^{\text{sp}}(u,v))$. Note that an IoU > 0.7 implies a distance $d^{\text{sp}} < 1.2$. We have chosen these parameters so as to yield reasonable results on the FBMS59 training set.

The cost c_{uv} is computed from the probability p_{uv} according to (7) with $\sigma = 2$. This large threshold accounts for the uncertainty in the bounding box localizations.

4.1.2 Baseline Experiments

As a baseline that helps assessing the impact of the segmentation templates from DeepLab [17], [55] , we experiment with a trivial template, i.e. an ellipsoid placed in the center of each bounding box with shape parameters 0.5 times the bounding boxes width and height, respectively. This template's link probability decreases linearly with the normalized distance from the bounding box center, being 1 for $d^{\text{sp2}} = 0$ and 0.5 for $d^{\text{sp2}} = 0.5$.

To further assess the impact of erroneous detections and segmentation templates on the optimization, we ran an oracle experiment using the provided sparse ground truth segmentations and their bounding boxes as high-level cues. In theory, these ground truth segmentations should support the grouping of point trajectories which belong to the same object while avoiding to group point trajectories from different objects. This should lead to less conflicting information than the use of detection and segmentation estimates.We evaluate the impact of the available sparse ground truth on the trajectory level segmentation quality.[1]

Fig. 5. Examples of CCC segmentation results densified by the variational method of Ochs et al. [52] on three sequences of the FBMS59 [54] benchmark.

To assess the impact of the joint model components, we evaluate, for 8 pixel trajectory sampling, not only the full model but also its performance if costs between detection nodes are omitted (CCC - E^{high}).

4.1.3 Results

The quantitative evaluation of results on the FBMS59 benchmark is shown in Tab. 1 in terms of precision and recall, the aggregate f-measure and the number of segmented objects with f-measure ≥ 0.75. The motion segmentation considering only the trajectory information from [40] performs already well on the FBMS59 benchmark. When the high-level information from object detections and DeepLab templates is added to this model (CCC - E_h), the f-measure improves by 2%. Our full model CCC yields a further improvement by 1%, for 8 pixel point sampling. Note that we outperform the baseline method [40] by a significant margin on the test set. We outperform also the higher-order spectral clustering method [53] as well as the higher-order multicut model from [39].

To assess the importance of the informative templates from DeepLab, we evaluate our ellipse-shaped baseline template. The according results are denoted by CCC BBX-baseline. It can be observed that this un-informed template still yields an improvement of about 1% in f-measure and an increase in the number of detected objects on both datasets over the baseline method [40].

From the experiment on the sparsely available oracle detections and segmentations (*sparse oracle* in Tab. 1), we can also observe an improvement over the baseline [40] without such information. However, since the ground truth is only provided for every 20th frame, the oracle results are poorer than the ones obtained using fasterRCNN detections and DeepLab segmentations. The additional, noisy information on *all* frames leads to an improvement over only sparsely available ground truth information.

For denser sampling rate with 4 pixel distance, we only compare our full model to the baseline method [40]. The behavior is similar. The densified version of our segmentations improves over those from [40] by more than 3% on both datasets. A visualization of densified results is shown in Fig. 5.

Qualitative results of the motion segmentation as well as the tracking are shown in Fig. 6 and 7. Due to the detection information,

1. Since we are solving a single, constrained optimization problem on trajectory and bounding box level, we can not directly investigate the impact of this ground truth information on the trajectory subproblem in terms of the resulting energy.

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

Algorithm	Sampling	Training set				Test set			
		Precision	Recall	f-measure	# Objects	Precision	Recall	f-measure	# Objects
SC [54]	8	85.10%	62.40%	72.0%	17/65	79.61%	60.91%	69.02%	24/69
SC+HO [53]		81.55%	59.33%	68.68%	16/65	82.11%	64.67%	72.35%	27/69
Lifted HO MC [39]		86.83%	77.79%	82.06%	32/65	87.77%	71.96%	79.08%	25/69
MCe [40]		86.73%	73.08%	79.32%	31/65	87.88%	67.7 %	76.48%	25/69
CCC BBX-baseline		86.92%	75.73%	80.94%	34/65	82.77%	72.36%	77.22%	31/69
CCC - E^{high}		83.46%	79.46%	81.41%	**35/65**	84.06%	76.89%	80.30%	**35/69**
CCC		84.85%	80.17%	**82.44%**	**35/65**	84.52%	77.36%	**80.78%**	**35/69**
sparse oracle		90.04%	76.19%	82.25%	34/65	86.53%	69.82%	77.14%	27/69
MCe [40]	4	86.79%	73.36%	79.51%	28/69	86.81%	67.96%	76.24%	25/69
CCC		83.81%	78.16%	**80.89%**	**32/69**	84.61%	77.28%	**80.78%**	**37/69**
treeDL [56]	dense	-	-	-	-	78.41%	65.52%	72.33%	-
MCe [40]		85.31%	68.70%	76.11%	24/65	85.95%	65.07%	74.07%	23/69
CCC		84.28%	75.15%	**79.66%**	**29/65**	83.17%	74.65%	**78.68%**	**32/69**

TABLE 1
Results for the FBMS-59 training and test set. For both trajectory sampling rates as well as for densified segmentations, the proposed model CCC improves over the state of the art.

