MULTIPLE TARGET TRACKING AND IDENTITY LINKING UNDER SPLIT, MERGE AND OCCLUSION OF TARGETS AND OBSERVATIONS

José C. Rubio, Joan Serrat and Antonio M. López

Computer Vision Center and Comp. Science Dept., Universitat Autònoma de Barcelona, 08193 Cerdanyola, Spain

Keywords: Tracking, Graphical models, MAP inference, Particle tracking, Live cell tracking, Intelligent headlights.

Abstract: Multiple object tracking in video sequences is a difficult problem when one has to simultaneously deal with the following realistic conditions: 1) all or most objects share an identical or very similar appearance, 2) objects are imaged at close positions so there is a data association problem which becomes worse when the number of targets is high, 3) the objects to be tracked may lack observations for a short or long interval, for instance because they are not well detected or are being temporally occluded by another non-target object, and 4) their observations may overlap in the images because the objects are very near or the image results from a 2D projection from the 3D scene, giving rise to the merging and subsequently splitting of tracks. This later condition poses the additional problem of maintaining the objects identity when their observations undergo a merge and split. We pose the tracking and identity linking problem as one of inference on a two-layer probabilistic graphical model and show how can it be efficiently solved. Results are assessed on three very different types of video sequences, showing a turbulent flow of particles, bacteria growth and on-coming traffic headlights.

1 INTRODUCTION

In the context of multiple target detection and tracking the following definitions will help us to state the goal. A target or object is some real moving entity, imaged in a video sequence, that we want to follow in order to analyze its motion for some purpose (like people and vehicles for surveillance (Benfold and Reid, 2011), particles in a turbulent flow for its characterization, live micro-organisms for lineage studies (Liu et al., 2009), (Li et al., 2007), or insects for behaviour studies (Laet et al., 2011). An observation or measurement is the detection of an object as it appears in an image. Note that a single observation can actually result from several objects whose observations overlap.

Data association is the process of relating objects to observations. In the absence of merges/splits, each target corresponds to a unique observation, and therefore targets are unambiguously identified as long as the track construction is correct. In presence of occlusions, mapping targets and observations is a difficult problem to solve. Moreover, tracking multiple objects implies multiple object interactions and mapping between observations, which is costly to solve optimally.

There are many works on visual multiple target tracking. Only some of them try to maintain identities in addition to build tracks and, being the most interesting type of result, we will focus on them in the following review. The usual classification of past works we have found is according to the strategy or the techniques employed for data association, that is, whether they are based on multiple hypothesis tracking (MHT) (Reid, 1979), joint probabilistic density association (JPDA) (T.Fortmann et al., 1983), particle filtering (Khan et al., 2003), integer linear programming, graph algorithms (like min-cut and set cover), inference on Bayesian networks (Nillius et al., 2006), etc. Being MHT and JPDA the most widely used approaches, they present some drawbacks. As MHT suffers from state space explosion when applied to real videos, JPDA assumes a fixed number of targets, and only considers measurements in the current frame step.

Another relevant categorization criterion is whether the tracking is batch (Zheng Wu and Betke, 2011), (Nillius et al., 2006) or online (Benfold and Reid, 2011), that is, tracks (and identities) are resolved once the whole sequence is available or it is done as each frame is ready. Clearly, the batch strategy has the advantage of working with all the data along time and it makes sense to use it in problems which do not require an online answer like live cell tracking or turbulent flow analysis. However,

Rubio J., Serrat J. and López A. (2012).
MULTIPLE TARGET TRACKING AND IDENTITY LINKING UNDER SPLIT, MERGE AND OCCLUSION OF TARGETS AND OBSERVATIONS.
In *Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods*, pages 15-24
DOI: 10.5220/0003710600150024
Copyright © SciTePress

in other applications a fast answer is needed to make a decision, like in surveillance or headlights control (Rubio and Serrat., 2010).

We believe that a better understanding of the state of the art can be grasped on the basis of the actual multiple target tracking problem being solved in each case. We mean that by just slightly changing the way the targets or the observations are assumed to evolve along time, or the (often implicit) relationships between a target and its observation (how may it appear in the image), one gets a very different problem to solve. This in turn determines the kind of methods to use. Just as an example, if targets are perfectly segmented (no false positives or negatives, each target gives rise to exactly one observation and to each observation corresponds one target) we have a purely problem of data one-to-one association which can be solved by the Hungarian method (Kuhn, 1955). However, if one target may be over-segmented into several regions and we want to be aware of it, the problem is quite different.

The different tracking scenarios can vary from the simplest case (one target is one measure, and one measure is one target), to more complicated situations. In the most general case, a target can produce 0,1, or more measurements, and one measurement can be produced by 0, 1 or many targets. Table 1 presents different scenarios regarding the evolution of targets in time. In order to unequivocally define our tracking application, we analyze both the behavior of our targets in time and its relationships with the image measurements. Table 2 presents these relationships for the different sequences we provide in the experiments: Synthetic flow in FIg. 1, Vehicle headlights in Fig 2 and Bacteria growht in Fig. 3.

