

Tracking with occlusions via graph cuts

Nicolas Papadakis and Aurélie Bugeau

Abstract

This work presents a new method for tracking and segmenting along time interacting objects within an image sequence. One major contribution of the paper is the formalization of the notion of visible and occluded parts. For each object, we aim at tracking these two parts. Assuming that the velocity of each object is driven by a dynamical law, predictions can be used to guide the successive estimations. Separating these predicted areas into good and bad parts with respect to the final segmentation and representing the objects with their visible and occluded parts permits handling partial and complete occlusions. To achieve this tracking, a label is assigned to each object and an energy function representing the multi-label problem is minimized via a graph cuts optimization. This energy contains terms based on image intensities, that enable segmenting and regularizing the visible parts of the objects. It also includes terms dedicated to the management of the occluded and disappearing areas, that are defined on the areas of prediction of the objects. The results on several challenging sequences prove the strength of the proposed approach.

Index Terms

Tracking, interacting objects, occlusions, graph cuts optimization.

1 INTRODUCTION

Despite lot of attention being dedicated to this problem over the last twenty years, tracking segmented objects remains a very concerning problem in computer vision. In particular, the problem of dealing correctly with occlusions is still an open subject.

1.1 State of the art on tracking

As presented in the recent review [33], three main categories of tracking methods exist: point, kernel and silhouette tracking. Here, we only focus on the latter which aims at extracting successive segmentations of the target over time using a temporal consistency. This consistency is often obtained using optical flow estimations. The modeling of the dynamics also enables dealing with the occlusions of an object.

The silhouette tracking algorithms can be decomposed into two groups, depending on whether the silhouette is represented by a set of parameters [17], [31] or by a continuous energy function. As parametric representation does not handle well topology changes without incorporating complex shape priors as in [9], this paper only concentrates on energy function methods. In such works, the object boundary is mostly defined by the zero level set of a continuous function [12], [15], [26], [27], [29]. However, these methods suffer from a high computational cost that we would like to avoid.

Using graph cuts is one solution to accelerate the tracking process. The advantages of min-cut/max-flow optimization are its low computational cost and the fact that it converges to the global minimum without getting stuck in local minima. This kind of approach was first used for tracking in [32] where the contour of the object at previous time is dilated into a narrow band. A graph is then constructed on this band, which results in a segmentation of the object. Nevertheless, as no temporal information is included, this

method is unable to deal with large displacements and complete occlusions. Graph cuts have also been used in [11] to successively segment one object or layer through time using motion information and in [14] for kernel tracking. To our knowledge there are only two kind of works [22], [23] and [5], [6] that rely on graph cuts minimization to segment and track multiple objects whilst using the object velocity or a dynamical model. These two types of methods are based on a prediction of the target at the next instance, through a velocity estimation, followed by its correction with a graph cuts segmentation method.

In [22], [23], Malcolm et al. define a method in which the velocity of each object is modeled by an auto-regressive model to provide a prediction for the next time step. A distance to the prediction is taken into account so that the successive segmentations are spatially constrained. This model then enables the process to be quite consistent in time. To consider strong changes of motion, the authors compute, for each object, a scalar coefficient which represents the error of prediction in order to weight the influence of predictions. However, the segmentations are quite unstable in time. Moreover, it does not cope well with partial and total occlusions as there is no specific process for dealing with interacting objects.

In [5], [6], Bugeau and Pérez used external detections to help track objects. All the pixels belonging to the objects are here represented. This method can be viewed as a filtering of the tracked objects with image intensity and external observations, without any need to associate objects and detections beforehand. These detections enable the process to be robust to partial occlusions, and, if the motion of the target is simple enough, to total occlusions also. On the other hand, no dynamical model is considered, as the motion is computed independently at each time using the Lucas-Kanade motion estimator [21]. The tracking is done in two phases of graph cuts: an individual tracking of each object, and a separation (through segmentation) of the possibly merged objects.

1.2 Discussion on graph construction

In the graphs built in [22], [23] one vertex corresponds to one pixel of the image. This classic and simple representation limits the occlusion management. Namely, when an object is occluded (by the background or by another object), it can not be represented using only one vertex per pixel.

The graph representation used in [5], [6] leads to some interesting points: the vertices are not only the pixels of the image, but there is also an extra vertex for each object detection, which enforces the temporal consistency of the successive segmentations. These additional vertices allow associating external observations to the tracked objects. Thus this graph can consider information outside the pixel grid.

The idea of using additional vertices that consider the state of objects was originally proposed in [11], [18]. These works concern the estimation of disparity from stereo images and combine dynamic programming approaches with graph cuts resolution. The authors associate state segmentation (coming from the three and four-state moves algorithms of [8], [10]) with spatial labeling of the pixels. The model proposed in this paper is related to this approach, as we will consider the state of predicted pixels (good if a predicted pixel of an object belongs to the final segmentation of this object or bad otherwise) simultaneously with the object segmentations. Nevertheless, the visual representation of the graph will be closer to the one of [5], [6].

1.3 Contributions of the paper

Through this work, we seek to combine the advantages of the two kinds of previously described tracking methods [5], [6], [22], [23]. More precisely, we are interested in tracking and segmenting several (possibly interacting) objects with a fast and accurate dynamic segmentation process. In particular, we focus on the occlusion representation and management. Knowing the initialization of the objects of interest at the first frame of an image sequence, we aim at tracking them along time. No assumption about static background is made.

To realize this multi-target tracking, each object is represented as a set of pixels that can be visible or occluded. Several objects can then be associated with a single pixel of the current image, but only one object will be visible. We introduce an energy involving a temporal consistency between visible and occluded parts of the objects, through a system of predictions. The pixels predicted by the dynamics of the objects are indeed segmented in two parts and labeled as good or bad. The predicted pixels of an object that will finally belong to the segmentation of this object are considered as good predictions while the other pixels will represent the bad predicted areas.

Hence, the whole representation has the capability of dealing easily with the partial and total occlusions of the targets, while taking into account errors of prediction. Our model then fully describes what is happening during real tracking applications: appearance, disappearance and occlusions. The energy is finally minimized with a graph cuts optimization.

Despite the similarity with the energy minimized in the work of [22], [23], we would like to emphasize that the overall process took most inspiration from [5], [6] and [19]. Indeed, we first added additional vertices to the classical image graph following [5], [6]. We also adapted the principle of active vertices (good and bad predictions) as well as the binary function that models occlusions and rejects impossible labeling from [19].

1.4 Important definitions and notations

Here, we focus on multiple objects tracking. We will assume that N objects are involved and denote as $\Omega^t \subset \mathbb{R}^2$ the set of pixels at time t of the image $I(x, t)$. The image $I(x, t)$ varies spatially with the pixel $x \in \Omega^t$ and temporally with $t \in [0, +\infty)$. We will refer to the i^{th} object at time t by \mathcal{O}_i^t . Let us now consider that only a subset of the pixels of each object is visible. To that end, we define an object as follows.

Definition 1.1: An object \mathcal{O}_i^t is represented by the union of two disjoint subsets: the visible set \mathcal{V}_i^t and the occluded set $\mathcal{O}_i^t \setminus \mathcal{V}_i^t$. These two subsets form a partition of the object, so $\mathcal{V}_i^t \subset \mathcal{O}_i^t$.

Such an object representation allows dealing with occlusion, as illustrated in Figure 1.

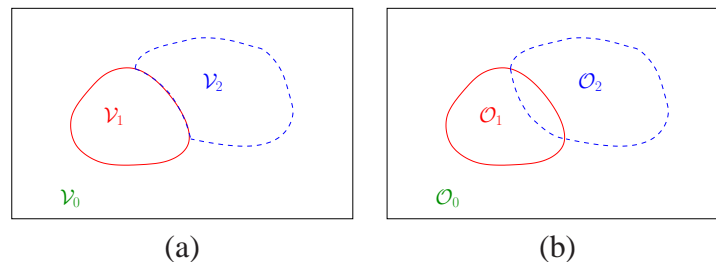


Fig. 1. Illustration of the definition of objects: The object 1 (resp. 2) is represented by the full (resp. dashed) line. (a) The visible parts of the objects (\mathcal{V}_1 and \mathcal{V}_2) and the background (\mathcal{V}_0) form a partition of the image domain Ω . (b) The whole objects area can intersect themselves in case of occlusions ($\mathcal{O}_1 \cap \mathcal{O}_2 \neq \emptyset$).

We will assume that the initial segmentation \mathcal{O}_i^0 of each object i at time 0 is known. We also suppose that the objects are initially entirely visible ($\mathcal{O}_i^0 = \mathcal{V}_i^0$). From this initial segmentation, a color distribution can be built for each object (from $I(x, 0)$ and $x \in \mathcal{V}_i^0$) and the probability $P_i(x)$ of a pixel $x \in \Omega^t$ to belong to an object i can be computed. The subscript $i = 0$ will be reserved for the background. Note that, at each time t , the visible parts of the objects and the background form a partition of the image domain:

$$\bigcup_{i=0}^N \mathcal{V}_i^t = \Omega^t \text{ and } \mathcal{V}_i^t \cap \mathcal{V}_j^t = \emptyset, \forall i \neq j.$$

We define $\mathcal{V}_0^t = \mathcal{O}_0^t$ for the background. In fact, we do not want to segment the occluded part of the background (in reality $\mathcal{O}_0^t = \Omega^t$) and we only focus on its visible part.

From the image point of view, the visible part of the objects can be represented with a labeling function $\lambda : \Omega^t \mapsto [0; N]$, that associates each pixel of the current image with an object or the background. We then have:

$$x \in \mathcal{V}_i^t \Leftrightarrow \lambda(x) = i.$$

1.5 Overview of the paper

This paper is organized as follows. We first detail two related existing methods from the literature in section 2. Next, the dynamical model and the proposed energy are presented in section 3. The discretization of the energy and the resolution by graph cuts is detailed in section 4. Some results and comparisons with the method of [22] are finally presented in section 5. The whole implementation details are given in Appendix.

2 RELATED WORKS

In this first section, we describe some state-of-the-art functionals dedicated to segmentation and tracking of objects. Many graph cuts based methods have been proposed for segmentation issues but very few works use this methodology for tracking. This section presents two works that are directly related to the proposed approach. They both consider that the whole objects are visible and do not take into account the occluding parts. In that case, for all $i = 0 \dots N$, $\mathcal{V}_i^t = \mathcal{O}_i^t$. In this section, we therefore refer to an object by its visible part.

2.1 Segmentation

A simple segmentation of the background and the N objects (inspired by the work of Boykov and Jolly [1]) can be obtained, at each time t , by minimizing the following energy with respect to the labeling function λ :

$$J(\lambda) = \mathcal{E}_D(\lambda) + \mathcal{E}_R(\lambda). \quad (1)$$

The data term \mathcal{E}_D measures the likelihood P_i of a pixel to belong to an object i :

$$\mathcal{E}_D(\lambda) = - \sum_{x \in \Omega^t} \sum_{i=0}^N \ln(P_i(x)) \delta(\lambda(x) - i),$$

where $\delta(l)$ is the characteristic function (equals to 1 if $l = 0$ and 0 otherwise). The regularization term \mathcal{E}_R is:

$$\mathcal{E}_R(\lambda) = R_\Omega \sum_{x \in \Omega^t} \sum_{z \in \mathcal{N}^l(x)} F(I(x, t), I(z, t)) [1 - \delta(\lambda(x) - \lambda(z))],$$

where $R_\Omega > 0$ is the regularization parameter and $F : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}^+$ is a decreasing function that penalizes the spatial discontinuities of the segmentation according to the image data. A cost is then paid when two neighbor pixels x and z have different labels, i.e., $\lambda(x) \neq \lambda(z)$ and $\delta(\lambda(x) - \lambda(z)) = 0$.

This regularization is equivalent to the minimization of the length of the boundaries between objects. Note that the neighborhood of a pixel x involved in energy (1) is defined by:

$$\mathcal{N}^l(x) = \{z \in \Omega^t \text{ such that } 0 < |z - x| \leq l\}. \quad (2)$$

With such a model, the segmentations obtained at each time obviously suffer from temporal inconsistencies.

2.2 Tracking with predictions

We now explain how using predictions allows enforcing the temporal consistency. To that end, we briefly review the method proposed in [22], [23], where the visible parts \mathcal{V}_i^t of the $i = 1 \dots N$ objects are tracked. In this work, the segmentation obtained at a frame t is used as a constraint for the segmentation at time $t + 1$. Assuming that the mean velocity (or a model of the mean velocity) is known for each object, the authors translate the current estimation at time t to have a prediction $\mathcal{V}_i^{t+1|t}$ of the object i at frame $t + 1$. The pixels that do not belong to the predicted set $\mathcal{V}_i^{t+1|t}$ are discouraged to be associated with the object i . This penalization is done with a new term:

$$\mathcal{E}_\gamma(\lambda) = \gamma \sum_{i=1}^N \sum_{x \in \Omega^{t+1}} d_i(x) \delta(\lambda(x) - i).$$

In this appearance term, an Euclidean distance function $d_i(x)$ to the prediction is introduced for all $x \in \Omega^{t+1}$ as

$$d_i(x) = \min_{z \in \mathcal{V}_i^{t+1|t}} |z - x|.$$

This distance (weighted by the cost $\gamma > 0$) is taken into account in order to constraint the new estimation to be in the spatial neighborhood of the prediction. A cost is then paid for the areas that do not belong to the prediction but are nevertheless segmented as objects (namely $\mathcal{V}_i^{t+1} \setminus \mathcal{V}_i^{t+1|t}$). This property makes the process able to deal with tracking problems by adding coherence on shape and position between successive temporal segmentations.