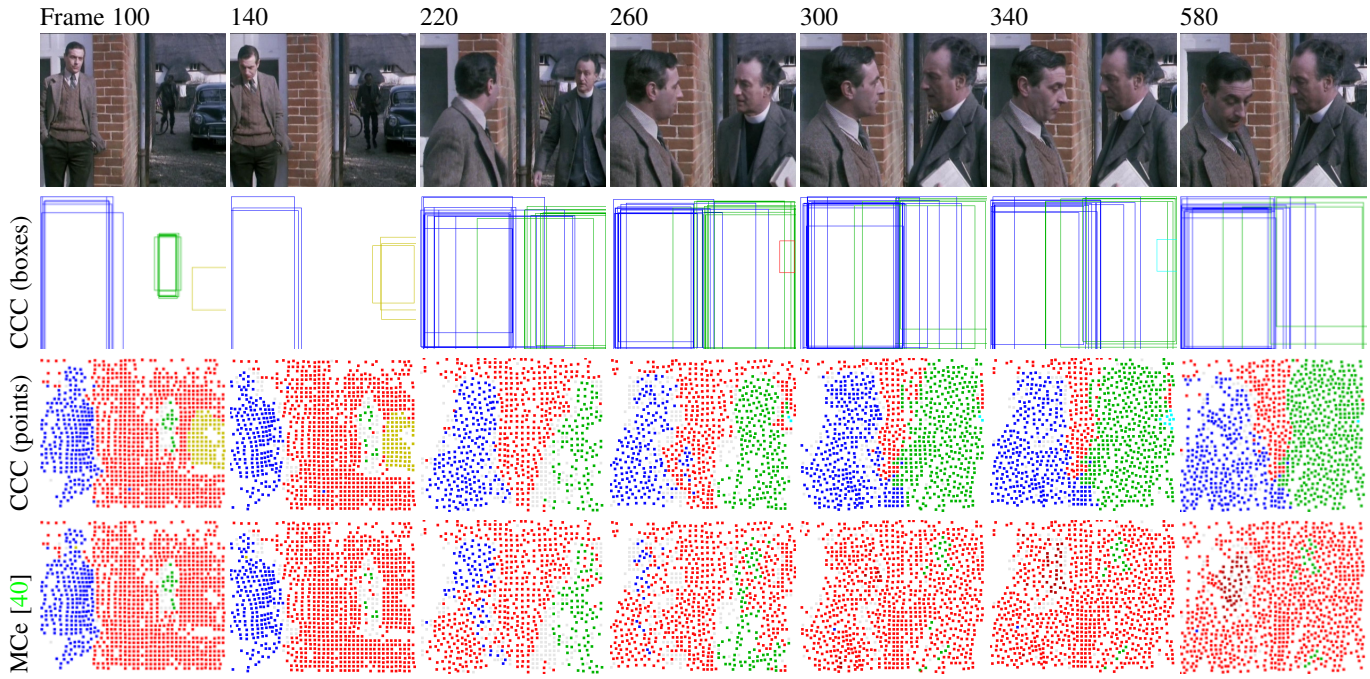


Fig. 6. Comparison of the proposed CCC model and the trajectory multicut (MCe) [40] on the *marple6* sequence of FBMS59. While MCe can not properly segment the persons, the tracking information from the bounding box subgraph helps our joint model to segment the two men throughout the sequence despite scaling and rotational motion. Additionally, static, consistently detected objects like the car in the first part of the sequence are segmented as well. As these are not annotated, this causes over-segmentation penalty on the FBMS59 metrics.

static objects like the car in the *marple6* sequence (yellow cluster) can be segmented. The man approaching the camera in the same sequence can be tracked and segmented (green cluster) throughout the sequence despite the scaling motion. Similarly, in the *horses06* sequence, all three moving objects can be tracked and segmented through strong partial occlusions. As the ground truth annotations of FBMS59 are sparse and only describe moving objects, we cannot assess the multiple object tracking performance for this data set.

4.2 Multi-Target Tracking on MOT

We now apply the proposed correlation co-clustering problem to the task of multiple object tracking and show the benefit of this joint approach in terms of the 2D MOT 2015 [44] (MOT15), MOT 2016 [50] (MOT16) and MOT 2017 (MOT17) benchmarks. These benchmarks contain videos from static and moving camera recorded in unconstrained environments. MOT15 contains 11 training and 11 test sequences, MOT16 and MOT17 consist of 7 sequences each in training and test. While the sequences in MOT16 and MOT17 are identical, the datasets differ (1) in the ground truth annotations, which have presumably been improved from MOT16 to MOT17, and (2) in the given pedestrian detections. In all three benchmarks, detections for all sequences are provided and allow for direct comparison to other tracking methods. While the detections in MOT15 are computed using the Aggregate Channel Features

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

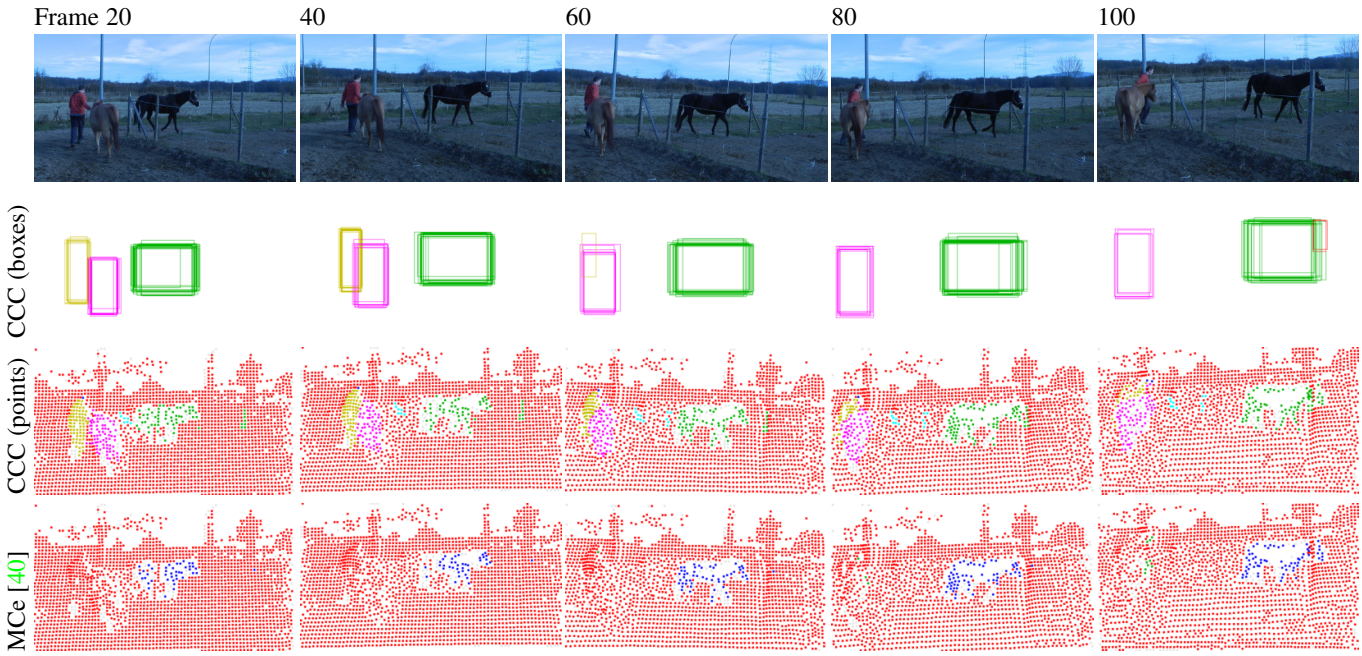


Fig. 7. Segmentation and tracking results of the proposed CCC model and the trajectory multicut (MCe) [40] on the *horses06* sequence of FBMS59. MCe can not segment the person and the horse next to him due to the difficult motion and strong partial occlusions.