Instead of designing a tracking method for a specific instance of a problem, our goal is to provide a generic multiple target tracking algorithm that can handle as many of those situations within a unique framework.

1.1 Overview of the Approach

We propose a two-component algorithm that outputs the complete trajectories of each of the targets in a video sequence. The first component handles the creation of tracklets within a local window of frames, and the other performs tracklet linking and data association. The Tracklet Creation is based on examining a window of a few frames, and establishing correspondences between the observations in each of these images. We define a tracklet as an ordered list of observations of the same target, between frames j and l, generated by a series of one-to-one associations between observations in consecutive frames.

Table 1: The five possible scenarios regarding the evolution of a track along time.

	Targets	
	t	$t+1$
(1). New targets may appear	0	1
(2). Targets can disappear	1	0
(3). Regular case	1	1
(4). A target can become n (e.g cell mitosis)	1	n
(5). m targets can become one (e.g cell fusion)	m	1

Table 2: Definition of our application tracking problem. Relationship target-observation, combined with the evolution in time of the targets.

	Target Scenario	Targets	Obs.
Synthetic Flow	(1),(2),(3)	1 n	1 1
Headlights	(1),(2),(3),(4)	1 1 1 m	0 1 n 1
Bacteria	(3),(4)	1	1

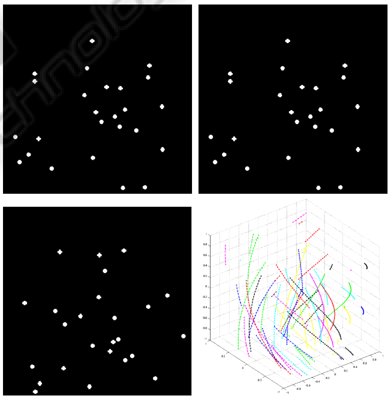


Figure 1: Sample frames of particles in a synthetic helical flow and flow lines. Each particle is always seen as a blob and one blob corresponds to one or several particles, if they overlap. Blobs merge and split but don't get occluded by other things.

This work is focused on tracking multiple targets which seldom produce any appearance information, or such information is useless because every target looks the same (See Figures 1-3, for samples of the applications' frames). This is an important handicap when addressing a problem of data association. We overcome this disadvantage by exploiting instead, the target's motion information, as well as assuming certain *rigidity* on the movement of the targets between contiguous frames. Graph Matching provides a per-

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

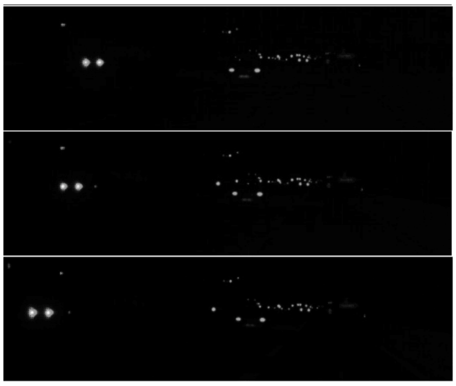


Figure 2: Successive frames from a night driving video sequence recorded by an on-board camera. One blob may correspond to two far away light sources or reflections. Blobs merge, split and get occluded by other vehicles, trees and fences.

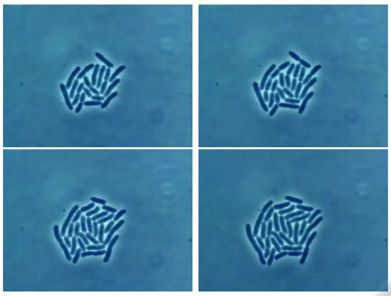


Figure 3: Sample frames of the bacteria growth video sequence. Every bacterium will be correctly segmented and to each region will correspond a single bacterium (perfect detection and no overlapping). Some bacteria divide (splits) while others just grow.

fect tool to encode this knowledge. We can see each of the frames as a graph, were every observation corresponds to a node in the graph, and it is represented by its centroid position in the image. To create the set of tracklets for a certain window of frames, we perform matching of these graph representations between every pair of consecutive frames in the window.

The Track Idenity Linking step has two main goals. First, finding the identity of the target of those observations presenting uncertainty or ambiguity about the identity of its corresponding target (data association). Second, linking tracklets of different windows which belong to the same target. We simultaneously solve these two problems by modeling the tracklet identity ambiguities in what we call an Hypothesis Graph, and then inferring the most likely hypothesis of track-target correspondences.

2 CONSTRUCTION OF LOCAL TRACKLETS

We present a probabilistic-based graph matching approach to construct target tracklets in a window of frames. Let w be the number of frames in a certain temporal window of the video sequence. We denote by $I_1, I_2, ...I_w$ the different frames within this window. Each frame contains a set of zero or more observations, indexed by $p, q, ...$. An association a is an ordered pair of observations from the same target, but at different frames. Let A be the set of all such associations,

$$A = \{a = (p,q) | p \in I_i, q \in I_j, 1 \leq i < j \leq w\}, \quad (1)$$

where $a, b, ...$ index the elements of A, so that we can denote all pairs of association without repeated combinations as $(a,b), a < b$. Let $\mathbf{X} = (...X_a...)$ be the vector of binary variables, one per association, where $X_a = 1$ if the corresponding association a exists, and zero otherwise. In the same way, the vector of all measurements is denoted by $\mathbf{Y} = (...Y_a...)$, where each association $a = (p,q)$ is represented by $Y_a = [p_x, p_y, q_x, q_y]$. Thus, each association is attributed with the spatial coordinates of its origin and destination points. Although, other properties may be also considered, like size, shape, or intensity measures.

