An Analytical Formulation of Global Occlusion Reasoning for Multi-Target Tracking

Anton Andriyenko[1] Stefan Roth[1] Konrad Schindler[2]

[1]Department of Computer Science, TU Darmstadt, Germany
[2]Photogrammetry and Remote Sensing Laboratory, ETH Zürich, Switzerland

Abstract

We present a principled model for occlusion reasoning in complex scenarios with frequent inter-object occlusions, and its application to multi-target tracking. To compute the putative overlap between pairs of targets, we represent each target with a Gaussian. Conveniently, this leads to an analytical form for the relative overlap – another Gaussian – which is combined with a sigmoidal term for modeling depth relations. Our global occlusion model bears several advantages: Global target visibility can be computed efficiently in closed-form, and varying degrees of partial occlusion can be naturally accounted for. Moreover, the dependence of the occlusion on the target locations – i.e. *the gradient of the overlap – can also be computed in closed-form, which makes it possible to efficiently include the proposed occlusion model in a continuous energy minimization framework. Experimental results on seven datasets confirm that the proposed formulation consistently reduces missed targets and lost trajectories, especially in challenging scenarios with crowds and severe inter-object occlusions.*

1. Introduction

Tracking multiple targets simultaneously – in particular tracking all relevant targets in a camera's field of view – has long been a difficult problem in computer vision [5, 19, 27]. Given an image sequence, the task is considered solved when the location of each object is known in every frame, and these locations are correctly associated across time to establish object identities. Despite significant progress, robust and reliable tracking of multiple targets is still far from being solved, especially in crowded environments. Probably the largest body of work in this area is concerned with people tracking [3, 11, 27, *etc.*], which is also the application focus here. Nonetheless, the proposed framework is generic and not limited to a specific target class.

Keeping track of a single object can be accomplished by detecting the object in each frame and "connecting the dots" to a consistent trajectory. Such *tracking-by-detection* approaches have enjoyed enormous popularity [*e.g.* 1, 6]. With multiple objects present, this becomes a lot more challenging due to the *data association* problem: in addition to localizing the targets, each of them needs to be uniquely identified – it is no longer obvious how many dots there should be, and which to connect. To resolve these ambiguities, physical constraints can be exploited as prior knowledge. For example, collisions between different targets should be avoided. The resulting complex dependencies between targets make the model energy difficult to minimize (or in probabilistic terms, the posterior hard to maximize); in a continuous setting the problem is highly non-convex [2], in a discrete setting it is in general NP-hard [*e.g.* 4].

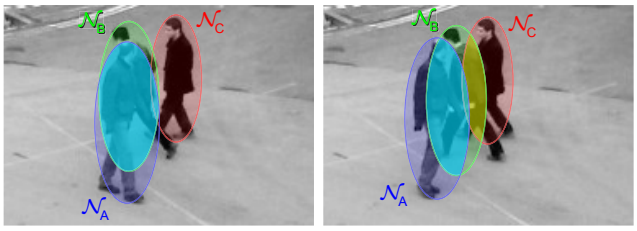

Figure 1. *(left)* Targets are represented as Gaussians in image space (red, green, blue). *(right)* Pairwise occlusions between all targets (cyan, yellow) are approximated by products of Gaussians.

A further crucial aspect of multi-target tracking is *inter-object occlusion*. In most real-world scenarios targets routinely become partially or completely occluded by other targets (and possibly also by other occluders). The occlusion results in a lack of evidence: the presence of an occluded target is not observable in the image data. However, simply treating occlusion as missing data, *i.e. ignoring the fact that the observed occluder actually predicts the lack of evidence, can heavily impair tracking performance*.

Consequently, explicit occlusion handling is important for successful multi-target tracking. Unfortunately, principled modeling of occlusion dependencies is rather tricky as the following example illustrates (see Fig. 1):

In Proc. of the *IEEE Int. Conf. on Computer Vision Workshops (ICCV Workshops)*, Barcelona, Spain, November 2011. doi:10.1109/ICCVW.2011.6130472

If target A *is at location* $\mathbf{X}_\mathsf{A}$*, then target* B *at* $\mathbf{X}_\mathsf{B}$ *is occluded; but if* A *is a bit further to the left and* B *slightly further to the right, then* B *is partially visible; however then it would partially occlude target* C*;* etc.

An explicit occlusion model thus leads to complex objective functions, which tend to be difficult and inefficient to optimize. Therefore, most previous approaches either ignore the issue altogether, or resort to some form of greedy heuristic, usually separating target localization from occlusion reasoning.

In this paper we present a global model of inter-object occlusion, which differs from previous work in several ways: *(1)* Occlusion modeling is an *integral part* of the global tracking framework: occlusions and their influence on the observation data are explicitly taken into account during trajectory estimation; nonetheless all other model assumptions (object dynamics, collision avoidance) are still enforced for all targets including those in occlusion "shadows"; *(2)* the influence of occlusions is systematically represented with closed-form functions that, moreover, are *continuously differentiable* in *closed form*, making the objective efficient and amenable to gradient-based optimization methods; *(3)* occlusion modeling is not reduced to a binary decision of whether a target is occluded or visible, but instead an *estimate of the visible portion* (fraction of the bounding box) of each target is maintained and exploited.