pedestrian detector [23], DMP v5 [27] detections are provided for MOT16. MOT17 provides three different sets of detections [26], [62], [78] for each sequence in order to encourage tracking approaches that generalize well over different object detectors.

The tracking performance on the official MOT15 [44], MOT16 [50], and MOT17 [44], [50] benchmarks is evaluated in terms of the CLEAR MOT evaluation metrics [10]. We report the ID F1 score, i. e. the ratio of correctly identified detections over the average number of ground-truth and computed detections (IDF1), the number of mostly tracked (MT) and mostly lost (ML) objects, the fragmentation (FM) and MOTA (multiple object tracking accuracy), which is a cumulative measure combining missed targets (FN), false alarms (FP), and identity switches (IDs).

4.2.1 Implementation Details

We connect every bounding box u to every other bounding box v within a distance of 3 frames in MOT15 and MOT16, 5 frames in MOT17. To compute pairwise costs c_{uv} between bounding boxes u and v, we consider the detection scores $s_u, s_v \in \mathbb{R}$, their minimum $s_{uv} := \min\{s_u, s_v\}$ and the Deep Matching distance $\text{IoU}^{\text{DM}}_{uv}$ as defined in equation (4). As Tang et al. [68], we define the feature vector f_{uv} as

$$f_{uv} := (\text{IoU}^{\text{DM}}_{uv}, s_{uv}, \text{IoU}^{\text{DM}}_{uv} \cdot s_{uv}, (\text{IoU}^{\text{DM}}_{uv})^2, s^2_{uv}) \quad (9)$$

and learn the costs c_{uv} from f_{uv} by logistic regression.

Pairwise costs between a bounding box $u \in V^{\text{high}}$ and a point trajectory $v \in V^{\text{low}}$ are computed according to (7), with $\sigma = 1.5$. The template T_u is computed as the average pedestrian shape from the shape prior training data provided in [21] and its horizontally flipped analogon. This template is depicted Fig. 8. It is identical for all bounding boxes up to scaling.

As the bounding boxes that come with the data set are relatively sparse (due to non-maximum suppression), the statistics of the graph are altered. To compensate for this fact, we apply a simple heuristic. Assuming that about 20 bounding boxes have been suppressed for every true detection in 2D MOT 2015 and about 4 bounding boxes have been suppressed for every true detection in MOT 2016, we weight the links between trajectory and detection nodes by factor 20 and 4 respectively. We are aware that this is a crude heuristic. Better options would be to learn this factor per sequence type or (better) to use the detections before non-maximum suppression which are unfortunately not provided. The conversion from clusters to tracks is done as in [68]. Specifically, in each frame, we obtain object locations by averaging all detections belonging to the same cluster, weighted by their detection score. A track is computed by connecting these averages of every cluster over time. Due to the detection scores included in the pairwise terms between bounding boxes, false detections tend to end up as isolated nodes. As [68], we eliminate all clusters of size less than 5 in all experiments. Missing detections within a track are hallucinated by bilinear interpolation. On the MOT15 data, we additionally hallucinate missing detections in up to three neighboring frames to a resulting track by following the point trajectories associated with this track if available.

Fig. 8. The average pedestrian shape template used for the computation of pairwise terms between pedestrian detections and trajectories.

4.2.2 Results

Here, we evaluate the tracking performance on the official MOT15 [44], MOT16 [50], and MOT17 [44], [50] benchmarks in terms of the CLEAR MOT evaluation metrics [10]. Results for the MOT15 benchmark are shown in Tab. 2. We compare to the state-of-the-art multi-target tracking method on MOT15 [19], and the very recent methods from [16], [63], which employ convolutional neural network based appearance features, Sadeghian et al. [63] in conjunction with LSTMs to establish long-term dependencies. Our results are competitive in MOTA and improve over methods

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

	IDF1	MT	ML	FP	FN	IDs	FM	MOTA
Long et al. [16]	**47.1**	8.7%	37.4%	**4,005**	33,203	586	1,263	**38.5**
Sadeghian et al. [63]	46.0	15.8%	**26.8%**	7,933	29,397	1,026	2,024	37.6
Choi [19]	44.6	12.2%	44%	7,762	32,547	**442**	823	33.7
Milan et al. [49]	31.5	5.8%	63.9%	7,890	39,020	697	**737**	22.5
CCC	45.1	**23.2%**	39.3%	10,580	**28,508**	457	969	35.6

TABLE 2
Multi-target tracking results on the 2D MOT 2015 benchmark. On the aggregate measure MOTA, we improve over [19] and [49], as well as in the important metrics MT (mostly tracked objects) and FN (the number of false negatives).

	IDF1	MT	ML	FP	FN	IDs	FM	MOTA
Choi [19]	**53.3**	18.3%	41.4%	9,753	87,565	**359**	**504**	46.4
Tang et al. [68]	46.3	15.5%	39.7%	6,373	90,914	657	1.114	46.3
Tang et al. [70]	51.3	18.2%	40.1%	6,654	86,245	481	595	**48.8**
Henschel et al. [32]	44.3	19.1%	**38.2%**	8,886	**85,487**	852	1,534	47.8
Levinkov et al. [45]	47.3	18,2%	40.4%	**5,844**	89,093	629	768	48.4
CCC	52.3	**20,4%**	46.9%	6,703	89,368	370	598	47.1

TABLE 3
Multi-target tracking results on the MOT16 benchmark. Here, we improve over the state of the art in the metric MT (mostly tracked objects), while all top methods are very close in the MOTA. Again, our CCC model yields a low number of ID switches.