Our goal is to find the most likely configuration of the set $\mathbf{X}$ of association states, given the set of all measurements $\mathbf{Y}$. This is, to find the maximum a posteriori estimation,

$$\mathbf{X}^* = \arg\max_{\mathbf{X}} \mathbf{p}(\mathbf{X}|\mathbf{Y}). \quad (2)$$

In a Bayesian framework, the posterior probability of the hidden variables $\mathbf{X}$, given the measurements, is proportional to the product of the likelihood and prior terms

$$p(\mathbf{X}|\mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{X}) p(\mathbf{X}). \quad (3)$$

The likelihood term $p(\mathbf{Y}|\mathbf{X})$ encodes the observation model. The prior $p(\mathbf{X})$ encodes certain constraints on the generation of tracklets. The next two sections detail how do we define and compute these two terms.

2.1 Observation Model

To build a generic observation model, we start by enumerating the set of premises that every Multiple Target Tracking scenario should satisfy:

- Measurements belonging to the same track can not move too far between consecutive frames.

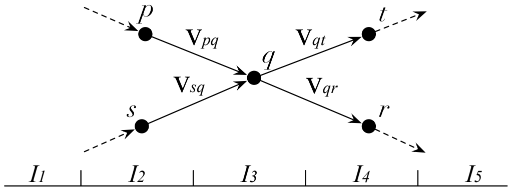


Figure 4: Target motion vectors involved in a two targets merging and splitting.

- Targets follow fairly smooth trajectories with constant speed between consecutive frames.
- Close targets in colliding directions are likely to merge.
- A target entrance and departure strongly depends on its location in the image.

We encode the first three assumptions in the following likelihood factorization. The fourth constraint will be modeled in the data association step, as we will explain later.

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{a \in A} p_A(Y_a|X_a) \cdot \prod_{(a,b) \in N} p_N(Y_a, Y_b|X_a, X_b)$$

The first term models the likelihood of an association being active or inactive, depending on the location of the two features (p,q) involved in each association $a \in A$. The second term, defined over the set N of pairs of associations, is the likelihood of two associations existing simultaneously. This pairwise terms smooths the object motion (speed and direction) along several frames, and also models the likelihood of merging and splitting events. See Figure 5.

Following we present the probabilistic modeling of each of the likelihood terms based on the previously stated assumptions.

Displacement. The likelihood of a single association is defined as:

$$p_A(Y_a|X_a) = \mathcal{N}(|\mathbf{v}_{pq}|, \mu_A, \sigma_A), \tag{4}$$

where $\mathcal{N}$ is as a Normal distribution, defined on the norm of the vector $\mathbf{v}_{pq}$, or the target velocity.In order to define our observation model as generic as possible we do not establish any correlation between the movement of a target and its appearance or image position. Although in the context of a specific application it would be convenient to apply constraints more complex and discriminative.
The pairwise term of the likelihood is, in turn, factorized in three different terms: p_L, p_M and p_S. The first penalizes sudden changes on speed and direction, and the other two model the likelihood of two targets merging and splitting.

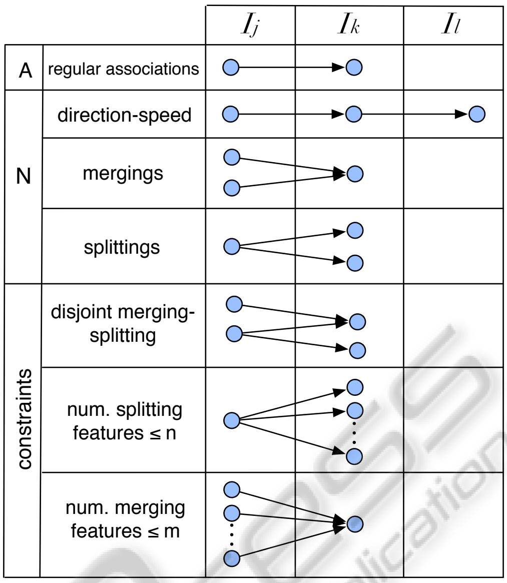


Figure 5: Sets of associations involved in the likelihood (A, N) and prior (constraints).