To the best of our knowledge such tightly integrated and accurate occlusion modeling has not been reported before in the tracking literature. To demonstrate its advantages, we present experimental results on seven different sequences, in which the proposed method consistently improves tracking accuracy. In particular, we demonstrate tracking accuracies of up to 64% on sequences from the VS-PETS benchmark, which were intended only for density estimation and considered too difficult for tracking individual targets.

2. Related Work

There is a vast body of literature on tracking, and a complete review lies beyond the scope of this paper. In the following, we focus on *visual multiple target tracking*, and on *occlusion handling*.

Multi-target tracking algorithms can be coarsely classified into two groups, *recursive* and *global*. Recursive methods estimate the current state only from the previous one and often optimize each trajectory independently. Early examples of such methods are Kalman filter approaches [*e.g.*, 5]; more recent work often uses particle filtering [14], allowing non-linear models and multi-modal posterior distributions [6, 13, 19, 22]. Global methods formulate tracking as an optimization problem where all trajectories within a temporal window are optimized jointly [3, 11, 15, 17, 26]. To render such global approaches computationally tractable, the space of possible target locations is restricted to a relatively small set of discrete points in space, either by first locating targets in each frame and then linking them, or by tracking on a discrete location grid. Leibe *et al*. [17] pose the task of detecting and linking targets as a quadratic binary program that is solved to local optimality by custom heuristics. Jiang *et al*. [15] employ integer linear programming to track multiple targets, however the number of targets needs to be known a-priori. To overcome this limitation, Berclaz *et al*. [4] introduce virtual source and sink locations to initiate and terminate trajectories. A common trait of these works is that they lead to binary optimization problems, which are usually solved to (near) global optimality by relaxing them to linear programs (LPs).

While global optimality is certainly desirable, it could so far only be achieved by simplifying the objective function such that it becomes amenable to LP-relaxation, at the cost of modeling the tracking task less faithfully. Recently, Andriyenko and Schindler [2] have shown that, in practice, local optimization of less contrived, non-convex energies can outperform formulations that focus on global optimality. In the present work we extend this approach by adding a continuous, global occlusion model that is amenable to gradient-based optimization.

Occlusion reasoning plays an important role in many areas of computer vision, including pose estimation [9, 20], object detection [10, 24], and more. In these cases, occlusion modeling improves the results for the same reason: the knowledge that the observed object is only partially (or not at all) visible predicts that less evidence will be found in the image, and the appraisal of the evidence can be adapted accordingly.

In the realm of multi-target tracking the inter-object occlusion problem has either been ignored [2, 4], or handled iteratively. Xing *et al*. [25] generate short tracklets without occlusion reasoning and then connect tracklets to longer trajectories in such a way that the connections can bridge gaps due to occlusions. Zhang *et al*. [26] propose a network flow approach, where an optimal subset of trajectories is first found with a network flow algorithm, then the trajectories are greedily extended into occluded regions.

Seriously crowded environments, where large numbers of dynamic targets and frequent occlusions make tracking difficult even for a human observer, are still only rarely processed at the level of individual targets. Notable exceptions include the work of Kratz and Nishino [16], which relies on spatio-temporal motion patterns of the crowd. Li *et al*. [18] also address crowded environments and learn tracklet associations online. Both approaches do not include any dedicated occlusion reasoning.

In the present work we propose a way of representing occlusions explicitly as part of a global tracking framework.

Symbol	Description
$\mathbf{X}$	world coordinates of all targets in all frames
$\mathbf{X}_i^t$	world coordinates of target i in frame t
$\mathbf{x}_i^t$	image coordinates of target i in frame t
(X, Y)	world coordinates on the ground plane
(x, y)	image coordinates
F	total number of frames
$F(i)$	number of frames where target i is present
N	total number of targets
$D(t)$	number of detections in frame t
$\mathbf{D}_g^t$	world coordinates of detection g in frame t
$v_i^t(\mathbf{X})$	visibility (fraction) of target i in frame t
$b(\mathbf{X}_i^t)$	minimum boundary distance (target i at t)

Table 1. Notation

Not surprisingly, taking into account occlusions directly during trajectory estimation significantly reduces the number of missed targets and lost tracks – especially in highly crowded environments.

3. Multi-Object Tracking

We formulate multi-object tracking as an energy minimization problem. Contrary to previous approaches [4, 15, 26] that restrict the state space to either non-maxima suppressed detection responses or to a discrete grid, the domain of our energy function is continuous. Although the objective function is not convex, [2] showed that in practice even a locally optimal solution can yield better results, because the formulation can be adapted to more truthfully represent the real world. Our approach extends this paradigm with a global framework for occlusion handling.

Before explaining the proposed occlusion reasoning approach in detail in Sec. 4, we first outline the non-convex energy minimization framework for multi-target tracking and the optimization scheme for finding strong local minima. Table 1 summarizes the notation.

3.1. Energy

The state vector $\mathbf{X}$ consists of (X, Y) ground plane positions of all targets in all frames. The energy function $E(\mathbf{X})$ is composed of an observation model $E_{\text{obs}}(\mathbf{X})$, which includes image evidence with explicit occlusion handling, three physically motivated constraints, and a regularization term $E_{\text{reg}}(\mathbf{X})$ to keep the solution simple:

$$E = E_{\text{obs}} + \alpha E_{\text{dyn}} + \beta E_{\text{exc}} + \gamma E_{\text{per}} + \delta E_{\text{reg}}. \quad (1)$$

A set of weights controls the relative influence of each term.