This new model completes the energy (1) as it is adapted to objects presenting a quite continuous deformation in space and time:

$$J(\lambda) = \mathcal{E}_D(\lambda) + \mathcal{E}_\gamma(\lambda) + \mathcal{E}_R(\lambda). \quad (3)$$

Note however that, in case of partial and total occlusions, there is no special model involved to recover the shape of the tracked object. Hence, in case of tracking of multiple objects, occlusions between objects can not be treated correctly as a single pixel can not belong to two different objects.

To illustrate this limitation, we applied this method to a sequence from PETS 2001¹, where we aim at tracking a truck and a pedestrian. On this sequence, the algorithm of [22] loses the pedestrian during its partial occlusion by the truck (see Figure 2). As the occluded parts of the objects are not considered, the pedestrian disappears due to the weight of the regularization term. Moreover, the tracking of the truck boils over the background and even segments another pedestrian with similar color.

Increasing the weight of the regularization term allows rejecting these last errors, as illustrated in Figure 3, but the tracking of the pedestrian is almost impossible with such a high regularization parameter. In this sequence, the only solution would be to apply independently the algorithm of [22] to the truck and to the pedestrian with different regularization parameters.

The previous experiments show the limitations of the methods only based on the tracking of the visible part of the objects. More precisely, there is no energy term dealing with the bad predicted areas $\mathcal{V}_i^{t+1|t} \setminus \mathcal{V}_i^{t+1}$ and the representation of the occluded parts of the objects $\mathcal{O}_i^t \setminus \mathcal{V}_i^t$ is not considered.

In this work, we want to deal with partial and total occlusions and also handle bad segmentations properly (reject false segmentation that may occur at one time) thanks to the motion information. To that end, we will present in the next sections a model that considers non empty intersections between objects segmentation, and explain how a dynamical model permits dealing correctly with occlusions. We will show that the temporal predictions enables modeling, regularizing and tracking both visible and occluded parts of the objects.

1. database PETS: Performance Evaluation of Tracking and Surveillance, available on <http://www.cvg.rdg.ac.uk/slides/pets.html>

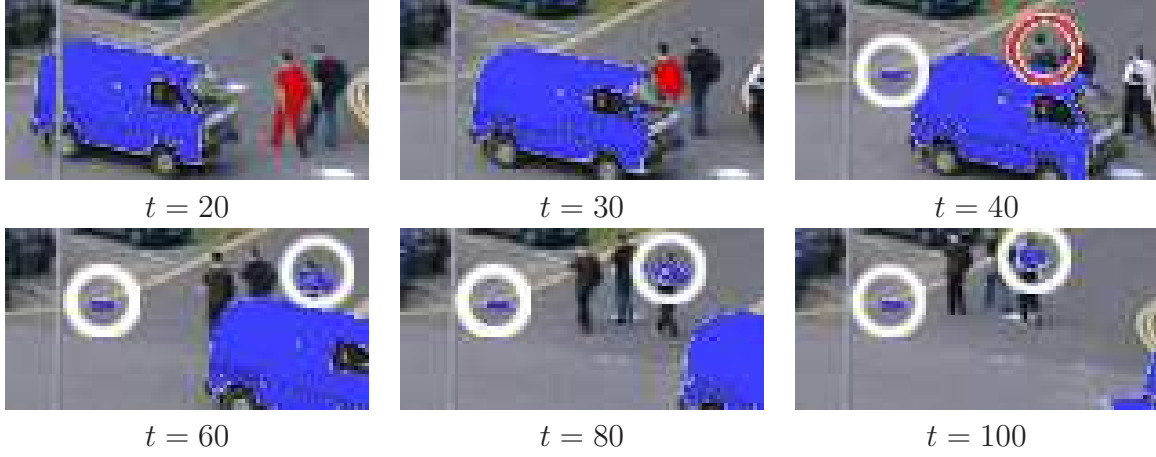


Fig. 2. **Truck and pedestrians [22]:** Only the visible part of the objects are tracked and the pedestrian is lost when partially occluded by the truck. Moreover, parts of the background and even another pedestrian are finally segmented as truck. In this example $\gamma = 1$ and $R_\Omega = 10$.

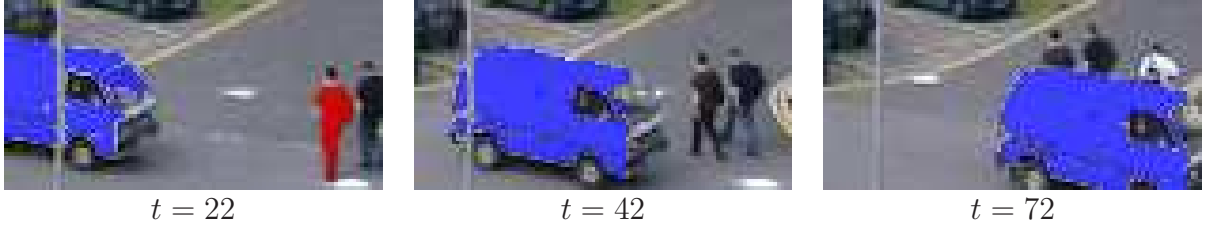


Fig. 3. **Truck and pedestrians [22]:** Increasing the weight of the regularization term ($R_\Omega = 15$) allows removing some artifacts. However, as the visible part of the pedestrian is too small, its tracking is lost even before the partial occlusion. In this example $\gamma = 1$.

3 DEFINITION OF THE PROPOSED TRACKING MODEL

In this section, a method to estimate both the visible and occluded parts of the tracked objects using predictions is proposed. By segmenting good and bad predictions, we aim at dealing with the occlusions, the disappearances and the regularization of the occluded parts of the tracked objects. In the end, our model will allow tracking and segmenting several objects, while encouraging the conservation of their shapes and motions.

3.1 Using predictions to deal with occlusions

In order to define a new functional taking into account both the predictions and the occluded parts of the tracked objects, we need to clearly define the predicted sets $\mathcal{V}_i^{t+1|t}$ and $\mathcal{O}_i^{t+1|t}$.

3.1.1 Predicted sets

Definition 3.1: Assuming that the estimation of the visible and occluded parts of the object i at time t is known, and that this object is guided by a mean velocity vector \bar{v}_i^t between time t and $t + 1$, the predicted sets $\mathcal{V}_i^{t+1|t}$ and $\mathcal{O}_i^{t+1|t}$ are defined as:

$$\begin{aligned}\mathcal{V}_i^{t+1|t} &= \{y + \bar{v}_i^t \in \Omega^{t+1}, \text{ such that } y \in \mathcal{V}_i^t\}, \\ \mathcal{O}_i^{t+1|t} &= \{y + \bar{v}_i^t \in \Omega^{t+1}, \text{ such that } y \in \mathcal{O}_i^t\}.\end{aligned}\tag{4}$$

Note that the visible predicted sets of different objects can have a non null intersection. We will discuss in details in subsection 3.3 how the velocities \bar{v}_i^t are computed.

A simple but useful observation can be made from the construction of the predicted sets. At time t , the visible part \mathcal{V}_i^t of the object i is a subset of the whole object \mathcal{O}_i^t . As the predicted sets $\mathcal{V}_i^{t+1|t}$ and $\mathcal{O}_i^{t+1|t}$ have been built by translating \mathcal{V}_i^t and \mathcal{O}_i^t , the prediction of the visible part is necessarily a subset of the prediction of the whole object: $\mathcal{V}_i^{t+1|t} \subset \mathcal{O}_i^{t+1|t}$.

3.1.2 Occluded sets

There is an important point that must be clarified now: what assumption do we need to handle the partial and total occlusions of the objects? Indeed, $\mathcal{O}_i^{t+1} \setminus \mathcal{V}_i^{t+1}$, the occluded part of an object i at time $t+1$, can not be estimated only with the color data available at time $t+1$. Some additional information is needed to track correctly this occluded part. In this work we make the following assumption:

Assumption 3.2: *We assume that the occluded part of an object at time $t+1$ is a subset of the prediction of the whole object from time t : $\mathcal{O}_i^{t+1} \setminus \mathcal{V}_i^{t+1} \subset \mathcal{O}_i^{t+1|t}$.*

This necessary assumption is the only strong assumption used in this paper. We nevertheless believe that it makes sense as motion models are able to deal correctly with simple occlusions [15], [26], [27] for any kind of target. In this work, we will use simple motion estimation on the visible areas of the tracked object through Lucas-Kanade approach [21]. Other assumptions could have been made. For instance, one could have used prior information on the shape to deal with partial occlusions as in [9]. However, this kind of approach needs a pre-processing learning step depending on the target, that we would like to avoid. We then prefer not having any morphological assumption on the tracked object and let the dynamical model do its job.

3.1.3 Defining and labeling good and bad predictions

The estimation of the object i at time $t+1$, \mathcal{O}_i^{t+1} , permits to segment the prediction $\mathcal{O}_i^{t+1|t}$ into two disjoint subsets: the good predicted set $\mathcal{O}_i^{t+1|t} \cap \mathcal{O}_i^{t+1}$ and the bad predicted set $\mathcal{O}_i^{t+1|t} \setminus \mathcal{O}_i^{t+1}$.

Naturally, if the estimation of \mathcal{O}_i^t and its motion are good enough, the bad predicted set $\mathcal{O}_i^{t+1|t} \setminus \mathcal{O}_i^{t+1}$ should be empty. In practice, it is obviously not the case, as the objects can be deformable and the motion we consider is a simple translation.

We now introduce a second labeling function $\pi : \cup_i \mathcal{O}_i^t \mapsto [0; 1]$. This labeling function represents the good and bad predictions: a pixel $y^i \in \mathcal{O}_i^t$ will be a good (resp. bad) prediction if $\pi(y^i) = 1$ (resp. $\pi(y^i) = 0$).

Let us now explain how the good and bad predictions will allow representing disappearance and occlusion of the tracked pixels of the objects.

3.1.4 Interacting pixels

From definition 3.1, the pixel y^i of the object \mathcal{O}_i^t is associated with the pixel $x = y^i + \bar{v}_i^t$ of the image at time $t+1$. In order to enhance the model and consider disappearances and occlusions, we create a strong link between the labels of these interacting pixels y^i and x . Recalling that $\lambda(x) = i \Leftrightarrow x \in \mathcal{V}_i^t$, four cases can be described:

- y^i is a good prediction ($x \in \mathcal{O}_i^{t+1|t} \cap \mathcal{O}_i^{t+1}$) and x is labeled with the object i ($x \in \mathcal{V}_i^{t+1}$): x belongs to \mathcal{V}_i^{t+1} , the visible part of the object i (more precisely $x \in \mathcal{V}_i^{t+1} \cap \mathcal{O}_i^{t+1|t}$).
- y^i is a good prediction and x is not labeled with the object i ($x \notin \mathcal{V}_i^{t+1}$): x belongs to the occluded part of the object i ($x \in \mathcal{O}_i^{t+1|t} \setminus \mathcal{V}_i^{t+1} \subset \mathcal{O}_i^{t+1|t}$ from assumption 3.2).
- y^i is a bad prediction ($x \in \mathcal{O}_i^{t+1|t} \setminus \mathcal{O}_i^{t+1}$) and x is not labeled with the object i : x belongs to the bad predicted area associated to the object i ($x \in \mathcal{O}_i^{t+1|t} \setminus \mathcal{O}_i^{t+1}$).
- y^i is a bad prediction and x is labeled with the object i : the situation is impossible.

As shown in Table 1, a summary of these possibilities can then be made in term of labels.

	$\pi(y^i) = 1$	$\pi(y^i) = 0$
$\lambda(x) = i$	Good prediction and pixel visible ($x \in \mathcal{V}_i^{t+1}$)	Impossible
$\lambda(x) \neq i$	Good prediction and pixel occluded ($x \in \mathcal{O}_i^{t+1} \setminus \mathcal{V}_i^{t+1}$)	Bad prediction ($x \in \mathcal{O}_i^{t+1 t} \setminus \mathcal{O}_i^{t+1}$)

TABLE 1

Description of the different cases associated to the label value of interacting pixels $y^i \in \mathcal{O}_i^t$ and $x \in \Omega^{t+1}$, with $x = y^i + \bar{v}_i^t$.

3.2 Extending the energy

Let us now explain how the predicted sets $\mathcal{O}_i^{t+1|t}$ will be used and incorporated into the energy function in order to obtain a better temporal consistency and deal with occlusions.

3.2.1 Temporal consistency and bad segmentations rejections

The bad predictions come from the disappearance of some pixels of the object (from deformation) and/or from error of construction of the predicted sets (due to the motion model).