Linear Trajectories and Speed. The set of pairs of associations related to the trajectory of the tracks is defined as

$$N_L = \{(a,b) \in N | a = (p,q), b = (q,r)\}. \tag{5}$$

and its pairwise likelihood is defined as a mixture of densities,

$$\begin{aligned} p_L(Y_a, Y_b|X_a, X_b) = {} & \alpha \mathcal{N}(\widehat{\mathbf{v}_{pq}\mathbf{v}_{qr}}, \mu_{dir}, \sigma_{dir}) \qquad (6) \\ & + (1-\alpha)\mathcal{N}(|\mathbf{v}_{pq}| - |\mathbf{v}_{qr}|, \mu_{vel}, \sigma_{vel}). \end{aligned}$$

where the parameter $\alpha \in [0,1]$ weights the contribution of each component. The first Normal distribution models inter-frame target direction changes in terms of angles between consecutive motion vectors. The second density encodes the changes in target velocity, which are expected to be near zero, always between consecutive frames. Figure 4 shows a simple example of target motion vectors.

Merging & Splitting. The following densities model the probability of two features merging, or one feature splitting in two. Their respective sets of pairs of associations are:

$$\begin{aligned} N_M = & \{(a,b) \in N | a = (p,q), b = (s,q)\}. \qquad (7) \\ N_S = & \{(a,b) \in N | a = (q,t), b = (q,r)\}. \end{aligned}$$

Their pairwise densities define a correlation on the direction and distance between merging or splitting features. In this case, no assumptions can be made about the data following a Gaussian distribution. Instead, we use a Kernel Density Estimator to model the functions $\hat{f}_M$, and $\hat{f}_S$ from training data.

SciTePress Science and Technology Publications

$$p_M(Y_a,Y_b|X_a,X_b) = \hat{f}_M(\widehat{\mathbf{v}_{pq}\mathbf{v}_{sq}}, |\overrightarrow{ps}|) \quad (8)$$

$$p_S(Y_a,Y_b|X_a,X_b) = \hat{f}_S(\widehat{\mathbf{v}_{qt}\mathbf{v}_{qr}}, |\overrightarrow{tr}|) \quad (9)$$

2.2 Hard Constraints

We include a constraint on the maximum number of features to which one feature can be associated. This may be used in tracking applications for which we know the bounds on the number of features involved in splits and merges. Given two frames I_i, I_j, from a window of length w, we define what we call the multi-assignment m-to-n constraint as

$$\sum_{a \in A(p)} X_a \leq m, \forall p \in I_i, i = 1 \dots w-1 \quad (10)$$

$$\sum_{b \in B(q)} X_b \leq n, \forall q \in I_j, j = 2 \dots w, \quad (11)$$

where $A(p)$ is the set of associations leaving feature $p \in I_i$ and $B(q)$ the set of those arriving at $q \in I_j$.

Split and merge handling gives rise to an additional constraint to avoid bizarre tracklet configurations, like a merge mixing with a split and vice versa. See Figure 5 (disjoint merging-splitting). This takes the form

$$X_a + X_b + X_c \leq 2, \quad (12)$$

where a, b, c are the three associations involved in the joint merging-splitting.

Note that all the constraints of Eqs. (10) - (12) have the form of an upper bound on a linear combination of a few association variables. Thus, if r is the number of constraints, all of them can be compactly expressed as $C\mathbf{X}^T \leq \mathbf{b}$, where $C = [\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_r]^T$ is a very sparse binary matrix whose rows select the variables of each constraint, and $\mathbf{b}$ is a column vector with bounds m, n, and 2. Then, the prior reduces to

$$P(\mathbf{X} = \mathbf{x}) = \begin{cases} 1 & \text{if } C\mathbf{x} \leq \mathbf{b} \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

3 ONLINE DATA ASSOCIATION

In the following section we introduce the second major contribution of this work, consisting of a probabilistic method to adress the data association problem. Given a set of tracklets generated in the previous step, the goal is to find the most probable one-to-one correspondences between tracklets and track identities. Some of the tracklet identities can be unambiguously determined if they do not interact with any other tracklet along their lifetime. Unfortunately, in a context with a great amount of targets it is less likely to find tracklets which do not interfere with each other.

Lets assume that a set of tracklets $\{t_1, t_2, ..., t_n\}$ is constructed up to frame s. Each tracklet is, in turn, a list of contiguous observations between two frames. In the present context, an observation or measurement is defined simply by the feature centroid in image coordinates, as $o_a \in O$. Thus, a tracklet a between two arbitrary frames I_i,I_k, is formally denoted as $t_a^{i:k} = \{o_a^i, o_a^{i+1}, ..., o_a^k\}$, being $i < k \leq s$. However, the measurements used to find the target identities are mainly related with the movement of the targets. We say that $M_a^k = o_a^k - o_a^{k-1}$ is the motion vector of the observation $o_a \in I_k$.

Following, we formally define the Hypothesis Graph, and introduce a probabilistic method to obtain the most likely hypothesis of track labels.

3.1 Hypothesis Graph

We define an Hypothesis Graph as an undirected graph $G = (V, E)$ over sets of vertices $V \subset O$ that represent *ambiguous* observations. The set E of graph edges contains pairs (a, b) of node indexes, and denotes dependency relationships between the graph nodes. We identify two types of dependencies: Label Smoothing, and Identity Coherence, respectively grouped in sets $E_{ls}, E_{ic} \in E$, as we will explain in Section 3.2. Figure 6 shows an example of Hypothesis Graph.