Observation model. Tracking-by-detection has emerged to be one of the most reliable approaches for tracking multiple targets, which we also adopt here. To separate targets from the background we use a sliding window linear SVM classifier. Features include HOG [8] as well as histograms of relative optic flow [23]. The basic premise of our observation model is to encourage trajectories to pass through image locations with high detector responses. To address localization uncertainty, the energy is modeled by smooth, Cauchy-like bell curves centered at the detection peaks $\mathbf{D}_g^t$ and weighted with the detector confidence ω_g^t:

$$E_{\text{obs}}(\mathbf{X}) = \sum_{t=1}^{F} \sum_{i=1}^{N} \left[\lambda \cdot v_i^t(\mathbf{X}) - \sum_{g=1}^{D(t)} \omega_g^t \frac{s_g^2}{\|\mathbf{X}_i^t - \mathbf{D}_g^t\|^2 + s_g^2} \right] \quad (2)$$

The detector confidence is obtained by fitting a sigmoid to the classifier margin, and is weighted with a simple Gaussian height prior ($\mu = 1.7\,\text{m}, \sigma = 0.7\,\text{m}$). The scale factor s_g accounts for the expected object size, and is set to 35 cm on the ground plane for people tracking. At the same time, the energy should increase if an existing target has no associated detection at all. To this end, the value $\lambda = 1/8$ is added uniformly to all targets. This penalty is only to be applied if the target is fully visible. Otherwise it should be reduced (in case of partial occlusion) or completely dropped when the object is not visible at all. We therefore scale λ according to the target's visible fraction $v_i^t(\mathbf{X})$ in the image. In Sec. 4 we describe how to compute v in closed form. For now it is important to note the dependence of the visibility on the states of *all* (other) targets.

Trajectory constraints. The remaining energy terms encode prior assumptions about the motion trajectories of the observed objects. This formulation as generic energy minimization does not constrain their form in any way (as long as they are differentiable), so that they can be chosen to suit different applications. For the present study we follow our previous work [2] and define them as

$$E_{\text{dyn}}(\mathbf{X}) = \sum_{t=1}^{F-2} \sum_{i=1}^{N} \left\| \mathbf{X}_i^t - 2\mathbf{X}_i^{t+1} + \mathbf{X}_i^{t+2} \right\|^2 \quad (3)$$

$$E_{\text{exc}}(\mathbf{X}) = \sum_{t=1}^{F} \sum_{i,j \neq i} \frac{s_g^2}{\left\| \mathbf{X}_i^t - \mathbf{X}_j^t \right\|^2} \quad (4)$$

$$E_{\text{per}}(\mathbf{X}) = \sum_{t \in \{1,F\}} \sum_{i=1}^{N} \frac{1}{1 + \exp\left(-q \cdot b(\mathbf{X}_i^t) + 1 \right)} \quad (5)$$

$$E_{\text{reg}}(\mathbf{X}) = N + \sum_{i=1}^{N} \frac{1}{F(i)}. \quad (6)$$

In a nutshell, the individual energy terms have the following effects:

- the *dynamical model* (Eq. 3) exploits the standard constant velocity assumption, which helps to rule out implausible motion patterns and to resolve ambiguities between crossing trajectories;

- the *exclusion term* (Eq. 4) avoids collisions between targets by penalizing configurations in which targets come too close to each other;

- the *persistence term* (Eq. 5) encourages uninterrupted trajectories that start and end on the boundary of the tracking area. $b(\mathbf{X}_i^t)$ is the distance of $\mathbf{X}_i^t$ to the nearest boundary and q is set to $1/350$ in all our experiments;

- finally, the *regularization term* (Eq. 6) drives the optimization towards simple solutions with fewer trajectories N that last over longer time spans $F(i)$.

3.2. Minimization

To obtain an initial set of target trajectories, we employ a conventional Extended Kalman Filter (EKF). The filter is run independently on each putative target (disregarding occlusions, collisions, and persistence), and the resulting trajectories form the starting point for energy minimization. Standard conjugate gradient descent is then employed to find local minima of the full energy function (Eq. 1).

Additionally, the optimization scheme executes jump moves to explore a larger region of the energy landscape, as suggested in [2]. Existing trajectories can be split, merged, extended, shrunk or removed entirely. Furthermore, new trajectories can be inserted around those detection responses that are not yet associated with any target. These jump moves allow to change the number of targets compared to the initialization, and to correctly handle varying numbers of targets. The different types of jumps are executed in a predefined order every 30 iterations *only if* they decrease the energy.

4. Occlusion Reasoning

In typical real-world scenarios three different types of occlusion take place: *(1)* in crowded scenes, targets frequently occlude each other causing *inter-object occlusion*; *(2)* a target may move behind static objects like trees, pillars or road signs, which are all examples of common *scene occluders*; *(3)* depending on the object type, extensive articulations, deformations or orientation changes may cause *self-occlusion*. All three types of occlusion reduce – or completely suppress – the image evidence for a target's presence, and consequently incur penalties in the observation model. Specifically, in our tracking-by-detection setting they cause the object detector to fail and thereby increase E_{obs}.