	FAF	MT	ML	FP	FN	IDs	FM	MOTA
Henschel et al. [32]	**1.3**	**21.2%**	**36.3%**	**22,732**	250,179	2,583	4,141	**51.2**
Kim et al. [47]	**1.3**	20.8%	36.9%	22,875	252,889	2,314	**2,865**	50.7
CCC	1.4	20.7%	37.4%	24,986	**248,328**	**1,851**	2,991	**51.2**

TABLE 4
Multi-target tracking results on the MOT17 challenge. Instead of the ID F1 score, the false alarm frequency (FAF) was reported in the challenge. Our CCC model yields the lowest number of ID switches while performing on par with Henschel et al. in terms of MOTA, outperforming all other challenge submissions.

which are, as ours, based on weak appearance terms [19]. In comparison, we observe a decrease in the number of false negatives while false positives increase. In fact, the large amount of false positives our method produces might be due to the hallucinated detections, which therefore seems to have a rather negative impact on the overall MOTA score. We show a clear improvement over the performance of the previously proposed method for joint tracking and segmentation [49].

Results for the MOT16 benchmark are shown in Tab. 3. Here, we first compare to the MOT 2016 Challenge winning approach by Tang et al. [68], as well as to the approach by Levinkov et al. [45], which is also based on correlation clustering. While [68] solve a correlation clustering problem on a bounding box graph with advanced features, [45] solve a node labeling minimum cost multicut problem that allows to discard unreliable bounding boxes. Our joint model can improve over [68] by reducing the number of identity switches and fragmentations while keeping the number of false alarms low, resulting in a better MOTA. Compared to [45] our CCC model is slightly worse in MOTA because of the higher number of false positives. However, we outperform [45] in terms of mostly tracked objects and ID switches. As for MOT15, our method is outperformed by a deep learning based approach [69], which establishes long term connections by a strong, learned appearance term. Such information could be included in our approach.

Results for the MOT17 challenge are shown in Tab. 4. Following the general tendency of the results on MOT15 and MOT16, the proposed approach achieves a low number of ID switches and a good MOTA score. Together with Henschel et al. [32], the proposed approach won the MOT17 challenge [2] This indicates good performance without extensive parameter optimization. After the MOT17 challenge, Henschel et al. [32] updated their results on the MOT17 benchmark and improved the MOTA by 0.1 on this data. Unlike our approach, their method is not only based on the provided object detections but employs a specifically trained head detector to provide an additional high-level cue.

4.3 Segmentation Evaluation on Tracking Sequences

In order to assess the quality of the resulting motion segmentations in the tracking scenario, we evaluate our sparse segmentations on the pedestrian tracking sequence *tud-crossing* from the MOT15 benchmark. For this sequence, segmentation annotations in every 10th frame have been published in [24]. The pedestrian motion segmentation is evaluated with the metrics precision (P), recall (R), f-measure (F) and number of retrieved objects (O) as proposed for the FBMS59 benchmark [54].

2. The MOT17 challenge was held during the 1st Joint BMTT-PETS Workshop on Tracking and Surveillance in conjunction with the Conference on Computer Vision and Pattern Recognition - CVPR 2017, https://motchallenge.net/MOT17_results_2017_07_26.html

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence

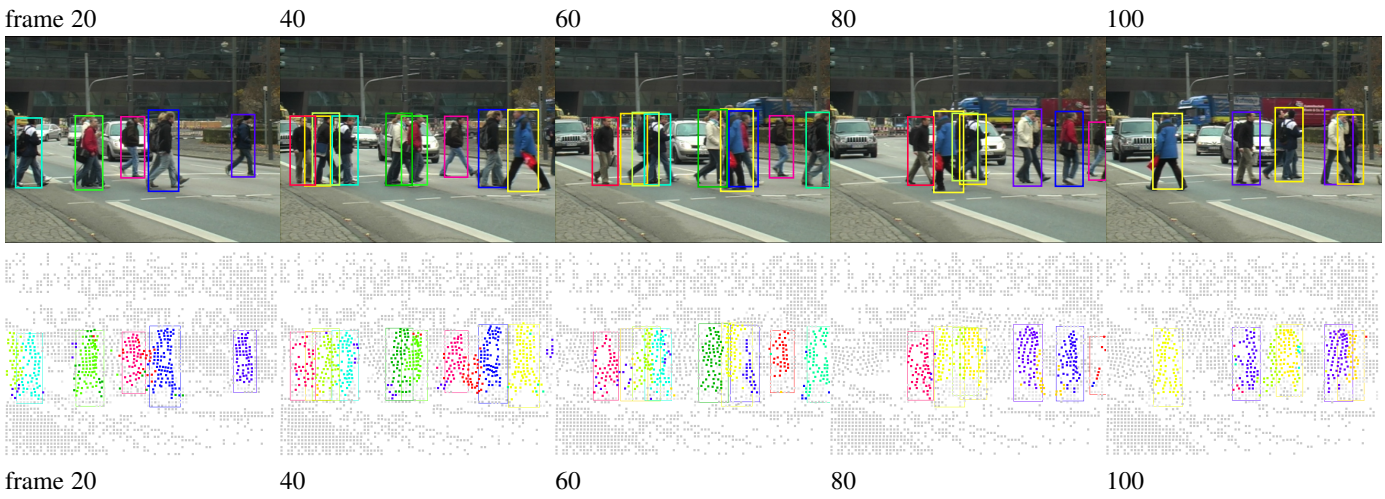


Fig. 9. Results of the proposed Correlation Co-Clustering model on the *tud-crossing* sequence from MOT15.