A new energy term is needed to monitor the bad predicted area $\mathcal{O}_i^{t+1|t} \setminus \mathcal{O}_i^{t+1}$. From definition 3.1, a pixel $y^i \in \mathcal{O}_i^t$ has as corresponding predicted pixel: $x = y^i + \bar{v}_i^t \in \Omega^{t+1}$. As the bad predicted area corresponds to the case $\pi(y^i) = 0$ and $\lambda(x) \neq i$, the new term is defined as

$$\mathcal{E}_\beta(\pi, \lambda) = \sum_{i=1}^N \sum_{y^i \in \mathcal{O}_i^t} \beta_i(y^i) \delta(\pi(y^i)) [1 - \delta(\lambda(x) - i)].$$

Here the penalization is made through the function $\beta_i \in \mathbb{R}$, described in the next section, that will measure the difference between the local measured velocity at point y^i and the mean motion of the object. Using the motion information will allow rejecting some bad segmentations that may occur at one time. Namely, if the velocity of a pixel y^i is far from the mean motion while x does not belong to the object, minimizing \mathcal{E}_β is equivalent to setting y^i as a bad prediction. This new term will also add temporal consistency to successive segmentations by keeping as object the predicted pixels corresponding to good predictions.

3.2.2 Tracking occluded parts of the objects

The good predicted set $\mathcal{O}_i^{t+1|t} \cap \mathcal{O}_i^{t+1}$ can be separated into two parts: its visible part ($\mathcal{O}_i^{t+1|t} \cap \mathcal{V}_i^{t+1}$) and its occluded part ($\mathcal{O}_i^{t+1|t} \cap (\mathcal{O}_i^{t+1} \setminus \mathcal{V}_i^{t+1})$).

As the original energy (1) already measures the likelihood of the whole visible set \mathcal{V}_i^{t+1} through the data term, we only need to handle the occluded part of the prediction. This will be done by penalizing the area of the occluded part ($\mathcal{O}_i^{t+1} \setminus \mathcal{V}_i^{t+1} = \mathcal{O}_i^{t+1|t} \cap (\mathcal{O}_i^{t+1} \setminus \mathcal{V}_i^{t+1})$, from assumption 3.2). This region corresponds to the pixels $y^i \in \mathcal{O}_i^t$ such that $\pi(y^i) = 1$, whose associated predicted pixel $x = y^i + \bar{v}_i^t$ is occluded by another object ($\lambda(x) \neq i$). A new energy term can then be defined:

$$\mathcal{E}_\mu(\pi, \lambda) = \mu \sum_{i=1}^N \sum_{y^i \in \mathcal{O}_i^t} \tilde{d}_i(x) \delta(\pi(y^i) - 1) [1 - \delta(\lambda(x) - i)].$$

This new term includes a weight parameter $\mu \in \mathbb{R}$. Moreover, as we want the occluded and visible parts of an object to be spatially close, we better use the distance $d_i(x)$ of a pixel $x \in \Omega^{t+1}$ to $\mathcal{V}_i^{t+1|t}$ to encourage the final occluded set to be close to the prediction of the visible part of the object. As this distance is null $\forall x \in \mathcal{V}_i^{t+1|t}$, we here consider the distance $\tilde{d}_i(x) = d_i(x) + 1$. Indeed, we also want to penalize the occluded parts contained inside the prediction: $\mathcal{O}_i^{t+1|t} \cap \mathcal{V}_i^{t+1}$. Let us note that, if the visible part of the object i is empty (complete occlusion of the object i), we compute $d_i(x)$ as the distance to the predicted set $\mathcal{O}_i^{t+1|t}$, $\forall x \in \Omega^{t+1}$.

3.2.3 Coherence constraint

Recalling that the good predicted visible area $(\mathcal{V}_i^{t+1} \cap \mathcal{V}_i^{t+1|t})$ is implicitly considered by the data and appearance terms of energy (3), the case $\pi(y^i) = 1$ and $\lambda(x = y^i + \bar{v}_i^t) = i$ is already treated. From this observation and the two last energy terms, it appears that only one case is still not addressed. In fact, when the pixel $y^i \in \mathcal{O}_i^t$ is a bad prediction of the object i , its corresponding predicted pixel $x = y^i + \bar{v}_i^t$ can not be associated with the object i . This last case can be explained mathematically as:

$$\pi(y^i) = 0 \Rightarrow \lambda(x = y^i + \bar{v}_i^t) \neq i. \quad (5)$$

We impose this constraint by adding an energy term:

$$\mathcal{E}_C(\pi, \lambda) = U \sum_{i=1}^N \sum_{y^i \in \mathcal{O}_i^t} \delta(\pi(y^i)) \delta(\lambda(x) - i),$$

with $U \rightarrow +\infty$ a huge value that prevents from impossible associations.

3.2.4 Spatial regularization of the whole objects

Finally, an additional term will allow regularizing spatially the occluded parts of the objects by penalizing the length of the boundary between good and bad predicted regions. It will therefore involves the neighborhoods $\mathcal{N}_i^{l,t+1}(y^i) = \{z^i \in \mathcal{O}_i^{t+1|t}, \text{ such that } z^i \in \mathcal{N}^l(y^i)\}$ and $R_p \geq 0$ a constant of regularization:

$$\mathcal{E}_{Rp}(\pi) = R_p \sum_{i=1}^N \sum_{y^i \in \mathcal{O}_i^t} \sum_{z^i \in \mathcal{N}_i^{l,t+1}(y^i)} [1 - \delta(\pi(y^i) - \pi(z^i))].$$

3.2.5 Final Model

Merging all the terms introduced from the beginning, our tracking problem consists in minimizing the following energy:

$$\min_{\pi, \lambda} \mathcal{E}(\pi, \lambda) = \underbrace{\mathcal{E}_D(\lambda) + \mathcal{E}_\gamma(\lambda)}_{(A)} + \underbrace{\mathcal{E}_\beta(\pi, \lambda) + \mathcal{E}_\mu(\pi, \lambda) + \mathcal{E}_C(\pi, \lambda)}_{(B)} + \underbrace{\mathcal{E}_R(\lambda)}_{(C)} + \underbrace{\mathcal{E}_{Rp}(\pi)}_{(D)} \quad (6)$$

The terms of the energy have been merged in the following sense. The term A consists of energies involving single pixels of the image domain Ω^{t+1} . The term B is dedicated to energies depending on pairs of interacting pixels: $y^i \in \mathcal{O}_i^t$ and $x = y^i + \bar{v}_i^t \in \Omega^{t+1}$. The term C (resp. D) involves pairs of neighboring pixels in the image area Ω^{t+1} (resp. the predicted areas \mathcal{O}_i^t).

Minimizing this energy gives the optimal labeling λ and enables obtaining the sets $\mathcal{V}_i^{t+1} \subset \mathcal{O}_i^{t+1}$ in the following way:

- for all $x \in \Omega^{t+1}$, if $\lambda(x) = i$, then $x \in \mathcal{V}_i^{t+1} \subset \mathcal{O}_i^{t+1}$,
- for all $y^i \in \mathcal{O}_i^t$, if $\pi(y^i) = 1$, then $x = y^i + \bar{v}_i^t \in \mathcal{O}_i^{t+1}$.

With respect to the energy of Malcom et al. (3), our model now fully describes what is happening in real tracking applications: appearance, disappearance and occlusions. Let us now explain how the motion information is used to define the functions β_i .

3.3 Use of motion information

To build the predicted sets and to compute the function $\beta_i(y^i)$, we assume that the objects move, up to an uncertainty, with a mean velocity \bar{v}_i^t . We therefore use Gaussian velocity models and characterize the motion of each object i at time t with the law $\mathcal{N}(\bar{v}_i^t, \sigma_i^t)$ defined by the mean motion \bar{v}_i^t and the variance σ_i^t .

To compute the unknowns \bar{v}_i^t and σ_i^t , a set of points of interest is considered: $\{\mathbf{p}_{ij}^t\}_{j=1 \dots N_i^t} \in \mathcal{V}_i^t$, where N_i^t is the number of points of interest detected (by the Harris corner detector [16], for example) in the visible part of object i at time t . The optical flow vectors v_{ij}^t are computed at these points with a simple Lucas-Kanade multi-resolution scheme [21] (using the values $I(\mathbf{p}_{ij}^t, t)$, $\nabla I(\mathbf{p}_{ij}^t, t)$ and $I(\mathbf{p}_{ij}^t, t+1)$ at the finer scale). More complex motion estimators could have been used. Nevertheless, as our prediction is finally obtained with a simple mean motion, we prefer to rely on a fast and simple motion estimator.

To add a temporal consistency in the successive estimations, we also rely on a dynamical model on the velocities of each object.

3.3.1 Dynamical model

Assuming that we already have a previous estimation of the mean \bar{v}_i^{t-1} and the variance σ_i^{t-1} of the velocity of the object, we use these values to filter the new vector estimation with:

$$\tilde{v}_{ij}^t = K_{ij} v_{ij}^t + (1 - K_{ij}) \bar{v}_i^{t-1} \quad (7)$$

$$\text{where } K_{ij} = \max \left(k, \underbrace{\exp\left(-\frac{|v_{ij}^t - \bar{v}_i^{t-1}|}{\sigma_i^{t-1}}\right)}_{\text{Filter the velocity values}} \underbrace{P_i(\mathbf{p}_{ij}^t)}_{\text{Probability of belonging to the object } i} \right),$$

and k is the minimum wanted value of mean velocity update rate. This parameter determines *a priori* the quality of the chosen constant motion model. If an object really follows this velocity model, one can set $k = 0$. On the other hand, if the velocity of the tracked object is quite unpredictable, k should be chosen closer to 1. From our experimentations, we fixed this parameter to 0.25, in order to ensure a minimum evolution of the motion value.

In [22], [23], the involved velocity model is a simple filter that projects the centroid of an object forward in time with respect to the moving average of the past $T > 0$ displacements. The authors also take into account the possible error of prediction, by computing a scalar coefficient for each object. We believe that such an error model is too coarse and can not be adapted in case of objects presenting strong changes of motion direction. Thus, using a parameter representing the velocity uncertainty (k instead of T) and applying a local process to the different points of interest enables us to capture more information. As will be demonstrated in the experimental section, in contrary to our method, the criteria used by Malcolm et al. is unable to recover bad prediction, and sometimes leads to unwanted over-segmentations.

Remark also that more sophisticated models involving dense motions [7] and/or the analysis of occluded motion could have been studied [24], [34]. Nevertheless, as a pixel to pixel correspondence is needed between the prediction \mathcal{O}_i^t and predicted $\mathcal{O}_i^{t+1|t}$ sets, we chose the simplest form of motion (translation) and a basic dynamical model, in order to reduce the computational cost. From equation (7), the new mean and variance values of the object velocity can be computed:

$$\bar{v}_i^t = \frac{1}{N_i^t} \sum_{j=1}^{N_i^t} \tilde{v}_{ij}^t, \text{ and } \sigma_i^t = \frac{1}{N_i^t} \sum_{j=1}^{N_i^t} (\tilde{v}_{ij}^t - \bar{v}_i^t)^2. \quad (8)$$

If no point of interest has been detected for an object i , we simply choose $\bar{v}_i^t = \bar{v}_i^{t-1}$ and $\sigma_i^t = \sigma_i^{t-1}$.

3.3.2 Predicted sets construction

Thanks to definition 3.1, for each object i , the mean motion vector \bar{v}_i^t enables obtaining, by translation of \mathcal{V}_i^t and \mathcal{O}_i^t , the predicted sets $\mathcal{V}_i^{t+1|t}$ and $\mathcal{O}_i^{t+1|t}$. It may happen that, for some pixels $y^i \in \mathcal{O}_i^t$, their correspondent $y^i + \bar{v}_i^t$ does not belong to the image domain Ω^{t+1} . Once the predictions have been realized, we redefine the sets \mathcal{O}_i^t *a posteriori*, by removing these pixels y^i :

$$\mathcal{O}_i^t = \mathcal{O}_i^{t|t+1} = \{y^i \in \mathcal{O}_i^t, \text{ such that } y^i + \bar{v}_i^t \in \Omega^{t+1}\}. \quad (9)$$

This step only affects the parts of the objects that leave the image domain, so that it has no consequence on the predictions. It nevertheless enables having a useful bijective correspondence between the sets \mathcal{O}_i^t and $\mathcal{O}_i^{t+1|t}$. In practice, we simply use the integer parts of \bar{v}_i^t and σ_i^t in order to have a pixel to pixel correspondence.

3.3.3 Error of prediction

The computed motion vectors also have an important role in the functions $\beta_i(y^i)$. More precisely, if an interest point \mathbf{p}_{ij}^t has a velocity v_{ij}^t very different from the mean velocity of the object i , its associated predicted pixel $x = \mathbf{p}_{ij}^t + \bar{v}_i^t$ is more unlikely to belong to object i . In other words, we can assume that the motion vector computed at this point is erroneous. Introducing the variance coefficient: $e_{ij}^t = \frac{|v_{ij}^t - \bar{v}_i^t|}{\sigma_i^{t-1}}$, we finally define the function of prediction errors $\beta_i(\mathbf{p}_{ij}^t)$ as:

$$\beta_i(\mathbf{p}_{ij}^t) = \beta + \underbrace{1/(e_{ij}^t + \epsilon)}_{\text{Encourages the vectors close to the mean value}} - \underbrace{(2^{e_{ij}^t} - 1)}_{\text{Penalizes the motion vectors far from the mean value}}, \quad (10)$$

where $\beta \in \mathbb{R}$ and $\epsilon > 0$ are some real parameters. If the measured motion v_{ij}^t at point \mathbf{p}_{ij}^t is close to the mean motion value \bar{v}_i^t of the object i , then the predicted pixel $x = \mathbf{p}_{ij}^t + \bar{v}_i^t$ should belong to the object i . In such a case, e_{ij}^t will be small and $\beta_i(x)$ high. Thanks to the second term of relation (10), the cost of assigning \mathbf{p}_{ij}^t as a bad prediction will be high.