In this paper we focus on the challenge of inter-object occlusions (although it is straight-forward to extend the presented method to static scene occluders). In order to deal with situations where dynamic targets occlude each other, the main task is to overcome the difficulties which arise from the complex dependence between a target's visibility and the trajectories of several other targets that could potentially block the line of sight.

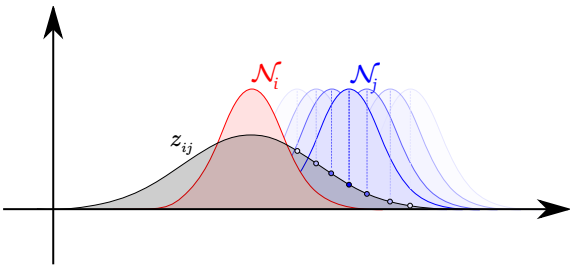

Figure 2. Schematic illustration (in 1D) of our occlusion term as a function of the occluder's position, $\mathbf{X}_j$. The relative overlap z_{ij} – the integral of the product of two Gaussians – is another Gaussian with a greater variance and can be computed in closed form.

4.1. Analytical global occlusion model

In the following we describe our approach to handle mutual occlusion between all targets with a closed-form, continuously differentiable formulation. Since this procedure is identical for each frame, the superscript t is omitted for better readability.

Relative overlap. Let us for now assume that each target i is associated with a binary indicator function

$$\mathbf{o}_i(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \text{bb}(\mathbf{X}_i) \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

which is 1 on the bounding box $\text{bb}(\mathbf{X}_i)$ of target i. The total image area of target i is thus given as $\int \mathbf{o}_i(\mathbf{x})\, d\mathbf{x}$. To compute the relative area of target i that is occluded by target j, we simply have to calculate the (normalized) integral of the product of both indicator functions:

$$\frac{1}{\int \mathbf{o}_i(\mathbf{x})\, d\mathbf{x}} \int \mathbf{o}_i(\mathbf{x}) \mathbf{o}_j(\mathbf{x})\, d\mathbf{x} \tag{8}$$

Note that we assume here that target j is in front of target i; we will address the more general case below. If we define the target visibility using the relative area as given in Eq. (8), then the visibility is not differentiable w.r.t. the object positions of $\mathbf{X}_i$ or $\mathbf{X}_j$, which precludes gradient-based optimization methods.

To address this issue we here propose to use a Gaussian "indicator" function $\mathcal{N}_i(\mathbf{x}) := \mathcal{N}(\mathbf{x}; \mathbf{c}_i, \mathbf{C}_i)$. Besides achieving differentiability, this is motivated by the fact that the shape of most objects can be well approximated by a circle or an oval (see Fig. 1 for an illustration). In our case of person tracking, each person in image space is represented by an anisotropic Gaussian with $\mathbf{c}_i = \mathbf{x}_i$ and

$$\mathbf{C}_i = \begin{pmatrix} \frac{1}{2}\left(\frac{s_i}{2}\right)^2 & 0 \\ 0 & \left(\frac{s_i}{2}\right)^2 \end{pmatrix}$$

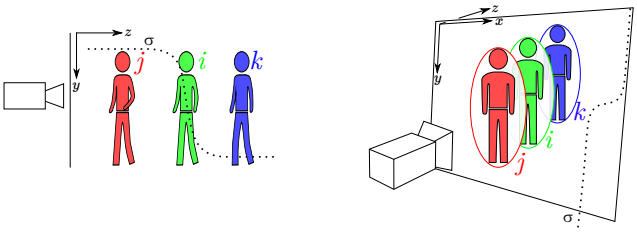

Figure 3. Target i has a non-zero overlap with j and with k. However, it is only occluded by j. Hence, the overlap is weighted with a sigmoid σ (dotted line) centered on y_i.

with s_i being the target's height on the image plane. As before, we compute the area of overlap by integrating the product of the two "indicator" functions, here Gaussians:

$$z_{ij} = \int \mathcal{N}_i(\mathbf{x}) \cdot \mathcal{N}_j(\mathbf{x}) \, d\mathbf{x} \tag{9}$$

Besides differentiability, the choice of Gaussians allows this integral to be computed in closed form. Conveniently, the integral is another Gaussian [7]: $z_{ij} = \mathcal{N}(\mathbf{c}_i; \mathbf{c}_j, \mathbf{C}_{ij})$ with $\mathbf{C}_{ij} = \mathbf{C}_i + \mathbf{C}_j$ (see Fig. 2 for a schematic illustration). Since we are interested in the *relative* overlap that corresponds to the fraction of occlusion between to targets, we compute it using the unnormalized Gaussian

$$V_{ij} = \exp\left(-\tfrac{1}{2}[\mathbf{c}_i - \mathbf{c}_j]^\top \mathbf{C}_{ij}^{-1}[\mathbf{c}_i - \mathbf{c}_j]\right), \tag{10}$$

which is differentiable w.r.t. $\mathbf{c}_i$ and $\mathbf{c}_j$ and has the desired property that $V_{ij} = 1$ when $\mathbf{c}_i = \mathbf{c}_j$. Moreover, due to the symmetric property of Gaussians we have $V_{ij} = V_{ji}$.