We say an observation $o_a^k \in V$, if any of the following statements is true:

- The measurement is the result of a splitting.
- The measurement comes from multiple tracks.
- The observation o_a^{k-1} was also ambiguous.
- It is the first measurement of the tracklet, and exist occluded tracklets in past frames which are candidates to be recovered.

Let $\mathbf{Z} = \{Z_1, Z_2, ..., Z_n\}$ be a vector of multidimensional random variables, each corresponding to a vertex from the Hypothesis Graph, and $\mathbf{M}$ be the set of all motion vectors. Each variable realization Z indexes one of the possible hypothesis present in its associated ambiguous observation. An Hypothesis h is defined as a set of an arbitrary number of track labels $\{l_1, l_2, ...\}$. The goal is to label each ambiguous variable with the most probable hypothesis.

We propose a similar probabilistic approach to the one presented in Section 2. The set of most likely hypothesis for each of the ambiguous measurements maximizes the posterior probability,

$$P(\mathbf{Z}|\mathbf{M}) = P(\mathbf{M}|\mathbf{Z})P(\mathbf{Z}) \quad (14)$$

where the likelihood function takes the form:

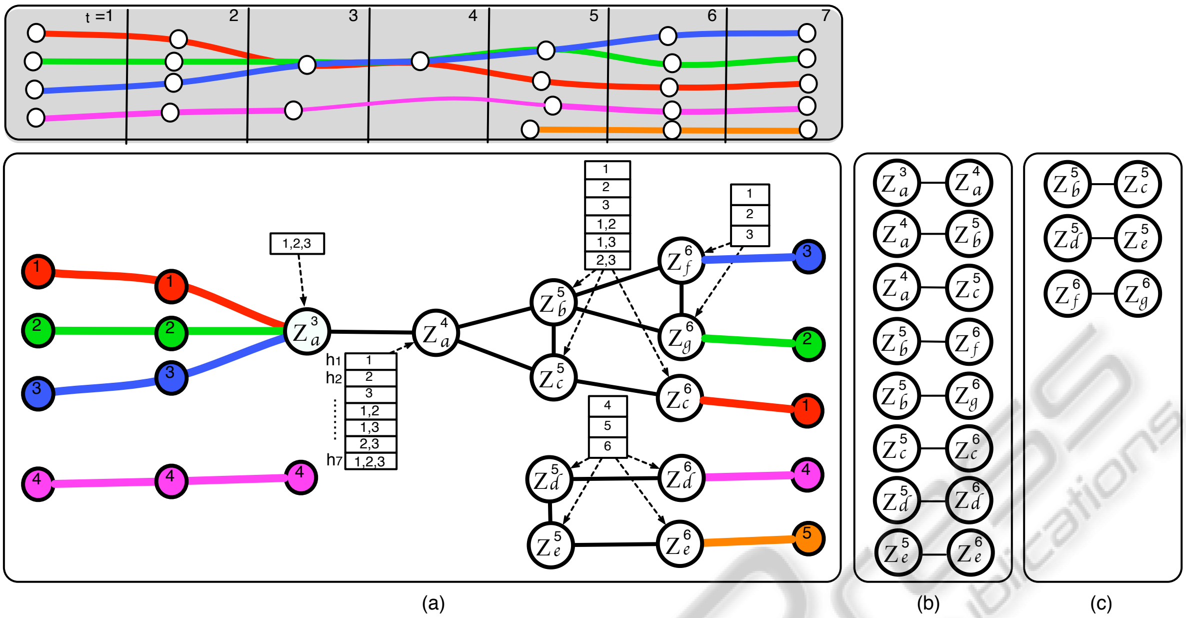


Figure 6: Example of Hypothesis Graph, in the presence of several ambiguous events. The (a) top shows 7 frames with white circles representing the observations and colored lines indicating the track each target follows. The dotted segment represents an occlusion. In (a) bottom, the Hypothesis Graph is represented. A white circle denotes a vertex of the Hypothesis Graph (an observation whose track label is unknown). The tables associated with such vertices show the list of hypothesis at each time step. In (b) the label Smoothing Dependencies are shown, and in (c) the set of Identity Coherence dependencies. Best viewed in color.

$$P(\mathbf{M}|\mathbf{Z}) = \prod_{o_a^k \in V} P(M_a^k|Z_a^k) \prod_{o_a^k, o_a^{k+1} \in E_{ls}} P(M_a^k|Z_a^k, Z_a^{k+1}). \tag{15}$$

The first likelihood term models the probability of the measurement M_a^k belonging to a track listed in any available hypothesis of Z_a^k. The second, imposes a smoothing constraint on the track label values between two contiguous observations, as well as modeling the probability of a track departure from an hypothesis. The smoothing constraint will be introduced in the next section. The first component is defined as:

$$P(M_a^k|Z_a^k) = \prod_{h \in Z_a^k} \prod_{l \in h} P(M_a^k|l) \tag{16}$$

where,

$$p(M_a^k|l) = \beta \mathcal{N}(|M_a^k - M^j(l)|; \mu_1(m), \sigma_1(m)) + \tag{17}$$
$$(1-\beta)\mathcal{N}(\widehat{M_a^k, M^j}(l); \mu_2(m), \sigma_2(m)).$$

The measurement $M^j(l)$ denotes the motion vector of the last detected observation which could be labeled with complet certainty as belonging to track l. The term encourages the selection of hypothetic tracks, whose motion vectors are similar to the original non-ambiguous track, in terms of direction and speed. The index j denotes the frame where the observation was detected, and $m = k - j$ refers to the age of the original measurement, influencing the shape of the normal distributions. This allows certain variability of the target movement depending on how long ago the last certain measurement of track l was detected. The weight β weights the contribution of the speed or the direction.