The third term has almost the opposite role: if the motion vector measured at point \mathbf{p}_{ij}^t is far enough from the mean velocity of the object i , its corresponding pixel may not belong to the object for the next time step. The exponential operator will then decrease the value of $\beta_i(y^i)$ and the pixel $x = \mathbf{p}_{ij}^t + \bar{v}_i^t$ will be encouraged to disappear (i.e., belong to the set $\mathcal{O}_i^{t+1|t} \setminus \mathcal{O}_i^{t+1}$).

Finally, for the points y^i belonging to \mathcal{O}_i^t which are not interest points, we set $\beta_i(y^i) = \beta$ (so, *a fortiori*, in the current occluded sets, we have $\beta_i(y^i) = \beta, \forall y^i \in \mathcal{O}_i^t \setminus \mathcal{V}_i^t$). As a specific regularization is involved on the predicted sets $\mathcal{O}_i^{t+1|t}$ (with the term E of the cost function), the local motion information, that is only available at the interest points \mathbf{p}_{ij}^t , will be diffused. Note also that if the value of the motion is not informative, with σ_i^{t-1} high or $e_{ij}^t = 1$, then $\beta_i(y^i) = \beta$.

This is a simple but important improvement of [22], [23], as our use of the motion measures allows correcting the bad estimations and predictions of the tracked objects.

4 GRAPH CONSTRUCTION

To minimize the energy (6) we need to create a graph adapted to the energy such that the minimum cut corresponds to the minimum of the energy.

Let $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ be a directed graph built for our graph cuts minimization problem. The set of vertices $\mathbf{V} = \{n_x\}$ classically corresponds to the set of pixels $x \in \Omega^{t+1}$ with two additional distinguished terminal vertices $\{S, T\}$ called the source and the sink.

A novelty of our graph, inspired from [5], [6], is that additional vertices n_{y^i} are considered, for all previously segmented pixels $y^i \in \mathcal{O}_i^t$. They are necessary to handle properly the occluded parts of the objects, since such information can not be represented considering only one vertex for one pixel of the current image. The set of edges representing the energy value and linking the vertices is denoted by \mathbf{E} . As illustrated in Figure 4, the vertices n_{y^i} and n_x , corresponding to a previously estimated pixel y^i of object i and its associated prediction $x = y^i + \bar{v}_i^t$ naturally communicate in the graph. These links are the key point of the tracking of occluded parts.

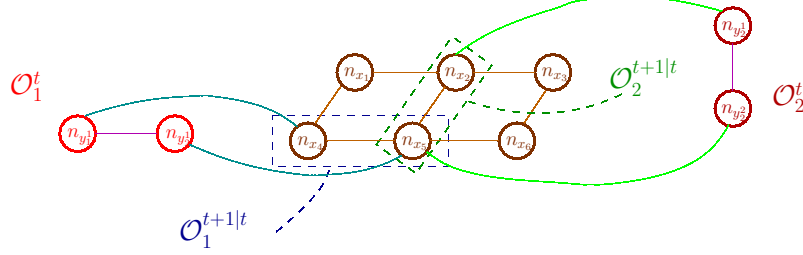


Fig. 4. Nodes of the graph. For each pixel $x_i \in \Omega^{t+1}$ of the current image, a vertex n_{x_i} is created. For each pixel y_j^i , $j = 1, 2$, of the object \mathcal{O}_i^t , $i = 1, 2$, some additional vertices $n_{y_j^i}$ are added. The predicted sets $\mathcal{O}_i^{t+1|t} \in \Omega^{t+1}$ are also presented and the bijective links between the vertices of \mathcal{O}_i^t and $\mathcal{O}_i^{t+1|t}$ are drawn.

In order to minimize the energy (6) within the graph, an α -expansion algorithm [3] is applied. In one so-called cycle, the α -expansion successively tests each label $\alpha \in \{0 \cdots N\}$. The algorithm then realizes cycles until convergence.

Assuming that we have current labeling functions λ and π , during an expansion corresponding to the label $\alpha \in \{0 \cdots N\}$, a node n_x associated to a pixel $x \in \Omega^{t+1}$, can shift to the label α or keep its current label $\lambda(x)$. The situation is different for the vertices n_{y^i} corresponding to the pixels $y^i \in \mathcal{O}_i^t$ of an object i . We recall that for these pixels, the label $\pi(y^i) = 0$ corresponds to a bad prediction, whereas $\pi(y^i) = 1$ denotes a good prediction. Remember also that the label of a pixel of the predicted set $y^i \in \mathcal{O}_i^t$ is linked to the label of its corresponding prediction $x = y^i + \bar{v}_i^t \in \Omega^{t+1}$. This leads to considering two cases for the possible moves of the label $\pi(y^i)$: $\alpha = i$ and $\alpha \neq i$.

For an expansion α , the pixels y^i of the set \mathcal{O}_i^t associated to the particular object $i = \alpha$ can keep their current label (0 or 1) or move it to 1 (i.e., good prediction). The interacting pixel $x = y^i + \bar{v}_i^t$ can take the label α , only if y^i is a good prediction.

On the other hand, the pixels of the sets \mathcal{O}_i^t relatives to the objects $i \neq \alpha$ can keep their current label (0 or 1) or move it to 0 (i.e., bad prediction). Namely, if y^i is not a good prediction, its interacting pixel $x = y^i + \bar{v}_i^t$ can not take the label α (from relation (5)). This last case is valid for the sets \mathcal{O}_i^t of all the objects ($i = 1 \cdots N$) when the background is treated ($\alpha = 0$).

In the Appendix, we give more details on the α -expansion algorithm that minimizes the cost function (6) and finds the labeling functions λ and π . One can check that all the involved energies are submodular or regular in the sense of [20]. In this work, we use the algorithm presented in [2] to find the maximal flow of the graph.

In practice, at each time t , the labeling function is initialized with $\lambda(x) = 0$ (all the pixels of the current image are associated to the background) and $\pi(y^i) = 0$ (all the predictions are bad).

Note that if we find $\mathcal{O}_i^{t+1} = \emptyset$, for an object i at time $t + 1$, we then remove this object from the tracking process. The overall process is summed up in algorithm 4.1.

Algorithm 4.1:

- 1 Initialization:
 - Assuming that all the objects are entirely visible at the initial time $t = 0$, each object $i = 1 \dots N$ is initialized with $\mathcal{V}_i^0 = \mathcal{O}_i^0$.
 - The set $\mathcal{V}_0^0 = \mathcal{O}_0^0$ is built.
 - The probability functions P_i , $i = 0 \dots N$, are built.
- 2 Process at time $t + 1$:
 - Find the points of interest in \mathcal{V}_i^t , $i = 1 \dots N$, and compute their optical flow.
 - Filter the motion vectors in order to obtain \bar{v}_i^t and the functions β_i , $i = 1 \dots N$.
 - Predict the sets $\mathcal{V}_i^{t+1|t} \subset \mathcal{O}_i^{t+1|t}$, $i = 1 \dots N$.
 - Initialize the sets at time $t + 1$ with $\mathcal{V}_i^{t+1} = \mathcal{O}_i^{t+1} = \emptyset$, $i = 1 \dots N$ and $\mathcal{V}_0^{t+1} = \mathcal{O}_0^{t+1} = \Omega^{t+1}$.
 - Construct the graph and apply the α -expansion algorithm given in Appendix to minimize the energy (6) and obtain \mathcal{V}_i^{t+1} and \mathcal{O}_i^{t+1} , $i = 0 \dots N$.
 - Set $t = t + 1$ and return to 2.

5 EXPERIMENTS

Before presenting experiments, the parameters related to the modeling choices are discussed.

5.1 Short discussion

As the energy we minimize is composed of 6 different terms, the minimum parameters that have to be tuned is 5. However, from our experiments, we have observed that most of them can be fixed.

5.1.1 Object distributions

As the results were obtained from color sequences, we chose to represent the probability of a pixel to belong to an object with a normalized 3D histogram. We decided to use 16 bins for each channel of color, so that each object is represented with 16^3 bins. To handle the changes of illumination and be able to deal with occlusions, the histograms must be updated carefully [25]. In this work, we update them continuously, taking into account both the past histogram and the current visible part of each tracked object.

5.1.2 Regularity function

To regularize the segmentation on the image, we use a classical function (as in [1]) that encourages the discontinuities of segmentation to coincide with the image discontinuities. For all $x \in \Omega^{t+1}$, $z \in \mathcal{N}^l(x)$, we then chose the function F as:

$$F(I(x, t), I(z, t)) = \frac{1}{|x - z|} \exp\left(-\frac{|I(x, t) - I(z, t)|^2}{\sigma_I}\right)$$

where σ_I is the allowed standard variation. Let us underline that in all our applications, we fixed $\sigma_I = 80$ and the regularization parameter weighting this energy to $R_\Omega = 10$. We use a 8-neighborhood system, that corresponds to $n = \sqrt{2}$ in definition (2). The regularization of the occluded parts of the objects are handled with the parameter $R_p = 1$.

5.1.3 Occlusion vs disappearance

As we would like to keep tracking the occluded part of the object, we should impose $\mu < \beta$. However, if the velocity computed at a point of interest y^i is far from the mean velocity of its corresponding object i , thanks to definition (10), we will have $\beta_i(y^i) < \mu$. This will encourage disappearance instead of occlusion. The parameter ϵ of definition (10) has been set to 0.01.

5.2 Results

The algorithm has been tested on 5 image sequences. The hand-made initializations, the tracking results, the different parameters as well as the mean computational costs are given. Both visible and occluded parts of the tracked objects are drawn (dark color for the visible part and light color for the occluded part²). The occluded part acts like an uncertainty guided by the prediction around the visible estimation and helps the process to deal with occlusions.

To compare our results with the method of [22], [23], we realized the same experiments fixing R_p , β_i and μ to 0, thus recovering the original energy (3). We did not compare with the method of [5], [6] as it requires some external observations.

As illustrated by the 5 image sequences treated, when tuning appropriately the three main parameters μ , β and γ , our method is more able to deal with partial and total occlusions and gives more stable segmentations with respect to [22], [23]. Indeed, the tracking of all the objects is realized conjointly by the global graph cuts minimization, and not independently as in [5], [6], [23]. Unlike [22], the representation of the occluded and visible parts of the objects allows dealing well with interacting objects. To show the limitations of our model, namely the tuning of the three parameters μ , β and γ , we will also present the results obtained by our method with standard values: $\mu = 1.5$, $\beta = 2.5$ and $\gamma = 0.5$ and explain the reason of the failures. Moreover, here we consider errors of prediction that enable to correct bad estimations. Let us also note that the kind of sequences we treat here is much more challenging than the one presented in [22], [23], where objects with slow motions in quite uniform background (as the football example here presented) were tracked.

5.2.1 Wakeboarder

(240 images, 340x240 pixels, 1 object, 5.3 frames/second).

Parameters: $\mu = 1.5$, $\beta = 2.5$ and $\gamma = 0.5$ (little enough to authorize the appearance of pixels quite far from the prediction, as the motion of the wakeboarder is large).

This first sequence presents a wakeboarder who has a lot of change of motion direction. In this simple sequence without occlusions, as illustrated in Figure 5, the wakeboarder is well tracked along time.

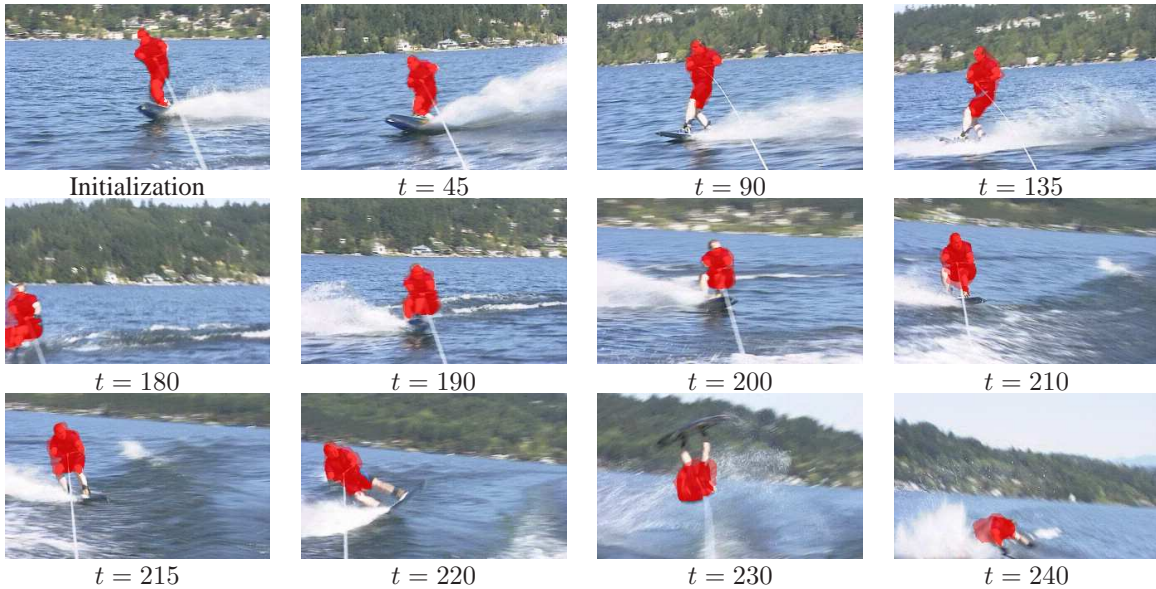


Fig. 5. Wakeboarder sequence: $\mu = 1.5$, $\beta = 2.5$, $\gamma = 0.5$. The dark red indicates the visible part of the tracked object whereas the light red denotes the occluded part. The large motion of the boarder at the end of the sequence is well handled by the algorithm.