Depth ordering. To also take into account the depth ordering of potentially overlapping targets, we could make use of a binary indicator variable, which once again has the issue of making the energy function non-differentiable. We again replace it with a continuous, differentiable version and use a sigmoid along the vertical image dimension centered on y_i (*cf*. Fig. 3): $\sigma_{ij} = (1 + \exp(y_i - y_j))^{-1}$. Note that this definition relies on the common ground plane assumption, which implies that the depth order corresponds to the order of y-coordinates of all targets. Also note that if we assume small variation in target size, then the occluder will always appear larger than the occluded object on the image plane and hence will entirely cover the farther target if their center points share the same image location.

Visibility. To define the overall visibility of each target, we first define an occlusion matrix $\mathcal{O} = (\mathcal{O}_{ij})_{i,j}$ with $\mathcal{O}_{ij} = \sigma_{ij} \cdot V_{ij}, i \neq j$ and $\mathcal{O}_{ii} = 0$. The entry in row i and column j of $\mathcal{O}$ thus corresponds to (a differentiable approximation of) the fraction of i that is occluded by j. We can now approximate the total occlusion of i as $\sum_j \mathcal{O}_{ij}$, because in practice it is quite unlikely that the same image region is occupied by more than two targets at once. The most straightforward definition of the visible fraction of i would thus be $\max(0, 1 - \sum_j \mathcal{O}_{ij})$. However, to avoid the non-differentiable $\max$ function, we prefer to use an exponential function, and define the visibility for target i as

$$v_i(\mathbf{X}) = \exp\left(-\sum_j \mathcal{O}_{ij}\right). \tag{11}$$

This definition allows us to efficiently approximate the visible area by taking into account mutual occlusion for each pair of targets. Furthermore, by consistently using appropriate differentiable functions the entire energy has a closed-form expression and remains continuously differentiable.

Derivatives For the derivative of $v_i(\mathbf{X})$ there are two cases depending on whether the derivative is taken w.r.t. the target itself or w.r.t. any other target:

$$\frac{\partial v_i(\mathbf{X})}{\partial X_k} = \begin{cases} -\sum_j \frac{\partial \mathcal{O}_{ij}}{\partial X_i} \cdot v_i(\mathbf{X}), & k = i \\ -\frac{\partial \mathcal{O}_{ik}}{\partial X_k} \cdot v_i(\mathbf{X}), & \text{otherwise.} \end{cases} \tag{12}$$

The partial derivatives of the occlusion matrix $\mathcal{O}_{ij}$ w.r.t. the target positions can be easily derived by systematically applying product and chain rules. Note that $\mathcal{O}$ (*cf*. Sec. 4.1) is computed in image space. Our tracking model, however, is entirely defined in world coordinates. Therefore, before computing the derivatives of $\mathcal{O}$, the centroids of the Gaussians are projected onto the ground plane.

5. Experiments

Datasets. We demonstrate the validity of our approach on seven publicly available video sequences, four of which are very challenging due to high crowd density and frequent inter-object occlusions.

The latest VS-PETS benchmark from 2009 [12] consists of 15 multi-view sequences of lengths between 91 and 795 frames and offers a wide range of crowd density. The maximal number of individuals in a single frame ranges between 7 and 42. In this work we are concerned with monocular tracking and use only the first view of six video sequences. Besides the two easy sequences (*S2L1* and *S3MF1*) with medium crowd density, we also demonstrate the importance of occlusion reasoning in extremely crowded environments (*S1L1-2, S1L2-1, S2L2, S2L3*), which were recorded as a benchmark for density estimation or event recognition. Additionally, we use the *TUD-Stadtmitte* dataset [1], which consists of 179 frames of a busy pedestrian street from a low view point, making precise 3D estimation difficult.

Ground truth. To quantitatively evaluate multi-target tracking algorithms, a manually annotated ground truth is necessary. Acquiring such data is a cumbersome and expensive procedure. Usually, a lot of frames containing many targets – where each individual target has to be identified and localized precisely – are needed to achieve meaningful figures. Unfortunately, only very few datasets exist with publicly available ground truth. For our experiments we annotated several very challenging scenarios of the VS-PETS benchmark[1]. During annotation, *all* targets were marked with a bounding box (including those that are entirely occluded) and associated with a unique ID.

Evaluation metrics. There is no standard evaluation protocol for multi-target tracking algorithms. We evaluate our algorithm using the *CLEAR* metrics [21], which have become the de-facto standard. Matching is done in 3D with a 1 meter hit/miss threshold. The Multi-Object Tracking Accuracy (*MOTA*) equally combines all missed targets, false positives and identity mismatches and is normalized with the total number of targets such that 100% corresponds to no errors. The Multi-Object Tracking Precision (*MOTP*) is simply the normalized distance between the estimated and the true target locations. Additionally, we compute the metrics proposed in [18] to count the number of mostly tracked (*MT*, $\geq 80\%$) and mostly lost (*ML*, $< 20\%$) trajectories.