3.2 Pairwise Potentials

We define two types of dependencies:

Label Smoothing. The label smoothing dependency favors a consistent labeling of ambiguous observations along time, and encourages the generation of long tracks by smoothing the label value between contiguous observations within a tracklet (See Figure 6.(b)) Bring to mind the formulation of the likelihood of a measurement belonging to an hypothesis in Eq. (15). The Smoothing term is then defined as

$$P(M_a^k|Z_a^k = h_1, Z_a^{k+1} = h_2) = \tag{18}$$
$$= \begin{cases} 1 & \text{if } h_1 = h_2 \\ P(o_a^k|h_1,h_2)^{|h_1|-|h_2|} & \text{if } |h_1| > |h_2| \\ 0 & \text{otherwise.} \end{cases}$$

This equation enforces continuity on the track labels between contiguous observations from different frames. Note that given two hypothesis h_1, h_2 from

SciTePress Science and Technology Publications

two connected nodes, we enforce the same track labels to appear in both hypothesis (smoothing). If the newest hypothesis has fewer number of track labels, we model the probability of a track disappearing with the term $P(o_a^k|h_1,h_2)$, which is a normal distribution constructed around the assumption that tracks close to the image borders are likely to disappear. Any other configuration is considered incoherent and forbidden.

Identity Coherence. The Identity Coherence is responsible of ensuring that two or more observations in the same frame, whose hypothesis realizations can be contradictory (e.g contain the same identity), will be coherent after the inference. Since it is independent of the observations, it acts as the prior of the probability of Eq.(14):

$$P(\mathbf{Z}) = \prod_{o_a^k, o_b^k \in E_{ic}} P(Z_a^k, Z_b^k), \tag{19}$$

where

$$P(Z_a^k = h_1, Z_b^k = h_2) = \begin{cases} 1 & \text{if } h_1 \cap h_2 = \emptyset \\ 0 & \text{otherwise.} \end{cases}$$

These pairwise terms are represented in Figure 6.(c). Notice that in some tracking applications this constraint does not exist (e.g. sobresegmentation produces several measurements of one target). We allow this restriction to be dropped depending on the tracking application. Furthermore, the Identity Coherence restriction can be selectively placed to distinguish both cases: grouping measurements of the same target, and mutual-occlusions of targets.

3.3 Handling of Long Occlusions

The last important detail to complete the method formulation is the handling of long occlusions. We address this issue with a very intuitive assumption: Every observation which starts a new tracklet is a candidate to contain the identity of track which ceased being observed during the last L frames. Lets denote as T_{occ} the list of these track identities, and let Z_a^k be the observation of a new tracklet a detected in frame k. Being N the number of tracks identified up to the present frame, the set of realizations (hypothesis) of Z_a^k is defined as

$$Z_a^k = \{T_{occ} \cup \{N+1\}\}. \tag{20}$$

The Eq. (16) is then slightly modified to include the likelihood of a new track appearing in the scene:

$$P(M_a^k|Z_a^k) = \begin{cases} \prod_{h \in Z_a^k} \prod_{l \in h} P(M_a^k|l) & \text{if } l \in T_{occ} \\ P_{new}(M_a^k) & \text{if } l = N+1 \end{cases} \tag{21}$$

Note that the distribution stays unchanged if the realization of Z_a^k suggests the recovery of an occluded track in T_{occ}. Otherwise, the density P_{new} indicates the probability of detecting a new track. The distribution P_{new} is assumed normal, defined on the minimum distance between the feature centroid and the nearest image border. Analogously to the departures, the entrance of targets is more likely in the limits of the image.

4 LEARNING AND IMPLEMENTATION

All probability densities assumed Gaussian are learned from training data using Maximum Likelihood Estimation. The densities which can take an arbitrary shape are as well learned using a non-parametric method like Kernel Density Estimator. The training data is annotated manually using a software specifically developed for that purpose.

In order to infer the most likely configuration of random variable values, we construct two Markov Random Fields, each of them representing the posterior probability for one of the layers: tracklet generation and data association. The maximization of both posteriors of Eq. (2) is usually NP-Hard. To overcome this problem, we approximate the solution using the Tree Reweighed Belief Propagation, which is a message passing algorithm which infers the Maximum a Posteriori configuration of the set of variable realizations. We use a C++ implementation of the algorithm (libDAI), developed in (Mooij, 2010).