2. The different colors are more visible in the electronic version or on the videos available at http://sites.google.com/site/nicolaspapadakis/video_tracking

To illustrate the influence of the parameters and the interest of our motion error model, we applied the method of [23] to this sequence by fixing R_p , β_i and μ to 0. In Figure 6, we see that both our method and the one of [23] fail if the parameter γ is too high, as the prediction is bad and the dynamical model is not adapted to the large change of motions.

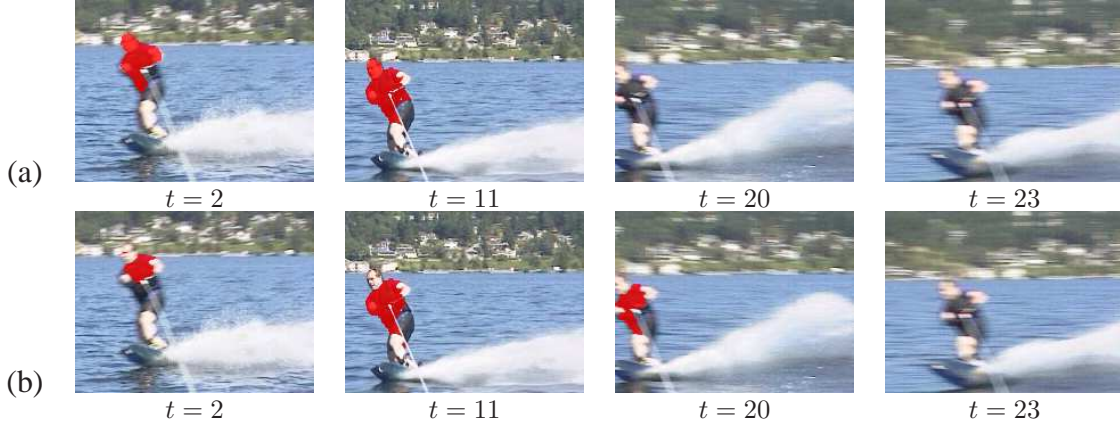


Fig. 6. Wakeboarder sequence results: (a) Application of [23] with $R_p = 0$, $\mu = 0$, $\beta_i = 0$ and $\gamma = 2$. (b) Our method with $R_p = 1$, $\mu = 2$, $\beta = 2.5$ and $\gamma = 2$. The motion of the wakeboarder presents some large changes of amplitude in time that are not well handled by the dynamical model. The tracking is thus lost for both methods with γ too high, as this parameter weights the distance of the estimated segmentation with respect to the prediction.

From Figure 7, it is clear that a smaller value of γ allows [23] to search farther from the prediction and track well the object. However, as a global error of prediction on the object is taken into account in [23], through a scalar number to weight the influence of γ , this modification leads to over-segmentations. The segmentation boils that has boiled over the background at one time (frame 70) is never corrected (frames 80 and 90). On the contrary, our scheme allows enforcing the points of interest to belong or not to the segmentation, thanks to the local decision taken by relation (10). We are then more robust to over-segmentation.

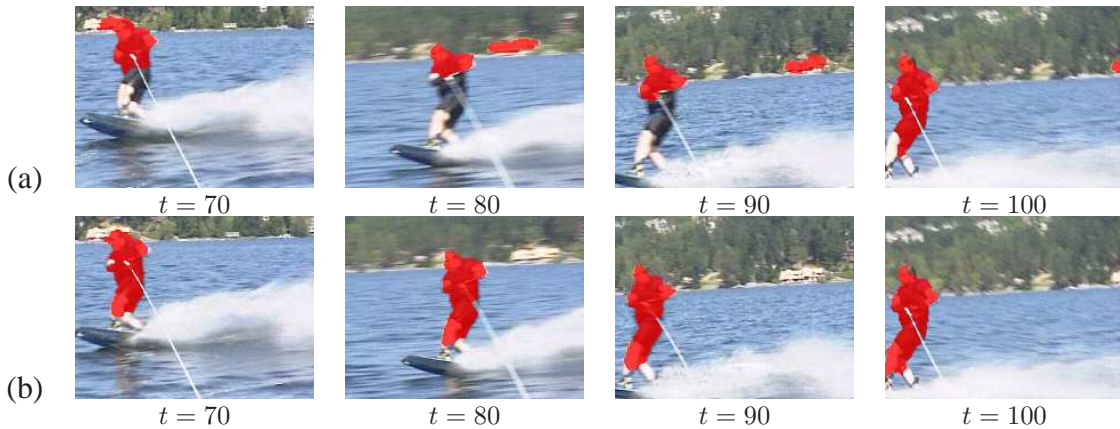


Fig. 7. Wakeboarder sequence results: (a) Application of [23] with $R_p = 0$, $\mu = 0$, $\beta_i = 0$ and $\gamma = 0.5$. (b) Our method with $R_p = 1$, $\mu = 2$, $\beta = 2.5$ and $\gamma = 0.5$. The tracking is quite good in this example, the large motion between consecutive frames is handled by reducing the value of the parameter γ to 0.5. This lower parameter leads to segmenting parts of the background as object with [23], whereas our error of prediction model (equation (10)) is able to reject some bad detections.

5.2.2 Truck and pedestrians

(100 images, 340x240 pixels, 2 objects, 4.2 frames/second).

Parameters: $\mu = 1$, $\beta = 1.5$ and $\gamma = 2.0$ (the motion of the objects are small).

In this sequence from PETS 2001, a truck and a pedestrian with linear motions are tracked. We can see, from the occluded part representation and the dynamic constraint, that the pedestrian is well recovered after its occlusion. Moreover, as illustrated on Figure 8, even if some bad segmentations of the truck are sometime present, the segmentation is always well recovered after some frames, as the motion of the bad part of the segmentation is rejected by the process. When comparing with results obtained by [22] on Figures 2 and 3, we see that our method is able to solve the two principal problems: dealing with occlusions and reject false segmentations.

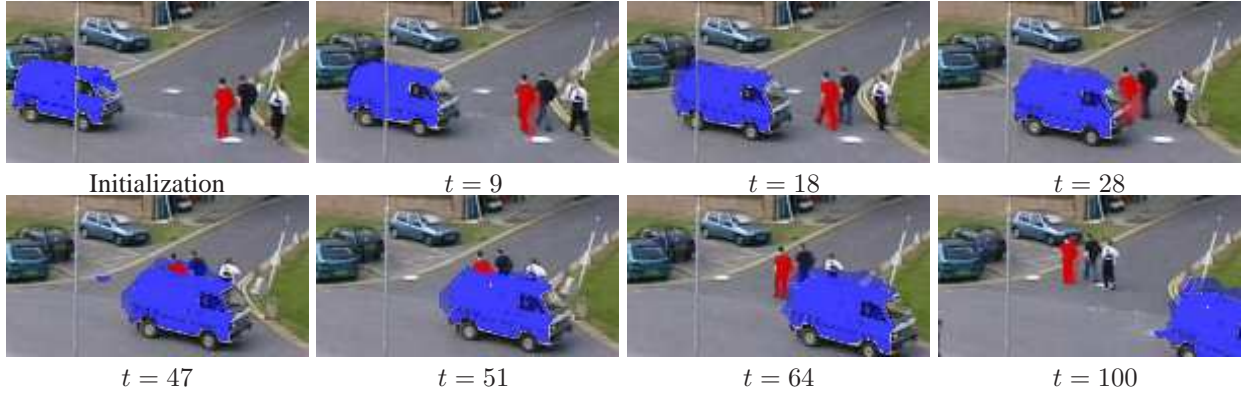


Fig. 8. **Truck and pedestrian sequence** $\mu = 1$, $\beta = 1.5$ and $\gamma = 2.0$: . The dark blue (resp. red) indicates the visible part of the truck (resp. the pedestrian) whereas the light blue (resp. red) denotes the occluded part. Thanks to the tracking of the occluded parts, the pedestrian is well recovered after its occlusion ($t = 64$). When the segmentation of the truck boils over the background (see $t = 47$), this erroneous area is rejected after few frames (see $t = 51$) thanks to the error model. It is also interesting to note that the electric pole that occludes the truck on frames $t = 9$ and $t = 18$ is always segmented as the background (it is colored in light blue which mean that the truck is occluded).

To illustrate the influence of the terms B and C of our energy we now show the results obtained by setting particular value of μ and β_i . First of all, let us consider that $\beta_i(x) = \beta$. As shown in Figure 9, the motion information is not used to reject bad segmentation that may occur at one time. The segmentation of the truck that boils over the background is never correctly recovered.



Fig. 9. **Truck and pedestrian sequence** $\mu = 1$, $\beta_i = \beta = 1.5$ and $\gamma = 2.0$: The pedestrian is well tracked, but the bad segmentations are not rejected, as no error is taken into account on the local motion of the interest points.

To moderate the quality of our result in this sequence, we show in Figure 10 that with the standard parameters, the tracking fails and oversegments the objects. Namely, as the appearance parameter is too small with respect to the size of the image, the process allows searching for the target too far away from the prediction. A solution would be to study the value of this parameter with respect to the velocity of the object. It will be the subject of future works.



Fig. 10. **Truck and pedestrian sequence with standard parameters $\mu = 1.5$, $\beta = 2.5$ and $\gamma = 0.5$:** The small value of the parameter γ leads the process to search for the target too far away from the predictions.

We now describe the extreme possible values of $\mu \in \mathbb{R}$ and $\beta \in \mathbb{R}$. If we set $\mu = -100$, for example, the model will always prefer to consider that the prediction is occluded. This case will then just put the objects as occluded and will not realize tracking. Similarly, fixing $\beta_i = -100$ will encourage the model to consider that the predictions are always bad, this will end the tracking after one frame. The cases $\mu = 100$ and/or $\beta_i = 100$ are more interesting. By highly increasing the value of the parameter μ only, we prohibit occlusions and the tracking of the pedestrian is lost after its partial occlusion (Figure 11). Respectively, we obtain increasing occluded areas by setting $\beta = 100$, as this value prohibits the disappearance of the tracked area of the objects (Figure 12). Finally, if both μ and β_i have high values, it leads to increasing the area of the visible parts of the objects with time (Figure 13).



Fig. 11. **Truck and pedestrian sequence $\mu = 100$, $\beta = 1.5$ and $\gamma = 2.0$:** The pedestrian is lost when partially occluded by the truck. Indeed, the model does not allow occlusion as $\mu = 100$. The oversegmentations in the background are still present.



Fig. 12. **Truck and pedestrian sequence $\mu = 1$, $\beta_i = 100$, and $\gamma = 2.0$:** The dark blue (resp. red) indicates the visible part of the truck (resp. the pedestrian) whereas the light blue (resp. red) denotes the occluded part. As the disappearance of the prediction is discouraged by the model (with $\beta_i = 100$), the areas of the occluded parts of the objects are increasing with time.



Fig. 13. **Truck and pedestrian sequence $\mu = 100$, $\beta_i = 100$, and $\gamma = 2.0$:** As the presence of occluded parts (with $\mu = 100$) and the disappearance of the prediction (with $\beta_i = 100$) are discouraged by the model, the areas of the visible parts of the objects are increasing with time.

5.2.3 People in the station

(300 images, 240x220 pixels, 4 objects, 2.5 frames/second).

Parameters: $\mu = 0.5$, $\beta = 1$ and $\gamma = 1$.

This sequence from PETS 2006 presents four pedestrians that are all dressed up with black clothes. Three men have a very similar motion and walk in group, whereas the woman has an opposite motion. As illustrated by Figure 14, the segmentation of the group of three people is quite well estimated all along the sequence, thanks to the temporal constraint. Moreover, even if the segmentation of one of the man boils over the woman when they cross, the motion constraint rejects this bad segmentation after few frames. However, the process sometimes rejects the feet of the pedestrian, as they have a motion different from the mean motion of the person. Considering additional information such as the direction of the motion could enable recovering these missing parts of the objects.