Implementation. We rely on a hybrid MATLAB/C implementation, for which tracking with occlusion reasoning proceeds at about 1 second per frame on a standard PC. It is likely that more optimized code or a GPU implementation could lead to near real-time performance.

5.1. Quantitative evaluation

The seven sequences used in our experiments (*cf*. Section 5) exhibit strong variations in crowd density, however the expected amount of evidence per target influences the weights of the energy terms in Eq. (1). Therefore, we split the data into two groups corresponding to *medium* and *high* crowd density. The parameters $\{\alpha, \beta, \gamma, \delta\}$ are set to {.05, 1, 2, .5} for medium and to {.1, 1, .5, .5} for high density.

Table 2 shows for each dataset the total number of targets *(GT)*, and the number of mostly tracked and mostly lost trajectories, as well as accuracy and precision. For comparison, the results obtained without explicit occlusion reasoning as well as of a simple Extended Kalman Filter are shown. Note that even our baseline without occlusion reasoning – for which we separately tuned the parameters to yield optimal results – outperforms the current state-of-the-art [2]. Still, by including the occlusion model we consistently outperform the baseline.

[1]The annotations can be downloaded from the authors' websites.

Sequence	Method	*GT*	MT	ML	MOTA	MOTP
TUD	OM	*9*	**5**	**0**	**68.6**	**64.0**
Stadtmitte	no OM	*9*	**5**	**0**	67.3	62.9
	EKF	*9*	3	**0**	58.2	58.3
PETS	OM	*23*	**20**	**1**	**88.3**	75.7
S2L1	no OM	*23*	19	**1**	85.1	75.8
	EKF	*23*	9	**1**	68.0	**76.5**
PETS	OM	*7*	**7**	**0**	**96.3**	**84.1**
S3MF1	no OM	*7*	**7**	**0**	95.1	82.9
	EKF	*7*	2	**0**	69.9	81.8
mean	OM	*13.0*	**10.7**	**0.3**	**84.4**	**74.6**
	no OM	*13.0*	10.3	**0.3**	82.5	73.9
	EKF	*13.0*	4.0	**0.3**	64.5	72.2

Table 2. Results for each dataset of the group with medium crowd density. OM: proposed method with occlusion modeling, no OM: same method without occlusion modeling, EKF: Extended Kalman filter baseline.

Sequence	Method	*GT*	MT	ML	MOTA	MOTP
PETS	OM	*75*	**25**	**8**	**60.2**	60.5
S2L2	no OM	*75*	20	14	55.2	**61.5**
	EKF	*75*	2	32	28.6	60.3
PETS	OM	*44*	**10**	**20**	**43.8**	66.3
S2L3	no OM	*44*	8	21	42.9	**68.3**
	EKF	*44*	1	35	20.4	63.3
PETS	OM	*36*	**20**	**7**	**64.1**	**67.5**
S1L1-2	no OM	*36*	18	10	62.1	65.4
	EKF	*36*	3	17	34.6	63.2
PETS	OM	*43*	**7**	**22**	**29.3**	**59.8**
S1L2-1	no OM	*43*	6	26	29.0	58.2
	EKF	*43*	0	37	6.3	58.3
mean	OM	*49.5*	**15.5**	**14.2**	**49.4**	**63.5**
	no OM	*49.5*	13.0	17.8	47.3	63.3
	EKF	*49.5*	1.5	30.2	22.5	61.3

Table 3. Results for the crowded sequences. Note the significant improvement in "mostly tracked" and "mostly lost" trajectories when using occlusion reasoning (OM). While tracking occluded targets may in some cases slightly impair precision, it always yields better accuracy by finding more targets.

It is important to bear in mind that improvements over the baseline can only be expected when a target is significantly ($> 25\%$) occluded in a frame. In the *medium* group, these constitute only 11% of all instances, such that the increase in accuracy is in fact quite significant.

The importance of explicit occlusion reasoning becomes more prominent for the *dense* group of sequences, which are characterized by frequent occlusions and high crowd density (*cf*. Table 3). In this set, 34% of all target instances are heavily occluded ($> 50\%$). We again improve the accuracy in each dataset between 1 and 10%, and are able to fully or partially track most targets even in extremely crowded sequences, which were originally intended only for estimating the crowd density and the number of people. The number of fully recovered trajectories rises more than 20% on average,

Crowd Density	Method	Detection Rate	FA Rate
low / medium	HOG+HOF	81.71 %	1.61
	no OM [2]	88.34 %	**0.27**
	with OM	**91.82 %**	0.28
high	HOG+HOF	41.39 %	4.01
	no OM [2]	44.30 %	**1.21**
	with OM	**48.41** %	1.53

Table 4. Detection and false alarm rates.

and by 30% in the best case (*S2L2*). Many trajectories are still lost in *S1L2-1* because more than 20% of all targets are occluded most of the time (*i.e*. occlusion is $> 50\%$ for more than half of their total life span). In this setting, the state-of-the-art pedestrian detector only achieves 19% recall. By integrating the proposed occlusion model we are able to detect 29% of all pedestrians. Note that the EKF tracker fails completely in such scenes, as it constantly loses track of targets disappearing into occlusion.