TUD-Crossing	Precision	Recall	f-measure	O (≥ 75)	O (≥ 60)
SC [54]	67.92%	20.16%	31.09%	0/15	1/15
MCe [40]	43.78%	38.53%	40.99%	1/15	1/15
CCC - E^{high}	62.05%	54.72%	58.15%	1/15	**9/15**
CCC - E^{low}	69.37%	48.88%	57.35%	**2/15**	**9/15**
CCC	67.22%	55.11%	**60.57%**	**2/15**	**9/15**

TABLE 5
Motion Segmentation on the Multi-Target Tracking sequence *tud-crossing*. **O** is the number of extracted objects, with f-measure $\geq 75\%$ and with f-measure $\geq 60\%$ respectively. All results are computed for sparse trajectory sampling at 8 pixel distance, leading to an average region density of 0.85%.

TUD-Crossing	Precision	Recall	f-measure
Milan et al. [49]	60.61%	19.25%	29.23%
dense CCC	61.01%	46.98%	**53.08%**

TABLE 6
Motion Segmentation on the *tud-crossing* sequence from MOT15.

PETS-S2L2	cl.err.	per-reg.err.	over-seg.	extr. obj.
Milan et al. [49]	**3.56**	24.34	1.42	7
dense CCC	4.38	**23.20**	**0.83**	**11**

TABLE 7
Segmentation evaluation on the *PETS-S2L2* sequence from MOT15. As Milan et al. [49], we report the clustering error (percentage of misclassified pixels); the per-region error (average ratio of wrongly labeled pixels per ground truth mask); the oversegmentation error (number of segments covering each mask); and the number of extracted objects as those correctly segmented in at least 90% of their area).

To assess the importance of the model parts, we consider two baseline experiments. Specifically, we not only evaluate the full CCC model but also the performance without costs between trajectories (CCC - E^{low}) as well as the performance when omitting the pairwise terms between tracklet nodes (CCC - E^{high}).

A qualitative result is shown in Fig. 9. The bounding boxes overlayed on the image sequence are, for every frame and cluster, the ones with the highest detection score. These were also used for the tracking evaluation. The second row visualizes the trajectory segmentation. Both detection and trajectory clusters look satisfying. Thanks to the segmentation, better localizations for the tracked pedestrians can be provided.

Quantitative results and a comparison with the motion segmentation methods [40], [54] are shown in Tab. 5. The comparison between the full model CCC and its parts CCC - E^{low} and CCC - E^{high} confirms that the full, joint CCC model performs best. On the important f-measure, CCC improves over the previous state-of-the-art in motion segmentation on this sequence.

We want to compare our motion segmentation results on tracking sequences to those from Milan et al. [49]. Therefore, we densify our sparse segmentation results using [51] and recompute the segmentation from [49] using their code with the default parameters. The results are given in Tab. 6. At a similar precision, our segmentations show a higher recall and consequently, a better f-measure.

For further comparison to Milan et al. [49], we also evaluate our densified segmentations on the *PETS-S2L2* sequence used in their paper for evaluation. Here we evaluate on the same standard segmentation measures as [49]. The results are given in Tab. 7. While the clustering error is lower for [49], the proposed CCC model outperforms [49] in all other metrics.

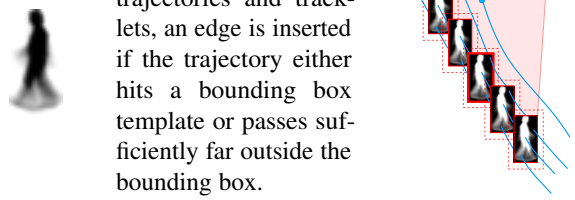
4.4 Comparison to Related Tracking Methods

We evaluate the tracking and segmentation performance of our Correlation Co-Clustering model on the publicly available sequences: TUD-Campus, TUD-Crossing [3] and ParkingLot [81]. These sequences have also been used to evaluate the Subgraph Multicut method by Tang et al. [67] and therefore allows for direct comparison to this method. A direct comparison to the Two-Granularity-Tracking method by Fragkiadaki et al. [30] is provided on the TUD-Crossing sequence for which results are reported in [30].

4.4.1 Implementation Details

To allow for direct comparison to Tang et al. [67], we compute all high-level information, i.e. the detection nodes $v \in V^{\text{high}}$,

is article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2018.2876253, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) Mean directed pedestrian shape template

(b) Trajectory-Tracklet edges: For every pair of trajectories and tracklets, an edge is inserted if the trajectory either hits a bounding box template or passes sufficiently far outside the bounding box.

Fig. 10. The average pedestrian shape template and the trajectory-tracklet edges used for the comparison to subgraph multicut [67].

edges $e \in E^{\text{high}}$, and their costs c_e exactly as reported in [67] with only one difference: the Subgraph Multicut models from [67] employs not only pairwise but also unary terms which our proposed Correlation Co-Clustering model does not require. We omit these terms.

In [67], DPM-based person detections [26] are used. To add robustness and enable the computation of more specific pairwise terms, these detections are grouped to small, overlapping tracklets of length 5 as in [3] without applying any Non-Maximum Suppression. This is in accordance to [67] and therefore beneficial for a direct comparison. Since tracklets are computed in every frame, the same detections can be part of several (at most 5) tracklets. In the experiments on the MOT benchmarks in Sec. 4.2, this tracklet computation is not possible because detections are only provided after non-maximum-suppression.

Pairwise terms between the tracklets are computed from temporal distances, normalized scale differences, speed, spatio-temporal locations and dColorSIFT features [82], combined non-linearly as in [67].

The computation of pairwise terms c_{uv} between nodes $u \in V^{\text{low}}$ and $v \in V^{\text{high}}$ has to be adapted in this setup. Unlike in our standard setup, a high level node $v \in V^{\text{high}}$ does not directly represent a detection bounding box but rather a set of 5 boxes. We compute the average pedestrian shape from the shape prior training data provided in [21] (see Fig. 10 (a)). For every detection v, T_v denotes the pedestrian template shifted and scaled to the kth bounding box position and size. The tracklet information allows to determine the walking direction of the pedestrian, such that the template can be flipped accordingly. For every detection u_k with $k = \{1, \dots, 5\}$ of a tracklet $v \in V^{\text{high}}$, the cut probability $p_{u_k w}$ to a trajectory node $w \in V^{\text{low}}$ is computed according to Eq. (7) with $\sigma = 1.2$. A trajectory node $w \in V^{\text{low}}$ is linked to a tracklet node $v \in V^{\text{high}}$ coexisting in a common frame with an edge cost $c_{wv} = \sum_{k=1}^{5} \text{logit}(p_{u_k w})$. Fig. 10 (b) visualizes the edges between tracklets and point trajectories.