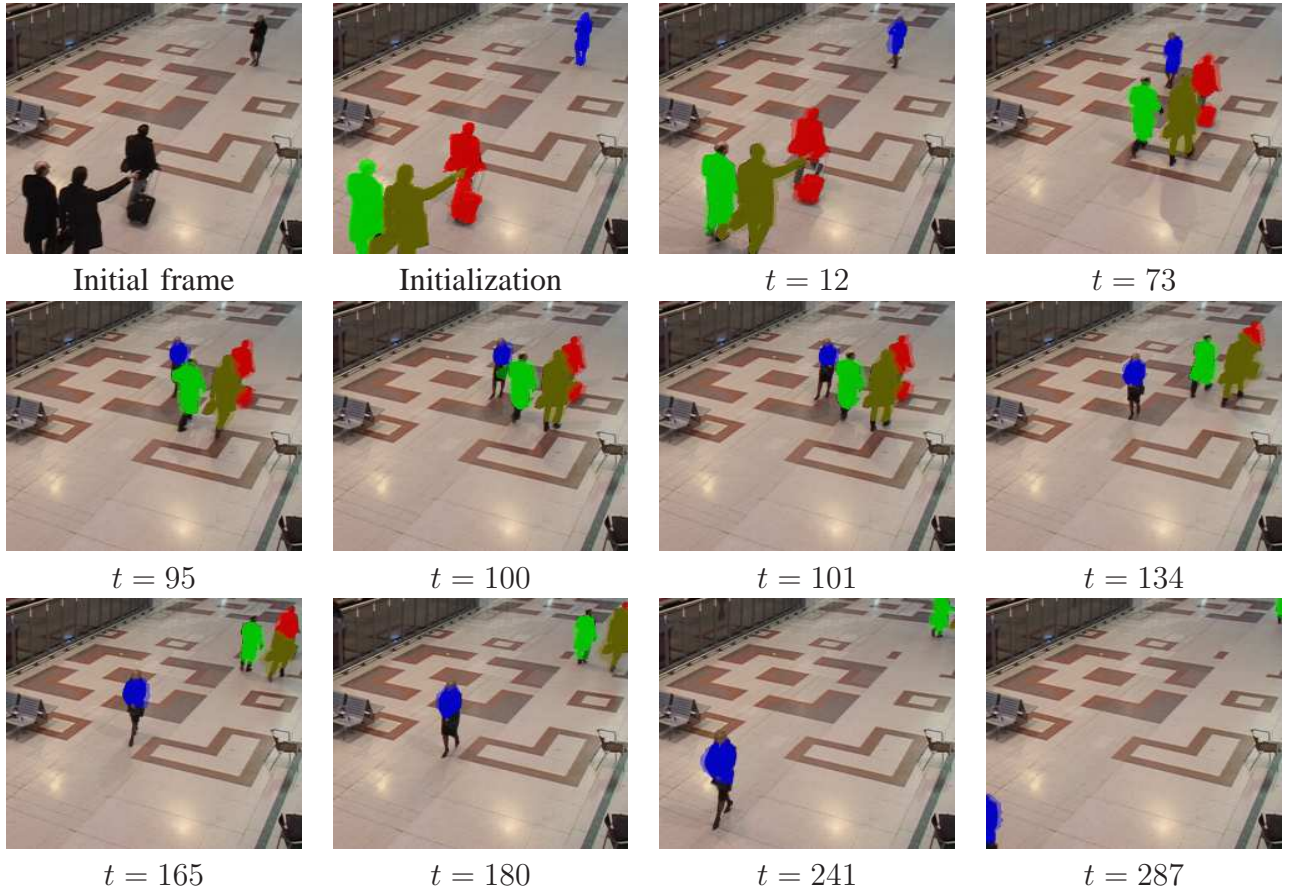


Fig. 14. **People in the station sequence:** $\mu = 0.5$, $\beta = 1$ and $\gamma = 1$. The dark colors indicate the visible parts of the tracked objects whereas the light ones denote the occluded parts. As illustrated by the first frame, all 4 pedestrians are dressed up with black clothes. The algorithm is nevertheless able to track correctly each person. Even if the segmentation of one of the man boils over the woman when they cross ($t = 95$), the motion constraint occludes this bad segmentation after a few frames ($t = 100$), before rejecting it ($t = 101$).

When using the standard parameters, the process gives globally good results but it merges two pedestrians, as illustrated in Figure 15. Note that the partial occlusion with the woman is nevertheless well recovered.



Fig. 15. **People in the station sequence - results with standard parameters:** $\mu = 1.5$, $\beta = 2.5$ and $\gamma = 0.5$. The algorithm fails by merging two pedestrians.

The method of [22] also performs quite well on this example, as can be seen on Figure 16. However, as the occluded parts are not tracked, the results are less accurate when partial occlusion occurs and the segmentation can boil over other pedestrian and the suitcase.

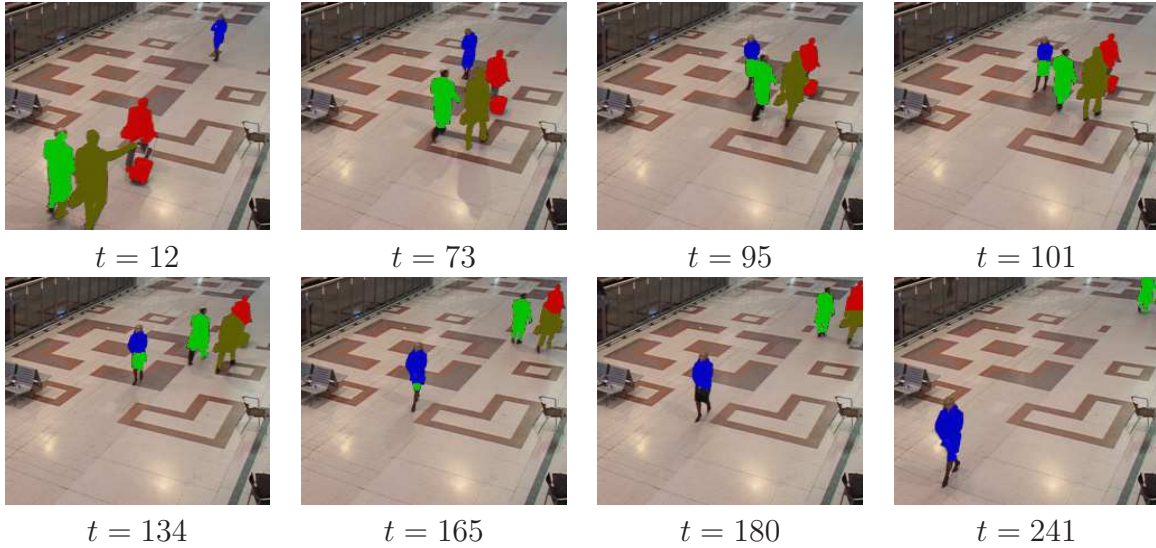


Fig. 16. **People in the station sequence - results with [22]:** $R_p = 0$, $\mu = 0$, $\beta_i = 0$ and $\gamma = 1$. The algorithm fails with the partial occlusion of persons in green and blue as illustrated on frames 95 - 134 and the false green part is only rejected after frame 165 thanks to the regularization. Moreover, the upper person in red is finally confounded with the pedestrian in brown (frame 180), from the absence of occluded part tracking.

5.2.4 Football game

(160 images, 488x300 pixels, 15 objects, 0.2 frame/second).

Parameters: $\mu = 1.5$, $\beta = 2.5$ and $\gamma = 0.5$.

In this noisy sequence from PETS 2003, 13 players and 2 referees that are all dressed up with similar clothes (red, white and black) are tracked. As illustrated by Figure 17, the disappearance of one of the player from the image is well handled (see $t = 30$) and the players are correctly recovered after partial occlusions ($t = 30$ and $t = 75$).

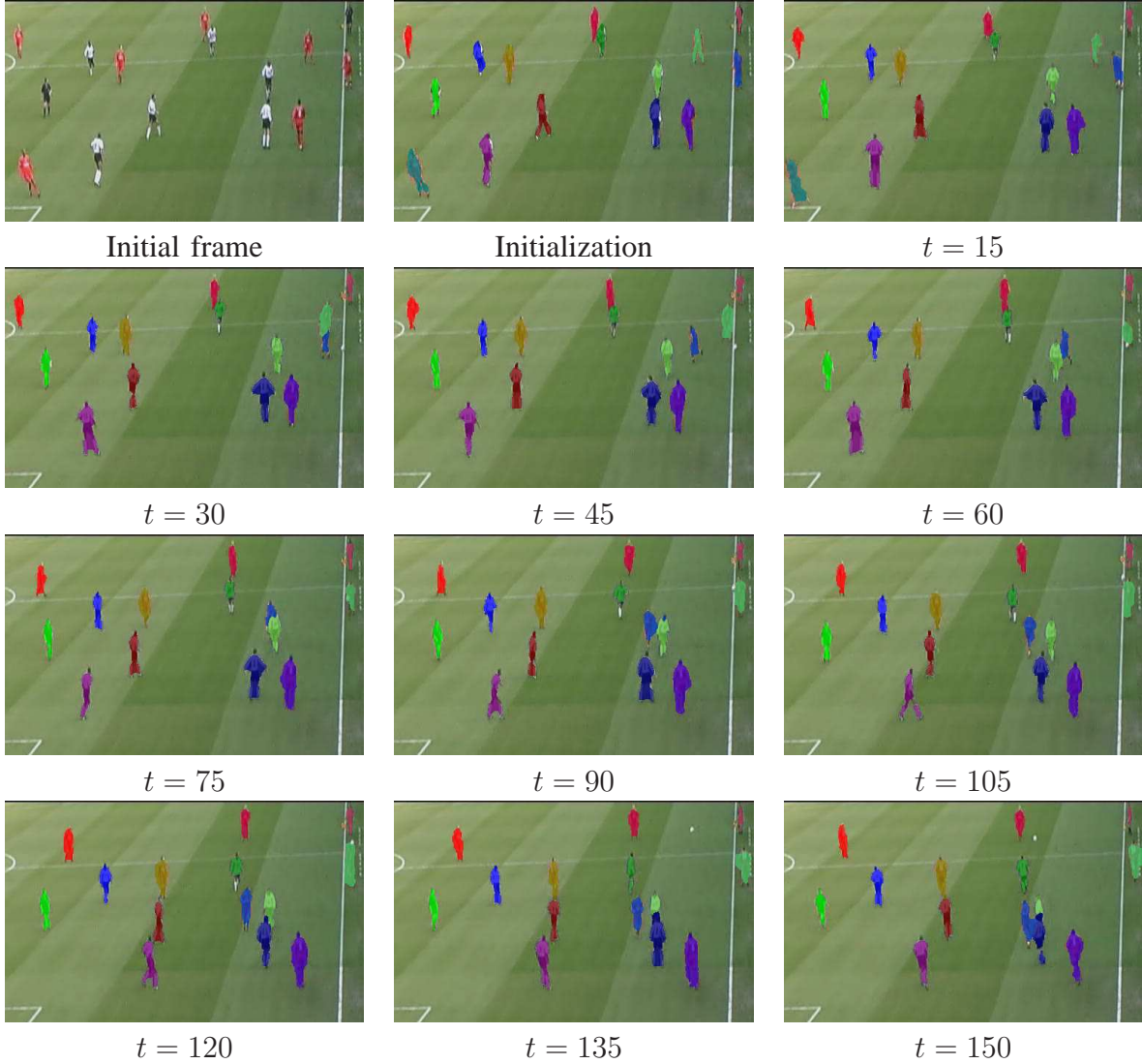


Fig. 17. Football sequence: $\mu = 1.5$, $\beta = 2.5$ and $\gamma = 0.5$. Only the visible parts of the tracked persons are presented. As illustrated by the first frame, the players of each team and the referees have similar colors. The algorithm is nevertheless able to track correctly each player.

For visual clarity, only the visible parts of the objects are shown. This example is quite simple, as the player motion is quite small. Since the method of [22] gives similar good results for this example, we did not show this experiment.

5.2.5 Man behind trees

(100 images, 360x288 pixels, 1 object, 4.1 frames/second).

Parameters: $\mu = 1.5$, $\beta = 2.5$ (*encourages occlusion instead of disappearance*), $\gamma = 0.5$.

The last sequence consists of a man wearing a (fortunately) red pullover and walking behind trees with a linear motion. This simple motion and the occluded parts representation enable the process to recover the target after the numerous partial and total occlusions. On the other hand, these occlusions make the segmentation quite rough (see Figure 18).

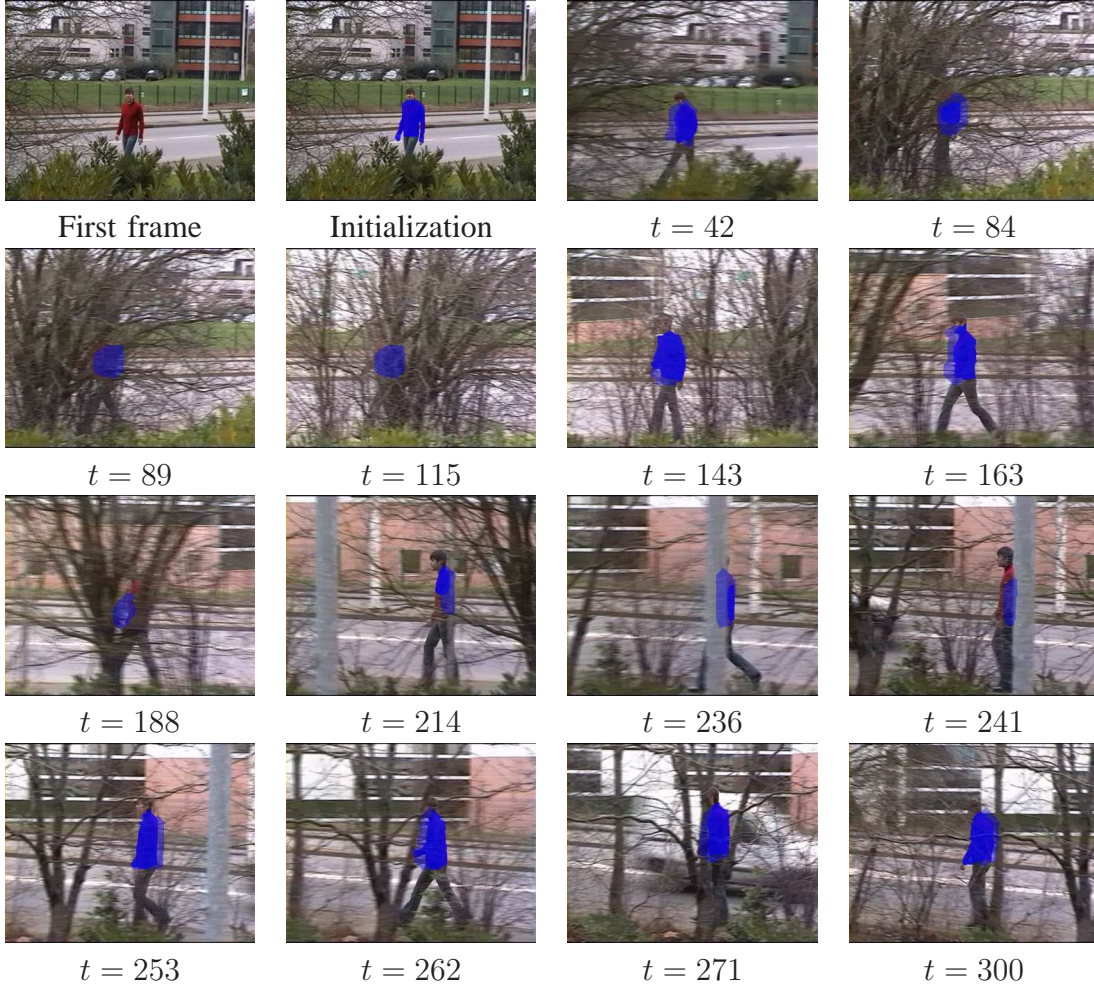


Fig. 18. **Man behind tree sequence:** $\mu = 1.5$, $\beta = 2.5$ and $\gamma = 0.5$. The dark blue indicates the visible part of the tracked object whereas the light blue denotes the occluded part. There are a lot of partial and total occlusions of the target, but the tracking process always recover the man.