The recall and the number of false alarms per frame for our full-body detector [23], our tracking framework without occlusion reasoning [2] and our proposed method with explicit occlusion modeling are summarized in Table 4. The figures are computed using the standard bounding box intersection over union criterion. Note that although occlusion reasoning imposes a slightly higher false alarm rate, the $\approx$10% gain in recall corresponds to a much larger number of recovered targets.

5.2. Qualitative results

Figure 5 shows example results. Targets that are tracked with our algorithm, but lost without explicit occlusion reasoning are highlighted with yellow bounding boxes. Note the substantial number of newly tracked targets with occlusion reasoning throughout all crowded scenes. False alarms (red boxes) are due to either persistent, false detector responses (*e.g*., frame 215, 4^{th} row) or to inaccurate target localization (*cf*. frames 205 and 229, 4^{th} row).

The impact of explicit occlusion reasoning on entire trajectories is shown in Figure 4. Here, targets that were mostly lost without explicit occlusion handling but partially recovered with our method are rendered with red lines. Similarly, blue lines show the change from partially to mostly tracked trajectories. Note that due to different assignments of targets to ground truth, trajectories may also deteriorate with occlusion reasoning (dashed lines).

6. Conclusion and Future Work

We presented a model for global occlusion reasoning and its applications to multi-target tracking. Contrary to previous approaches, we model occlusions with analytical functions that are continuously differentiable in closed form, which makes global occlusion computations efficient and suitable for gradient-based optimization. Moreover, our formulation maintains an estimate of the visible portion of each object and relies on occlusion reasoning as an integrated part of a global tracking framework. Consequently, occlusions are taken into account during tracking and not only as part of a post-processing step. We quantitatively evaluated our approach on several difficult datasets, which in part were originally designed for crowd density estimation and event recognition. Our results show a consistent, significant performance improvement from explicit occlusion handling, especially in crowded scenes.

In future work we plan to investigate appearance modeling in combination with occlusion reasoning. Furthermore, we would like to utilize more specific body part detectors to capture more targets.

References

[1] M. Andriluka, S. Roth, and B. Schiele. Monocular 3D pose estimation and tracking by detection. In *CVPR*, 2010.

[2] A. Andriyenko and K. Schindler. Multi-target tracking by continuous energy minimization. In *CVPR*, 2011.

[3] J. Berclaz, F. Fleuret, and P. Fua. Robust people tracking with global trajectory optimization. In *CVPR*, 2006.

[4] J. Berclaz, F. Fleuret, and P. Fua. Multiple object tracking using flow linear programming. In *Winter-PETS*, 2009.

[5] J. Black, T. Ellis, and P. Rosin. Multiview image surveillance and tracking. In *Motion&Video Computing Workshop*, 2002.

[6] M. Breitenstein, F. Reichlin, and L. Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *ICCV*, 2009.

[7] M. Brookes. *The Matrix Reference Manual*, 2005.

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

[9] M. Eichner and V. Ferrari. We are family: Joint pose estimation of multiple persons. In *ECCV*, 2010.

[10] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. In *CVPR*, pages 990–997, 2010.

[11] A. Ess, B. Leibe, K. Schindler, and L. Van Gool. Robust multi-person tracking from a mobile platform. *IEEE TPAMI*, 31(10):1831–1846, 2009.

[12] J. M. Ferryman and A. Shahrokni. PETS2009: Dataset and challenge. In *Winter-PETS*, 2009.

[13] J. Giebel, D. Gavrila, and C. Schnörr. A Bayesian framework for multi-cue 3D object tracking. In *ECCV*, 2004.

[14] M. Isard and A. Blake. CONDENSATION – Conditional density propagation for visual tracking. *IJCV*, 29(1), 1998.

[15] H. Jiang, S. Fels, and J. J. Little. A linear programming approach for multiple object tracking. In *CVPR*, 2007.

[16] L. Kratz and K. Nishino. Tracking with local spatio-temporal motion patterns in extremely crowded scenes. In *CVPR'10*.

[17] B. Leibe, K. Schindler, and L. Van Gool. Coupled detection and trajectory estimation for multi-object tracking. In *ICCV*, 2007.

[18] Y. Li, C. Huang, and R. Nevatia. Learning to associate: HybridBoosted multi-target tracker for crowded scene. In *CVPR*, 2009.

[19] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and

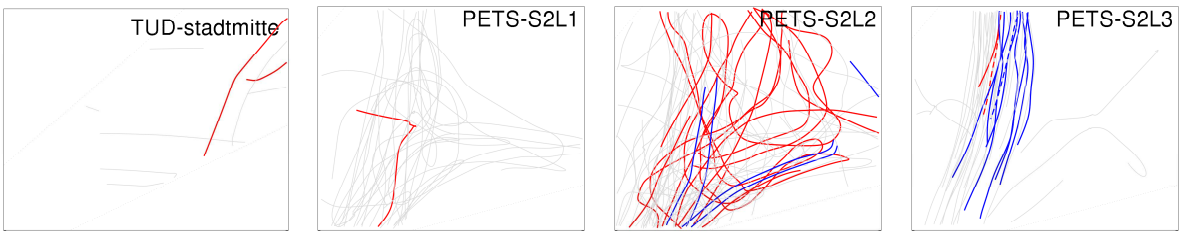

Figure 4. Comparison of tracking with and without occlusion reasoning (except for the occlusion model, the employed trackers are identical). For trajectories shown in gray, both methods yield the same result. Trajectories shown in solid blue improve from *mostly lost* to *partially tracked* with occlusion reasoning, those in dashed blue degrade from *partially tracked* to *mostly lost*. In the same way, solid red denotes an improvement from *partially* to *fully tracked* when using occlusion reasoning, and dashed red denotes the opposite change.