5 EXPERIMENTS AND RESULTS

We evaluate our Multiple Target Tracking algorithm in experiments on synthetic and real image sequences, and provide quantitative results of the experiments. Usually works on Multiple Target Tracking use standard metrics to evaluate the error on the prediction of the localization of the targets in each frame. A popular approach in recent works suggests the use of MOT Metrics to evaluate MTT precision and accuracy (Bernardin and Stiefelhagen, 2008). This measure takes into account four different aspects of the quality of the results:

- Precision of the hypothesis localization.
- False positive errors.
- Missed detections.
- Number of track label miss-matches.

An important difference between our experimental demonstration against other examples shown in the literature is that we do not include a detection phase in the tracking process. This means that we do not filter the objects that appear in the image, and thus we consider every observation as a potential target to track. This is justified due to the nature of the applications we are dealing with. In the synthetic scenario it is obvious that all the objects present in the images are valid targets, since we do not introduce any artificial clutter or noise. In the headlight tracking application, we threshold the intensity values of the images to discern the interesting blobs, and we track indistinctivelly every light, and every reflection, which are both present in our ground-truth as valid targets. In the last example, the bacteria growth sequence, we manually construct a perfect segmentation, which does not produce any undesired artifacts.

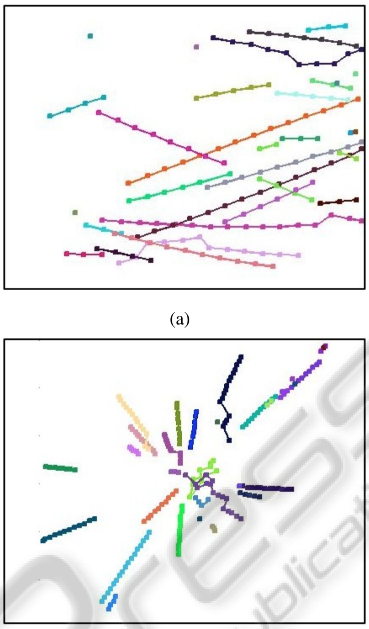
Therefore, we cannot evaluate the precision of our hypothesis, since the hypothesis location is always the same as the target location. A target cannot be miss-detected, since non-occluded targets have at least one observation, and every detection has at least one target associated, meaning that false-positives cannot occur. The only MOT component that we can use as a quality measure is the number of track label miss-matches. Moreover, we also measure the accuracy of tracklet generation using a typical feature-matching evaluation metric, by simply counting the ratio of correct matchings against the total. Table 3 shows the quantitative results for the experiments.

5.1 Synthetic Sequences

We have generated two synthetic sequences of 100 frames, each containing a number of targets imaged as a circle with a fixed radius. The sequence *A* contains an average of 10 particles per frame, the sequence *B*, 15 particles per frame. The radius of the particles is 10 and 5 for each sequence respectively. Figure 7 shows a timestamp of 20 contiguous frames of both sequences. In the first sequence the motion is achieved with a XZ projection of a 3D helical motion of targets. In the second the particles move towards a sink in the center of the image . It can be seen how the particles follow more or less linear trajectories in both cases. Each color represents a track label. Sudden changes of colors, or sharp corners along tracks, indicate a miss-match of track labels.

5.2 Tracking of Car Headlights

In the context of an Intelligent Headlight Control Application, the main problem is to classify a blob in the image as a car or a reflection, in order to automate the activation of the light beams. Usually, a complex classifier gathers features from every blob in the image, and labels them as vehicle or non-vehicle. A tracker can also be included working in parallel with the classifier (Rubio and Serrat., 2010), in order to provide additional information (e.g combining the classifier beliefs of a given target between different frames). This is the reason why, in this specific application, we are interested of tracking every blob in the image, and there is no need to perform a detection process.


(a)

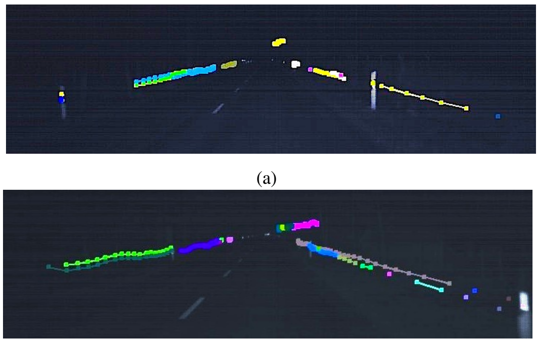
(b)

Figure 7: Timestamp of 20 frames in both synthetic tracking sequences. In (a), targets move from left to right disappearing in the right image border. In (b),targets move from the image borders towards the image center, where they disappear.

We perform multiple target tracking in two sequences of 100 frames each. Note that this scenario is specially difficult because the camera recording the images is constantly moving, since it is mounted in a car. Far away lights are represented as very tiny blobs of few pixels that are very hard to track. Classic trackers like Kalman Filter would certainly perform poorly in this situation, since measurements of distant targets are separated by a few image pixels, and the movement of the camera makes very hard to solve the data association. Moreover, there are hardly any appearance features to rely on.