One could point out that the red pullover is quite simple to recover, but we would like to underline that even with this fact, it is very difficult to track such an occluded object continuously in time. For instance, methods based on the color would segment the wall (which also contains a high level of red color) during occlusions. Concerning tracking methods as [23], the object is obviously lost after the first total occlusion, as illustrated by the Figure 19.



Fig. 19. **Man behind tree sequence - results with [23]:** $R_p = 0$, $\mu = 0$, $\beta_i = 0$ and $\gamma = 0.5$. The target is lost after the first total occlusion.

5.2.6 Tuning of parameters

In order to help using this method, we now give some hints about parameters tuning. The parameter α represents the weight of occluded area. If set to infinity, the model will only consider visible areas. The parameters μ (resp. β) denotes the appearing (resp. disappearing) areas, if set to infinity, the object size can only decrease (resp. increase). These three parameters usually take value in the range $[0, 3]$.

When the velocity of the object is high, the prediction can be bad, so the appearance parameter μ should be small to allow areas far from the prediction to be segmented. Another important point is the difference between the values of α and β . If β is bigger than α , then the model will encourage the disappearance of objects with respect to their occlusions.

One simple example for non deformable objects can finally be detailed. In this case the appearance and disappearance parameters μ and γ can be set to infinity and the object size will remain constant. The position of the object is then only determined by the prediction that varies with respect to the last parameter α . If the occlusion parameter α is also infinite, the object will always be fully visible so the motion model will determine the tracking and no corrections based on the image intensity will be done. In case of occlusions, the tracking of the objects will be lost. On the other hand, if α is small enough, the model will consider occluded areas and the motion will be computed only on the visible parts, allowing the process to deal with occlusions.

5.2.7 Computational cost

From these five experiments, we can see that the mean computational time of our non optimized algorithm is around 4 frames per seconds for the tracking of one object in images of size 360x300 with a standard desktop PC. Note that the computational cost of the graph cuts process is proportional to $2N$, N being the number of tracked objects. More precisely, the process requires N α -expansions for a cycle and at least 2 cycles of α -expansions (if there is more than one object) to converge. In our applications, we set the maximum number of cycles to 2 as it seems a good compromise between velocity and visual quality of the results. Let us also note that all the pixels of the images are processed. Considering only a narrow band around the predicted sets would naturally speed-up the process. The study of this band will be the subject of future works for real time implementation purpose.

6 CONCLUSION

In this work, we have formalized the notion of visible and occluded parts of an object in an original way. The corresponding energy function contains some new terms that allow tracking and segmenting these two parts of an object of interest. Moreover, this representation permits to naturally deal with the partial and total occlusions of interacting targets.

A lot of perspectives can be drawn upon. First of all, as in [6], some external detectors could be incorporated to make the tracking more robust in the case of persistent occlusions but also to handle the entrance of new objects in the scene. Unlike [6], these detections could be used as a set of pixels instead of a simple vertex.

Next, the velocity model could be improved in order to create better predictions for deformable objects. Moreover, the direction of the motion measurements could be taken into account in the function β to detect bad predictions. Another important point is the tuning of the parameters. A study of the ratio between object and image sizes could allow setting the regularization parameter to deal properly with small objects. The analysis of the velocity could also permit to define a narrow band around the predictions, which would help fixing the appearance parameter value and decrease the computational cost.

Finally, we would aim to impose some global constraints on each tracked object to enhance both visible and occluded segmentations. This could be done with a shape constraint, as in [30], or simply with a pre-determined range of size for each object. An alternative solution is to select a compact characterization of the shape (e.g., pose parameters [4], ellipse parameters [35], normalized central moments [44], or some top-down knowledge [28]). This will be studied in future works.

ACKNOWLEDGMENT

We would like to thank the two reviewers for their helpful suggestions and Vicent Caselles for the many useful discussions we had with him. We would also like to acknowledge the support received from the i3media Spanish project (CENIT 2007-1012) and the Torres Quevedo fellowship from the Ministerio de Educación y Ciencia Español.

APPENDIX

In this appendix, we detail, for all the terms of energy (6), the different graph cases associated to a current labeling λ during an expansion corresponding to the label α . This part enables to re-implement exactly the proposed algorithm.

Let $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ be a directed graph built for our graph cuts minimization problem. The set of vertices $\mathbf{V} = \{n_x\}$ corresponds to the set of pixels $x \in \mathcal{P}$, where $\mathcal{P} = \cup_{i=1}^N \mathcal{O}_i^t \cup \Omega^{t+1}$, with two additional distinguished terminal vertices $\{S, T\}$ called the source and the sink. The set of edges linking the vertices is denoted by \mathbf{E} . A cut $\mathcal{C} = \{\mathbf{V}^S, \mathbf{V}^T\}$ is a partition of the set of vertices such that $S \in \mathbf{V}^S$, $T \in \mathbf{V}^T$. The cost of the cut is the sum of the weights of the edges between a vertex in \mathbf{V}^S and a vertex in \mathbf{V}^T . A minimum cut is a cut with a minimum cost. It can be found by computing the maximal flow using the Ford and Fulkerson algorithm [13]. In this work, we use the algorithm presented in [2]. Any cut can be described by a set of binary variables $\{u_i\}_{i=1, \dots, m}$, one for each vertex in $\mathbf{V} = \{n_i\}_{i=1, \dots, m}$, so that $u_i = 0$ when $n_i \in \mathbf{V}^S$ and $u_i = 1$ when $n_i \in \mathbf{V}^T$. Thus, if the graph represents an energy, this energy can be viewed as a function of the m binary variables $\{u_i\}$.

We recall that the set of labels is a finite set $[0; N]$ with $\pi(y^i) \in \{0; 1\}$, for the pixels $y^i \in \mathcal{O}_i^t$ associated to the vertices n_{y^i} and $\lambda(x) \in [0; N]$ for the pixels of the image $x \in \Omega^{t+1}$ associated to the vertices n_x . To simplify the notations, let us introduce the whole labeling function $f = \{\lambda, \pi\}$. An energy $\mathcal{E}(f)$ corresponding to a labeling \tilde{f} within an α -expansion of f can be represented by the binary variables and a related energy E , through relations :

- Unary energies $\mathcal{E}_u(\tilde{f}(x))$, for $x \in \mathcal{P}$:

$$\mathcal{E}_u(\tilde{f}(x)) = E_u(0)(1 - u(n_x)) + E_u(1)u(n_x),$$

- Binary energies $\mathcal{E}_b(\tilde{f}(x), \tilde{f}(y))$, for $(x, y) \in \mathcal{P} \times \mathcal{P}$:

$$\begin{aligned} \mathcal{E}_b(\tilde{f}(x), \tilde{f}(y)) &= E_b(0, 0)(1 - u(n_x))(1 - u(n_y)) + E_b(0, 1)(1 - u(n_x))u(n_y) \\ &\quad + E_b(1, 0)u(n_x)(1 - u(n_y)) + E_b(1, 1)u(n_x)u(n_y). \end{aligned}$$

The binary energies are graph representable and can be minimized by graph cuts as soon as [20]:

$$E_b(0, 0) + E_b(1, 1) \leq E_b(0, 1) + E_b(1, 0). \quad (11)$$

As a consequence, we have to check that all our binary energies verify this necessary condition of submodularity. From the different possible values of $E_u(\cdot)$ and $E_b(\cdot, \cdot)$, corresponding edges are built in the graph (see [20] for construction details).

For a pixel $x \in \Omega^{t+1}$, the binary variable $u(n_x) = 1$ represents the cost of assigning the label α whereas $u(n_x) = 0$ is the cost of staying with its current label $\lambda(x)$. The situation is slightly modified for the vertices n_{y^i} corresponding to the pixels $y^i \in \mathcal{O}_i^t$, as we have to consider two cases: $\alpha = i$ and $\alpha \neq i$. Indeed, during an expansion α , the pixels y^i of the set \mathcal{O}_i^t relative to the object $i = \alpha$ can keep their current label (0 or 1) or move it to 1 (i.e., good prediction).

On the other hand, the pixels of the sets \mathcal{O}_i^t associated to the objects $i \neq \alpha$ can keep their current label (0 or 1) or move it to 0 (i.e., bad prediction). This last case is valid for all the sets \mathcal{O}_i^t ($i = 1 \dots N$), when the background is treated ($\alpha = 0$). In term of energy, we have:

- If $\alpha = i$, then $u(n_{y^i}) = 1$ represents the cost of labeling y^i as a good prediction, whereas $u(n_{y^i}) = 0$ is the cost that considers that the current label $\pi(y^i)$ will be conserved.
- If $\alpha \neq i$, then $u(n_{y^i}) = 1$ represents the cost of labeling y^i as a bad prediction, whereas $u(n_{y^i}) = 0$ is the cost that considers that the current label $\pi(y^i)$ will be conserved.

We can now discretize the different terms of the energy (6) for a current expansion $\alpha \in [0; N]$.

(A) Visible and appearance data term:

$\forall x \in \Omega^{t+1}$, the costs $\mathcal{E}_D(\lambda(x))$ and $\mathcal{E}_\gamma(\lambda(n_x))$ can be jointly represented by the energy $E_{D\gamma}$ associated to the vertex n_x with the following values: $E_{D\gamma}(1) = -\ln(P_\alpha(x)) + \gamma d_\alpha(x)$, $E_{D\gamma}(0) = -\ln(P_{\lambda(x)}(x)) + \gamma d_{\lambda(x)}(x)$.

(B) The terms including the prediction errors, the occlusions and the coherence are merged in one unique binary energy. $\forall y^i \in \mathcal{O}_i^t$ and $x = y^i + \tilde{v}_i^t \in \Omega^{t+1}$, the cost $\mathcal{E}_{\mu\beta C}^i(\pi(y^i), \lambda(x))$ corresponds to the energy $E_{\mu\beta C}^i$ with the following values:

(I) If $\pi(y^i) = 1$ (If the prediction of pixel y^i is good)

1) If $\alpha = i$ (If we are currently testing the label associated to the object i)

a) If $\lambda(x) = i$ (If the pixel x is well predicted and currently associated to object i)

$$E_{\mu\beta C}^i(0,0) = E_{\mu\beta C}^i(0,1) = E_{\mu\beta C}^i(1,0) = E_{\mu\beta C}^i(1,1) = 0.$$

b) else

$$E_{\mu\beta C}^i(0,0) = E_{\mu\beta C}^i(1,0) = \mu \tilde{d}_i(x) \text{ (Occlusion)}, E_{\mu\beta C}^i(0,1) = E_{\mu\beta C}^i(1,1) = 0 \text{ (Good prediction)}.$$

2) else ($\alpha \neq i$)

a) If $\lambda(x) = i$ (If the pixel x is well predicted and currently associated to object i)

$$E_{\mu\beta C}^i(0,0) = 0 \text{ (Good prediction)}, E_{\mu\beta C}^i(0,1) = \mu \tilde{d}_i(x) d_i(x) \text{ (Occlusion)}, E_{\mu\beta C}^i(1,1) = \beta_i(y^i) \text{ (Bad prediction)}, E_{\mu\beta C}^i(1,0) = +\infty \text{ (Impossible, } x \text{ can not be associated to the object } i \text{ if } y^i \text{ is a good prediction)}.$$

b) else

$$E_{\mu\beta C}^i(0,0) = E_{\mu\beta C}^i(0,1) = \beta_i(y^i) \text{ (Bad prediction)}, E_{\mu\beta C}^i(1,0) = E_{\mu\beta C}^i(1,1) = \mu \tilde{d}_i(x) \text{ (Occlusion)}.$$

(II) else ($\pi(y^i) = 0$, in this case $\lambda(x) = i$ is impossible)

1) If $\alpha = i$

$$E_{\mu\beta C}^i(0,0) = \beta_i(y^i) \text{ (Bad prediction)}, E_{\mu\beta C}^i(1,0) = \mu \tilde{d}_i(x) \text{ (Occlusion)}, E_{\mu\beta C}^i(1,1) = 0 \text{ (Good prediction)}, E_{\mu\beta C}^i(0,1) = +\infty \text{ (Impossible, } x \text{ can not be associated to the object } i \text{ if } y^i \text{ corresponds to a good prediction)}.$$

2) else ($\alpha \neq i$)

$$E_{\mu\beta C}^i(0,0) = E_{\mu\beta C}^i(0,1) = E_{\mu\beta C}^i(1,0) = E_{\mu\beta C}^i(1,1) = \beta_i(y^i).$$

We omit the regularization energies of terms (D) and (E) as it is something classical in graph construction. One can check that all these energies are submodular, in the sense of [20], by checking relation (11) for each case.