4.4.2 Results

Quantitative results on the pedestrian tracking task are given in Tab. 8. Again, we evaluate the importance of the model parts (denoted by CCC-E^{high} and CCC-E^{low}). Among these, the proposed CCC model performs best on the MOTA metric, showing that the joint approach works better than any of its parts.

Compared to other methods, the proposed approach shows the general tendency to reduce the number of false negatives, while the number of false positives is higher than in [67].

On the sequences *TUD-Campus* and *TUD-Crossing*, we also compare to previous approach to joint segmentation and tracking [49]. The results for *TUD-Campus* were obtained using their code, while the result for [49] on *TUD-Crossing* is taken from the paper. For both sequences, our joint approach CCC outperforms this previous method. Fragkiadaki et al. [30] also provide results for the TUD-crossing sequence. They achieve a MOTA of 82.9 on this sequence. This result is close to but below ours.

	GT	MT	ML	FP	FN	IDs	FM	MOTA
TUD-Campus	8							
Milan et al. [49]		1	4	25	242	**0**	1	25.6
Subgraph MC [67]		5	1	**2**	58	**0**	1	83.3
CCC - E^{low}		6	1	19	**35**	**0**	**0**	85.0
CCC - E^{high}		5	1	20	63	3	2	76.0
CCC		5	1	5	45	1	**0**	**85.8**
TUD-Crossing	13							
Fragkiadaki et al. [30]		-	-	-	-	0	-	82.9
Milan et al. [49]		3	3	37	456	15	16	53.9
Subgraph MC [67]		8	2	**11**	198	**1**	**1**	80.9
CCC - E^{low}		9	0	22	161	5	11	82.9
CCC - E^{high}		12	0	204	**83**	14	5	72.7
CCC		9	0	22	160	2	9	**83.3**
ParkingLot	14							
Subgraph MC [67]		13	0	**113**	95	**5**	18	**91.4**
CCC - E^{low}		13	0	164	85	9	**13**	89.5
CCC - E^{high}		13	0	307	**79**	6	15	84.1
CCC		13	0	129	85	6	15	91.1

TABLE 8
Tracking result on multi-target tracking sequences TUD-Campus, TUD-Crossing [3] and ParkingLot [81]

4.5 Discussion

The proposed Correlation Co-Clustering method jointly deals with the related problems of trajectory-level motion segmentation and multiple object tracking. The joint task is achieved by phrasing a single and clean mathematical objective. The current setup has two limitations. First, the graph construction itself depends on several parameter choices. Currently, these parameters are manually set. Provided a sufficient amount of training data, these parameters could be learned or optimized by a grid search. Second, certified optimal solutions to the large and hard instances of the apx-hard problem we consider are out of our reach at the time of writing.

Contributions to both of these issues will most likely lead to a further improvement of results and will be subject to future research.

5 Conclusion

We have proposed a correlation co-clustering model for combining low-level grouping with high-level detection and tracking. We have demonstrated the advantage of this approach by combining bottom-up motion segmentation by grouping of point trajectories with high-level multiple object tracking by clustering of bounding boxes. We show that solving the joint problem is beneficial at the low level, in terms of the FBMS59 motion segmentation benchmark, and at the high level, in terms of the MOT detection and tracking benchmarks. Results of the proposed method are state-of-the-art in motion segmentation and winning entry of the MOT17 challenge for multiple object tracking.

REFERENCES

[1] B. Andres, J. H. Kappes, T. Beier, U. Köthe, and F. A. Hamprecht. Probabilistic image segmentation with closedness constraints. In *ICCV*, 2011. 2

[2] B. Andres, T. Kröger, K. L. Briggman, W. Denk, N. Korogod, G. Knott, U. Köthe, and F. A. Hamprecht. Globally optimal closed-surface segmentation for connectomics. In *ECCV*, 2012. 2

[3] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008. 5, 10, 11

[4] M. Andriluka, S. Roth, and B. Schiele. Monocular 3d pose estimation and tracking by detection. In *CVPR*, June 2010. 2, 3

[5] A. Andriyenko, K. Schindler, and S. Roth. Discrete-continuous optimization for multi-target tracking. In *CVPR*, 2012. 2, 3

[6] N. Bansal, A. Blum, and S. Chawla. Correlation clustering. *Machine Learning*, 56(1–3):89–113, 2004. 2

[7] T. Beier, B. Andres, U. Köthe, and F. A. Hamprecht. An efficient fusion move algorithm for the minimum cost lifted multicut problem. In *ECCV*, 2016. 2

[8] T. Beier, T. Kroeger, J. Kappes, U. Kothe, and F. Hamprecht. Cut, glue, & cut: A fast, approximate solver for multicut partitioning. In *CVPR*, 2014. 2

[9] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *CVPR 2011*, pages 3457–3464, June 2011. 2

[10] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: The clear mot metrics. *Image and Video Processing*, 1:1–10, 2008. 8

[11] G. Bertasius, J. Shi, and L. Torresani. High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision. In *ICCV*, 2015. 1, 2

[12] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010. 2

[13] T. Brox and J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE TPAMI*, 33(3):500–513, 2011. 4

[14] J. Chang, D. Wei, and J. W. F. III. A Video Representation Using Temporal Superpixels. In *CVPR*, 2013. 2

[15] V. Chari, S. Lacoste-Julien, I. Laptev, and J. Sivic. On pairwise costs for network flow multi-object tracking. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5537–5545, 2015. 2, 3

[16] L. Chen, H. Ai, C. Shang, Z. Zhuang, and B. Bai. Online multi-object tracking with convolutional neural networks. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 645–649, Sept 2017. 8, 9

[17] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 5, 6

[18] A. Cheriyadat and R. Radke. Non-negative matrix factorization of partial track data for motion segmentation. In *ICCV*, 2009. 2

[19] W. Choi. Near-online multi-target tracking with aggregated local flow descriptor. In *ICCV*, 2015. 1, 2, 8, 9