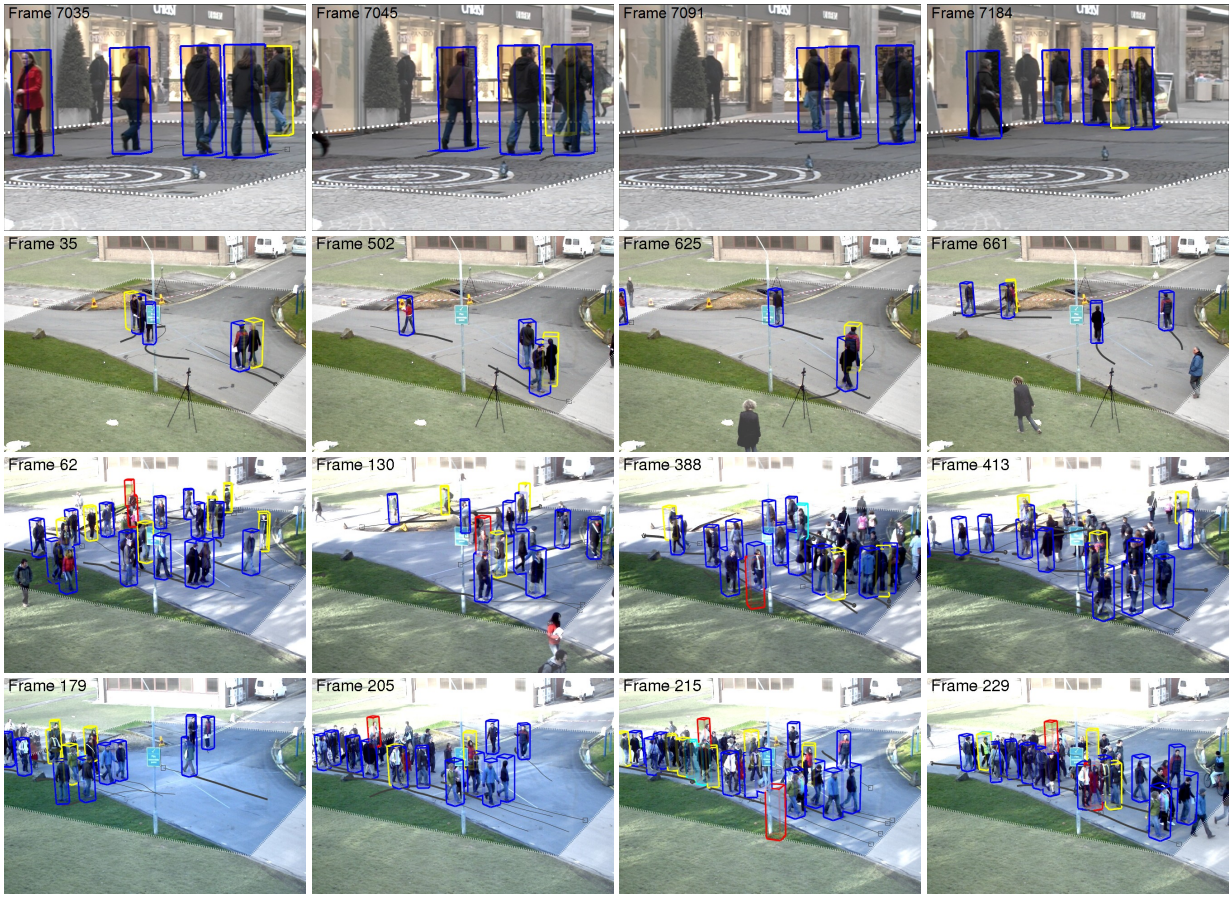

Figure 5. Tracking results on the datasets (top to bottom): *TUD-Stadtmitte, PETS S2L1, S2L2, S2L3*. Four sample frames for each dataset show targets found only without occlusion reasoning (cyan), only with occlusion reasoning (yellow), and with both methods (blue). Red boxes depict false positives. The background outside the tracking area is grayed out.

D. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*, 2004.

[20] L. Sigal and M. Black. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In *CVPR*, 2006.

[21] R. Stiefelhagen, K. Bernardin, R. Bowers, J. S. Garofolo, D. Mostefa, and P. Soundararajan. The CLEAR 2006 evaluation. In *CLEAR*, 2006.

[22] J. Vermaak, A. Doucet, and P. Perez. Maintaining multi-modality through mixture tracking. In *ICCV*, 2003.

[23] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *CVPR*, 2010.

[24] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by Bayesian combination of edgelet part detectors. In *ICCV*, 2005.

[25] J. Xing, H. Ai, and S. Lao. Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses. In *CVPR*, 2009.

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

[27] T. Zhao and R. Nevatia. Tracking multiple humans in crowded environment. In *CVPR*, 2004.