Once the graph has been created and cut, following the algorithm of [2], the values of the binary variables $u(n_x)$ and $u(n_y)$ are obtained and the expansion $\tilde{\lambda}$ is created as follows:

- for all $x \in \Omega^{t+1}$, we associate the label $\tilde{\lambda}(x) = \alpha$ when $u(n_x) = 1$ (n_x is associated to the sink) and $\tilde{\lambda}(x) = \lambda(x)$ when $u(n_x) = 0$,
- for all $y^i \in \mathcal{O}_i^t$, with $i = \alpha$, we associate the label $\tilde{\pi}(y^i) = 1$ when $u(n_{y^i}) = 1$ (n_{y^i} is associated to the sink) and $\tilde{\pi}(y^i) = \pi(y^i)$ when $u(n_{y^i}) = 0$,
- for all $y^i \in \mathcal{O}_i^t$, with $i \neq \alpha$, we associate the label $\tilde{\pi}(y^i) = 0$ when $u(n_{y^i}) = 1$ (n_{y^i} is associated to the sink) and $\tilde{\pi}(y^i) = \pi(y^i)$ when $u(n_{y^i}) = 0$.

The update of the visible and occluded parts is then done such that:

- for all $x \in \Omega^{t+1}$, if $\lambda(x) = i$, then $x \in \mathcal{V}_i^{t+1} \subset \mathcal{O}_i^{t+1}$,
- for all $y^i \in \mathcal{O}_i^t$, if $\pi(y^i) = 1$, then $x = y^i + \bar{v}_i^t \in \mathcal{O}_i^{t+1}$, else $x = y^i + \bar{v}_i^t \notin \mathcal{O}_i^{t+1}$.

In a so-called "cycle", this process is applied once for all $\alpha \in [0; N]$. Cycles are then repeated until convergence. In practice, the process converges most of the time in two cycles (just one cycle is needed if there are no occlusions between objects). A sketch of the graph for a current α -expansion is given in Figure 20.

Remark: We would like to underline that despite the similarity with the energy minimized in the work of [22], [23], the overall process has been more inspired from [6] and [19], by adding additional vertices to the classical graph [6] and combining the principle of active vertices (good and bad predictions) as well as the binary function that models occlusions and rejects impossible labeling (see cases (I)2a and (II)1 of energies $E_{\mu\beta C}^i$) [19].

Current expansion: $\alpha = 1$

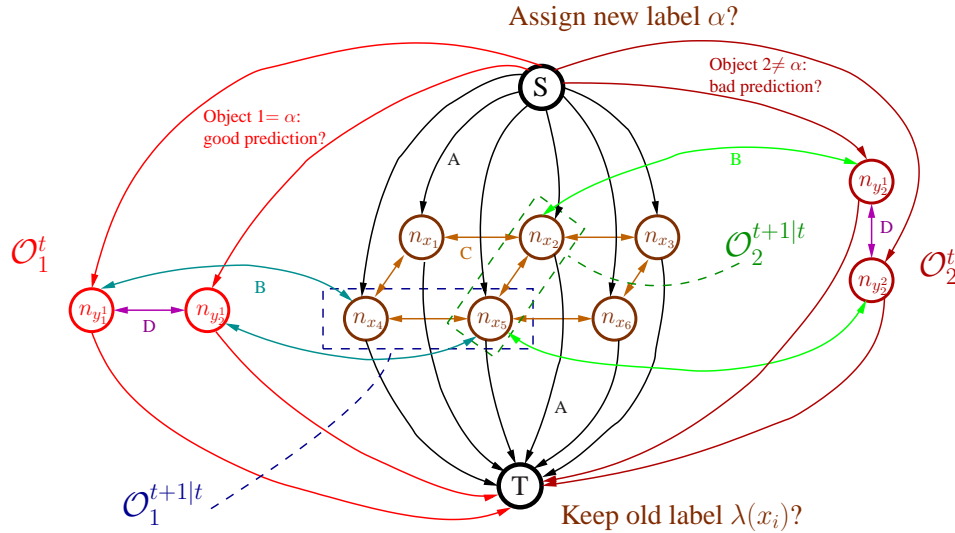


Fig. 20. The current expansion corresponds to $\alpha = 1$. Each vertex corresponding to the pixel $x_i \in \Omega^{t+1}$ of the current image can keep its old label $\lambda(x_i)$ (if the edge from the vertex n_{x_i} to the sink T is cut) or take the new label α (if the edge from the vertex n_{x_i} to the source S is cut). Concerning the predicted sets, we have to differentiate the object 1 ($=\alpha$) and the object 2 ($\neq \alpha$). For the object 1, that corresponds to the current expansion, the vertices $n_{y_j^1}$ corresponding to pixels $y_j^1 \in \mathcal{O}_1^t$, $j = 1, 2$, take the label $\pi(y_j^1) = 1$ if, at the end of the cut, they are associated with the source S. On the other hand, for the object 2, the vertices $n_{y_j^2}$ corresponding to the pixels $y_j^2 \in \mathcal{O}_2^t$, $j = 1, 2$, take the label value $\pi(y_j^2) = 0$, if at the end of the cut, they are associated with the source S. Note that, when the current expansion corresponds to the background ($\alpha = 0$), all the objects act like object 2 in the current example. In this graph, the different energies are also illustrated, the black edges linking the vertices representing the pixel of the image to the sink correspond to the term (A). The green and blue edges linking the prediction and the current segmentation ($\mathcal{O}_i^{t+1|t}$ and \mathcal{O}_i^t , $i = 1, 2$) represent the term (B). The last edges correspond to the two spatial regularizations: in brown the image regularization (term C) and in purple the prediction regularization (term D).

REFERENCES

- [1] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *IEEE Int. Conf. Comp. Vis. (ICCV'01)*, volume 1, pages 105–112, 2001.
- [2] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. on Pat. Anal. and Mach. Intell.*, 26(9):1124–1137, 2004.
- [3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pat. Anal. Mach. Intell.*, 23(11):1222–1239, 2001.
- [4] M. Bray, P. Kohli, and P. Torr. Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *Europ. Conf. on Com. Vis. (ECCV'06)*, 2006.
- [5] A. Bugeau and P. Pérez. Track and cut: simultaneous tracking and segmentation of multiple objects with graph cuts. In *Proc. Int. Conf. Comp. Vis. Theory and Appl. (VISAPP'08)*, volume 2, pages 447–454, 2008.
- [6] A. Bugeau and P. Pérez. Track and cut: simultaneous tracking and segmentation of multiple objects with graph cuts. *EURASIP J. on Image and Video Proces.*, 2008:1–14, 2008.
- [7] T. Corpetti, E. Mémin, and P. Pérez. Dense estimation of fluid flows. *IEEE Trans. on Pat. Anal. and Mach. Intell.*, 24(3):365–380, 2002.
- [8] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs. A maximum likelihood stereo algorithm. *Comp. Vis. Image Underst.*, 63(3):542–567, 1996.
- [9] D. Cremers. Dynamical statistical shape priors for level set-based tracking. *IEEE Trans. on Pat. Anal. and Mach. Intell.*, 28(8):1262–1273, 2006.
- [10] A. Criminisi, A. Blake, C. Rother, J. Shotton, and P. H. Torr. Efficient dense stereo with occlusions for new view-synthesis by four-state dynamic programming. *Int. J. Comput. Vision*, 71(1):89–110, 2007.
- [11] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov. Bilayer segmentation of live video. In *IEEE Conf. Comp. Vis. Pat. Rec. (CVPR'06)*, volume 1, pages 53–60, 2006.
- [12] S. Dambreville, Y. Rath, and A. Tannenbaum. Tracking deformable objects with unscented kalman filtering and geometric active contours. *American Control Conf.*, 2006.
- [13] L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian J. of Mathematics*, 8:399–404, 1956.
- [14] D. Freedman and M. Turek. Illumination-invariant tracking via graph cuts. In *IEEE Conf. Comp. Vis. Pat. Rec. (CVPR'05)*, volume 2, pages 10–17, 2005.
- [15] D. Freedman and T. Zhang. Motion detection and estimation - active contours for tracking distributions. *IEEE Trans. on Image Proces.*, 13(4):518–526, 2004.
- [16] C. Harris and M. Stephens. A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vis. Conf.*, pages 147–151, 1988.
- [17] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *Int. J. of Comp. Vis.*, 29(1):5–28, 1998.
- [18] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross, and C. Rother. Bi-layer segmentation of binocular stereo video. In *IEEE Conf. Comp. Vis. Pat. Rec. (CVPR'05)*, volume 2, pages 407–414, 2005.
- [19] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions via graph cuts. In *IEEE Int. Conf. Comp. Vis. (ICCV'01)*, volume 2, pages 508–515, 2001.
- [20] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. on Pat. Anal. Mach. Intell.*, 26(2):147–159, 2004.
- [21] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereovision. In *Int. Joint Conf. on Artificial Intell. (IJCAI)*, pages 674–679, 1981.
- [22] J. Malcolm, Y. Rath, and A. Tannenbaum. Multi-object tracking through clutter using graph cuts. In *IEEE Int. Conf. Comp. Vis. (ICCV'07)*, 2007.
- [23] J. Malcolm, Y. Rath, and A. Tannenbaum. Tracking through clutter using graph cuts. In *Brit. Mach. Vis. Conf. (BMVC'07)*, 2007.
- [24] C. Mota, I. Stuke, T. Aach, and E. Barth. Spatial and spectral analysis of occluded motions. *Signal Processing: Image Communication*, 20:529–536, 2005.
- [25] H. T. Nguyen and A. W. M. Smeulders. Fast occluded object tracking by a robust appearance filter. *IEEE Trans. on Pat. Anal. and Mach. Intell.*, 26(8):1099–1104, 2004.
- [26] M. Niethammer and A. Tannenbaum. Dynamic geodesic snakes for visual tracking. In *IEEE Conf. Comp. Vis. Pat. Rec. (CVPR'04)*, volume 1, pages 660–667, 2004.
- [27] N. Papadakis and É. Mémin. A variational method for the tracking of curve and motion. *J. of Math. Imag. and Vis.*, 31(1):81–103, 2008.
- [28] D. Ramanan. Using segmentation to verify object hypotheses. In *IEEE Conf. Comp. Vis. Pat. Rec. (CVPR'07)*, 2007.
- [29] Y. Rath, N. Vaswani, A. Tannenbaum, and A. J. Yezzi. Particle filtering for geometric active contours with application to tracking moving and deforming objects. *IEEE Trans. on Pat. Anal. Mach. Intell.*, 29(8):1470–1475, 2007.
- [30] F. Schmidt, E. Aarts, D. Cremers, and Y. Boykov. Efficient shape matching via graph cuts. In *Energy Minimization Methods in Comp. Vis. Pat. Rec. (EMMVCPR'07)*, pages 39–54, 2007.
- [31] D. Terzopoulos and R. Szeliski. Tracking with kalman snakes. *Active vision*, pages 3–20, 1993.
- [32] N. Xu, R. Bansal, and N. Ahuja. Object segmentation using graph cuts based active contours. *Comp. Vis. and Image Underst.*, 107(3):210–224, 2007.
- [33] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Comp. Surv.*, 38(4), 2006.
- [34] W. Yu, G. Sommer, S. Beauchemin, and K. Daniilidis. Oriented structure of the occlusion distortion: Is it reliable? *IEEE Trans. on Pat. Anal. Mach. Intell.*, 24:1286–1290, 2002.
- [35] L. Zhao and L.S. Davis. Closely coupled object detection and segmentation. In *IEEE Int. Conf. Comp. Vis. (ICCV'05)*, 2005.



Nicolas Papadakis was born in 1981 in France. He currently has a post-doctoral position in the foundation Barcelona Media in relation with the Univerity Pompeu Fabra in Barcelona, Spain. He graduated in 2004 from the National Institute of Applied Sciences (INSA) of Rouen in Applied Mathematics and received the Ph.D. degree in Applied Mathematics from the University of Rennes, France, in 2007. His main research interests are tracking, depth estimation and motion analysis.



Aurélie Bugeau received her Ph.D. degree in signal processing from the University of Rennes, France, in 2007. Since November 2007, she has been holding a post-doctoral position in the foundation Barcelona Media, Barcelona, Spain. Her main research interests include objects detection and tracking, data clustering, image and video inpainting.