5.3 Bacteria Growth

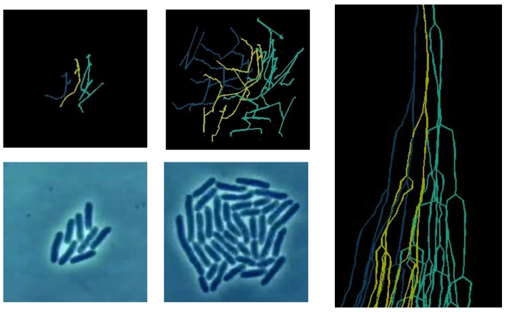
This experiment consists of tracking a growing num-



(a)

(b)

Figure 8: Representation of the target tracks in both headlights sequences. Each color represents a different label.



(a) (b)

Figure 9: In (a), bottom, two frames of the bacteria growth sequence. In (a) top, the corresponding paths. The three different colors indicate the original bacteria parent that originated the track. In (b), it is represented the lineage tree, by plotting each track's horizontal component against time.

ber of bacteria, which are continuously dividing in two. This is an example of a tracking scenario where a target can become two, and we are interested in tracking these targets at the same time that we construct what is known as the cell mitosis lineage. The sequence provided has 54 frames, reaching in the last frame a maximum of 43 targets simultaneously in the image.

In this application we slightly modify the appearance likelihood of Eq. (4), to improve the results, by profiting from the little appearance information that the targets display. We use the overlap ratio between the pixel areas the bacteria cover in consecutive frames, to determine the most likely correspondence between bacteria.

Table 3: Results for every video sequence. First column shows the total number of objects that appear in the sequence. Second column the ratio of incorrect tracklets. Third column shows number of track label miss-matches against total number of objects.

Application	N. Obj	% Trackets	% Labels
Synthetic1	36	0.12	0.16
Synthetic2	52	0.19	0.27
Headlights1	29	0.31	0.24
Headlights2	35	0.27	0.34
Bacteria	43	0	0.11

6 CONCLUSIONS AND FURTHER WORK

In this paper we have modeled the problem of Multiple Target Tracking with presence of occlusions, merges and splits, as a two stage probabilistic method. The probability densities that model the target behavior and data association are all learnt form training data. We have provided insights into the different scenarios one can find when dealing with the problem of Tracking, and we also present our model as a general solution to deal with different tracking scenarios simultaneously. We have proved the suitability of our approach in three different experiments, one synthetic and two with real images, in which we track particles presenting non or very poor appearance features. This makes it a challenging problem, mainly when addressing the data association of objects and observations.

Avenues for future research include increasing the quantity and quality of experiments, covering a wider spectrum of tracking scenarios. Moreover, introducing a detector in the model will allow our method to be applied in a large range of classic tracking applications, as well as qualifying it to follow standard evaluation metrics and benchmarks of the state of the art of the MTT.

ACKNOWLEDGEMENTS

This work was partially founded by Universitat Autònoma de Barcelona and by Spanish MICINN projects TRA2011-29454-C03-01, TIN2011-29494-C03-02, and Consolider Ingenio 2010: MIPRCV (CSD200700018).

REFERENCES

Benfold, B. and Reid, I. (2011). Stable multi-target tracking in real-time surveillance video. In *CVPR*, pages 3457–3464.

Bernardin, K. and Stiefelhagen, R. (2008). Evaluating multiple object tracking performance: the clear mot metrics. *J. Image Video Process.*, 2008:1:1–1:10.

Khan, Z., Balch, T., and Dellaert, F. (2003). An mcmc-

SCITEPRESS Science and Technology Publications

based particle filter for tracking multiple interacting targets. In *in Proc. ECCV*, pages 279–290.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97.

Laet, T. D., Bruyninckx, H., and Schutter, J. D. (2011). Shape-based online multitarget tracking and detection for targets causing multiple measurements: Variational bayesian clustering and lossless data association. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99.

Li, K., Chen, M., and Kanade, T. (2007). Cell population tracking and lineage construction with spatiotemporal context. In *Proceedings of the 10th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 295 – 302.

Liu, M., Roy Chowdhury, A., and Reddy, G. (2009). Robust estimation of stem cell lineages using local graph matching. In *MMBIA09*, pages 194–201.

Mooij, J. M. (2010). libDAI: A free and open source C++ library for discrete approximate inference in graphical models. *Journal of Machine Learning Research*, pages 2169–2173.

Nillius, P., Sullivan, J., and Carlsson, S. (2006). Multi-target tracking - linking identities using bayesian network inference. In *In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2187–2194.

Reid, D. B. (1979). An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24:843–854.

Rubio, J. C., A. L. D. P. and Serrat., J. (2010). Multiple target tracking for intelligent headlights control. In *Intelligent Transportation Systems Conference, 2010. ITSC 2010. IEEE.*

T.Fortmann, Bar-Shalom, Y., and Scheffe, M. (1983). Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, 8:173–184.

Zheng Wu, T. H. K. and Betke, M. (2011). Efficient track linking methods for track graphs using network-flow and set-cover thechniques. *in CVPR*.

SciTePress
Science and Technology Publications