[20] S. Chopra and M. Rao. The partition problem. *Mathematical Programming*, 59(1–3):87–115, 1993. 2, 3

[21] D. Cremers, F. R. Schmidt, and F. Barthel. Shape priors in variational image segmentation: Convexity, lipschitz continuity and globally optimal solutions. In *CVPR*, 2008. 8, 11

[22] E. D. Demaine, D. Emanuel, A. Fiat, and N. Immorlica. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361(2–3):172–187, 2006. 2

[23] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *TPAMI*, 2014. 8

[24] B. L. E. Horbert, K. Rematas. Level-set person segmentation and tracking with multi-region appearance models and top-down shape information. In *ICCV*, 2011. 9

[25] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *ICCV*, 2013. 2

[26] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *TPAMI*, 2010. 8, 11

[27] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *TPAMI*, 32(9):1627–1645, 2010. 8

[28] K. Fragkiadaki, P. Arbelaez, P. Felsen, and J. Malik. Learning to segment moving objects in videos. In *CVPR*, 2015. 1, 2

[29] K. Fragkiadaki and J. Shi. Detection free tracking: Exploiting motion and topology for segmenting and tracking under entanglement. In *CVPR*, 2011. 1, 2

[30] K. Fragkiadaki, W. Zhang, G. Zhang, and J. Shi. Two-granularity tracking: Mediating trajectory and detection graphs for tracking under occlusions. In *ECCV*, 2012. 1, 2, 3, 5, 10, 11

[31] R. Girshick. Fast R-CNN. In *ICCV*, 2015. 6

[32] R. Henschel, L. Leal-Taixé, D. Cremers, and B. Rosenhahn. Improvements to frank-wolfe optimization for multi-detector multi-object tracking. *CoRR*, abs/1705.08314, 2017. 2, 9

[33] R. Henschel, L. Leal-Taixe, and B. Rosenhahn. Efficient multiple people tracking using minimum cost arborescences. In *GCPR*, 2014. 2, 3

[34] R. Henschel, L. Leal-Taixé, B. Rosenhahn, and K. Schindler. Tracking with multi-level features. *CoRR*, abs/1607.07304, 2016. 2

[35] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *ECCV*, 2008. 2, 3

[36] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. DeeperCut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, 2016. 2

[37] P. Ji, H. Li, M. Salzmann, and Y. Dai. Robust motion segmentation with unknown correspondences. In *ECCV*. 2014. 2

[38] J. H. Kappes, M. Speth, G. Reinelt, and C. Schnörr. Higher-order segmentation via multicuts. *Computer Vision and Image Understanding*, 143(C):104–119, 2016. 2

[39] M. Keuper. Higher-order minimum cost lifted multicuts for motion segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 6, 7

[40] M. Keuper, B. Andres, and T. Brox. Motion trajectory segmentation via minimum cost multicuts. In *ICCV*, 2015. 1, 2, 4, 6, 7, 8, 10

[41] M. Keuper, E. Levinkov, N. Bonneel, G. Lavoué, T. Brox, and B. Andres. Efficient decomposition of image and mesh graphs by lifted multicuts. In *ICCV*, 2015. 2, 3, 5

[42] S. Kim, C. D. Yoo, S. Nowozin, and P. Kohli. Image segmentation using higher-order correlation clustering. *TPAMI*, 36(9):1761–1774, 2014. 2

[43] R. Kumar, G. Charpiat, and M. Thonnat. Multiple object tracking by efficient graph partitioning. In D. Cremers, I. Reid, H. Saito, and M.-H. Yang, editors, *Computer Vision – ACCV 2014*, pages 445–460, Cham, 2015. Springer International Publishing. 2, 3

[44] L. Leal-Taix, A. Milan, I. Reid, S. Roth, and K. Schindler. Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv:1504.01942*, 2015. 2, 5, 7, 8

[45] E. Levinkov, J. Uhrig, S. Tang, M. Omran, E. Insafutdinov, A. Kirillov, C. Rother, T. Brox, B. Schiele, and B. Andres. Joint graph decomposition & node labeling: Problem, algorithms, applications. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 9

[46] J. Lezama, K. Alahari, J. Sivic, and I. Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *CVPR*, 2011. 2

[47] C. K. F. Li, A. Ciptadi, and J. Rehg. Multiple hypothesis tracking revisited. In *ICCV*, 2015. 9

[48] Z. Li, J. Guo, L. Cheong, and S. Zhou. Perspective motion segmentation via collaborative clustering. In *ICCV*, 2013. 2

[49] A. Milan, L. Leal-Taix, K. Schindler, and I. Reid. Joint tracking and segmentation of multiple targets. In *CVPR*, 2015. 2, 5, 9, 10, 11

[50] A. Milan, L. Leal-Taixé, I. D. Reid, S. Roth, and K. Schindler. MOT16: A benchmark for multi-object tracking. *CoRR*, abs/1603.00831, 2016. 1, 2, 5, 7, 8

[51] S. Müller, P. Ochs, J. Weickert, and N. Graf. Robust interactive multi-label segmentation with an advanced edge detector. In *German Conference on Pattern Recognition (GCPR)*, volume 9796 of *LNCS*, pages 117–128. Springer, 2016. 2, 10

[52] P. Ochs and T. Brox. Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In *ICCV*, 2011. 2, 6

[53] P. Ochs and T. Brox. Higher order motion models and spectral clustering. In *CVPR*, 2012. 2, 6, 7

[54] P. Ochs, J. Malik, and T. Brox. Segmentation of moving objects by long term video analysis. *IEEE TPAMI*, 36(6):1187 – 1200, Jun 2014. 2, 4, 5, 6, 7, 9, 10

[55] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille. Weakly- and semi-supervised learning of a dcnn for semantic image segmentation. *arxiv:1502.02734*, 2015. 5, 6

[56] J.-M. Pérez-Rúa, T. Crivelli, P. Pérez, and P. Bouthemy. Discovering motion hierarchies via tree-structured coding of trajectories. In *27th British Machine Vision Conference (BMVC 2016)*, York, United Kingdom, Sept. 2016. BMVA. 7

[57] E. L.-M. Pia Bideau. It's moving! a probabilistic model for causal motion segmentation in moving camera videos. In *European Conference on Computer Vision (ECCV)*, 2016. 2

[58] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR*, 2011. 2, 3

[59] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR*, 2011. 3

[60] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler, and B. Schiele. DeepCut: Joint subset partition and labeling for multi person pose estimation. In *CVPR*, 2016. 2

[61] H. Rahmati, R. Dragon, O. M. Aamo, L. V. Gool, and L. Adde. Motion segmentation with weak labeling priors. In *GCPR*, 2014. 2

[62] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 4, 6, 8

[63] A. Sadeghian, A. Alahi, and S. Savarese. Tracking the untrackable:

Learning to track multiple cues with long-term dependencies. In *ICIP*, 2017. 8, 9
[64] F. Shi, Z. Zhou, J. Xiao, and W. Wu. Robust trajectory clustering for motion segmentation. In *ICCV*, 2013. 2
[65] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Tracking multiple people under global appearance constraints. In *ICCV*, 2011. 3
[66] P. Swoboda and B. Andres. A message passing algorithm for the minimum cost multicut problem. *CoRR*, abs/1612.05441, 2016. (to appear in CVPR 2017). 2
[67] S. Tang, B. Andres, M. Andriluka, and B. Schiele. Subgraph decomposition for multi-target tracking. In *CVPR*, 2015. 1, 2, 3, 5, 10, 11
[68] S. Tang, B. Andres, M. Andriluka, and B. Schiele. Multi-person tracking by multicut and deep matching. *CoRR*, abs/1608.05404, 2016. 1, 2, 3, 4, 8, 9
[69] S. Tang, M. Andriluka, B. Andres, and B. Schiele. (Confidential title). In *CVPR*, 2017. (accepted). 9
[70] S. Tang, M. Andriluka, B. Andres, and B. Schiele. Multi people tracking with lifted multicut and person re-identification. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 9
[71] S. Tang, M. Andriluka, and B. Schiele. Detection and tracking of occluded people. *IJCV*, 2014. 2, 3
[72] Y. T. Tesfaye, E. Zemene, M. Pelillo, and A. Prati. Multi-object tracking using dominant sets. *IET Computer Vision*, 10(4):289–297, 2016. 3
[73] X. Wang, E. Turetken, F. Fleuret, and P. Fua. Tracking interacting objects optimally using integer programming. In *ECCV*, 2014. 3
[74] X. Wang, E. Turetken, F. Fleuret, and P. Fua. Tracking interacting objects optimally using integer programming. In *ECCV*, 2014. 3
[75] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *ICCV*, 2013. 4
[76] C. Wojek, S. Roth, K. Schindler, and B. Schiele. Monocular 3d scene modeling and inference: Understanding multi-object traffic scenes. In *ECCV*, 2010. 3
[77] C. Wojek, S. Walk, S. Roth, K. Schindler, and B. Schiele. Monocular visual scene understanding: Understanding multi-object traffic scenes. *IEEE TPAMI*, 2013. 3
[78] F. Yang, W. Choi, and Y. Lin. Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 8
[79] J. Yarkony, A. Ihler, and C. C. Fowlkes. Fast planar correlation clustering for image segmentation. In *ECCV*, 2012. 2
[80] S. X. Yu and J. Shi. Understanding popout through repulsion. In *CVPR*, 2001. 2
[81] A. R. Zamir, A. Dehghan, and M. Shah. GMCP-Tracker: Global multi-object tracking using generalized minimum clique graphs. In *ECCV*, 2012. 2, 3, 5, 10, 11
[82] R. Zhao, W. Ouyang, and X. Wang. Unsupervised salience learning for person re-identification. In *CVPR*, 2013. 11
[83] V. Zografos, R. Lenz, E. Ringaby, M. Felsberg, and K. Nordberg. Fast segmentation of sparse 3d point trajectories using group theoretical invariants. In *ACCV*, 2014. 2

Margret Keuper is a Juniorprofessor for Computer Vision at the University of Mannheim, Germany. Before joining the University of Mannheim, she worked as a postdoctoral researcher for the University of Freiburg and at the Max Planck Institute for Informatics in Saarbruecken. She did her Ph.D. under the supervision of Thomas Brox at the University of Freiburg.

Siyu Tang is a research group leader in the Department of Perceiving Systems at the Max Planck Institute for Intelligent Systems. She was a postdoctoral researcher at the Max Planck Instiute for Intelligent Systems, advised by Michael Black and did her PhD at the Max Planck Institute for Informatics, under the supervision of Prof. Bernt Schiele.

Bjoern Andres is Senior Researcher at the Max Planck Institute (MPI) for Informatics, a Research Group Leader at the Bosch Center for Artificial Intelligence and a Honorary Professor of the University of Tuebingen. His research is in the intersection of image analysis and discrete optimization. Before joining MPI, Bjoern worked as a Postdoctoral Fellow at Harvard University. He holds a Ph.D. in Physics from the University of Heidelberg.

Thomas Brox received his Ph.D. degree in computer science from the Saarland University in Germany in 2005. He spent two years as a postdoctoral researcher at the University of Bonn and two years at the University of California at Berkeley. Since 2010, he is heading the Computer Vision Group at the University of Freiburg in Germany. His research interests are in computer vision, in particular video analysis and learning from videos. Prof. Brox is associate editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence and the International Journal of Computer Vision. He has been an area chair for ACCV, ECCV and ICCV, and reviews for several funding organizations. He received the Longuet-Higgins Best Paper Award and the Koendrink Prize for Fundamental Contributions in Computer Vision.

Bernt Schiele received the MSc degree from the University of Karlsruhe and INP Grenoble in 1994 and the PhD degree from INP Grenoble in 1997. He was a postdoctoral associate and visiting assistant professor with MIT between 1997 and 2000. From 1999 until 2004, he was assistant professor at ETH Zurich and, from 2004 to 2010, he was full professor with TU Darmstadt. In 2010, he was appointed as a director at the Max Planck Institute for Informatics and professor at Saarland University. His main interests are computer vision, perceptual computing, wearable computers, and integration of multimodal sensor data. He is IEEE Fellow.