Pattern Recognition Letters 33 (2012) 765-774

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Spatio-temporal target-measure association using an adaptive geometrical approach

Abir El Abed, Séverine Dubuisson*, Dominique Béréziat

Laboratoire d'Informatique de Paris 6, Université Pierre et Marie Curie, 4 place Jussieu, 75005 Paris, France

ARTICLE INFO

Article history: Received 30 July 2010 Available online 29 November 2011 Communicated by S. Sarkar

Keywords: Data association Target tracking State and measure spaces

ABSTRACT

Data association is of crucial importance to improve target tracking performance in many complex visual environments (non-linear dynamics, occlusions, *etc*). Usually, association effectiveness is based on prior information and observation category. However, association becomes difficult if targets are similar. Problems also arise in cases of missing data, complex motions or deformations over time. To remedy, we propose a new method for data association, that uses the evolution of the dynamic model of targets. The main idea is to measure an adaptive geometric accuracy between possible trajectories of targets, by only using positions as information, that constitutes its main advantage.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Traditionally, multiple target tracking consists in estimating states of moving targets from their associated available measures to get a track (Vermaak et al., 2005). The main difficulty is to determine over time the correct assignments of given measures to a target whose model is generally unknown. The literature contains some classical approaches to solve the problem of data association: we distinguish deterministic approaches from probabilistic ones.

Deterministic approaches determine the best associations by using a score function (Vermaak et al., 2005). The simplest method might be the nearest-neighbor standard filter (NNSF) (Rong and Bar-Shalom, 1996) that selects the closest valid measure to a predicted target and uses it for its state estimation. The proximity is generally given by the Mahalanobis distance. This filter has been used in many simple target tracking problems (Song et al., 2005; Konstantinova et al., 2003). But since it does not take into account the possibility of incorrect associations, its performance might be poor in some cases (especially for multi-target tracking, for example when two measures are equidistant from the same target or if target tracks cross) inducing association errors and degrading performance over time.

Probabilistic approaches are based on *posterior* probability estimation and make an association decision using a probability error (Rasmussen and Hager, 2001). Probabilistic data association filter (PDAF) (Cox, 1993; Singer et al., 1974) is an extension of the Kalman Filter (Bar-Shalom and Fortman, 1988) and an approximation of the optimal Bayesian filter. It considers two assumptions: (i) a

E-mail address: severine.dubuisson@lip6.fr (S. Dubuisson).

measure is associated with a track or is a false alarm, and (ii) a track generates at most one measure. Here, a probability of association between the target and each measure is computed and used to weight the measures for the track update. Because of its simplicity, PDAF was used in target tracking problems (Jaward et al., 2006; Yilmaz et al., 2006), but its formulation has some limitations. First, it assumes that all measures come from the track being updated, that is not true in case of dense target conditions. Secondly, PDAF gives poor performances if the signal to noise ratio is small: in such cases, for example, the "good" measure can be located outside the validation region, and the filter cannot estimate the state of the target. Joint probabilistic data association filter (JPDAF) (Fortmann et al., 1983; Bar-Shalom and Li, 1995) is an extension of PDAF to the multiple target case (this number is known and constant): it works by running one PDAF per target, under the same hypothesis than PDAF, enforces an exclusion principle that prevents two or more trackers from latching onto the same target by calculating target-measure association probabilities jointly. JPDAF computes all possible joint events and their probabilities. Then, the algorithm determines to which of the marginal events this joint event contributes. JPDAF uses a weighted sum for all measures near the predicted state, each weight corresponding to the posterior probability for a measure to come from a target. JPDAF is widely used for target tracking because it deals with complex cases (Rasmussen and Hager, 2000; Chen et al., 2001) and provides an optimal solution in the Bayesian framework. However, the number of possible hypothesis increases exponentially with the number of targets, requiring prohibitive computation times, and tracks cannot be deleted or created. Multiple hypothesis tracking (MHT) was proposed by Reid (1979) for multiple target tracking. The number of targets is unknown, and varies over time. Reid's algorithm defines





^{*} Corresponding author. Fax: +33 0 1 44 27 74 95.

^{0167-8655/\$ -} see front matter \odot 2011 Elsevier B.V. All rights reserved. doi:10.1016/j.patrec.2011.11.018

a systematic way in which multiple data association hypotheses are made and evaluated for the problem of multiple targets dealing with false alarms and cluttered background. When new data are available, each hypothesis is expanded into a set of new hypotheses by considering all association possibilities, that are propagated, implying the computation of each possible hypothesis, updated for each new available measure. To avoid an exponential increasing of the complexity of the algorithms, MHT eliminates less probable hypotheses, merges similar ones and keeps most probable. MHT is a recursive algorithm and easy to implement that consists in creating a tree: each hypothesis gives a path in this tree. This approach is very efficient because it considers all possible data association hypotheses, adds, removes or merges tracks, and in cases of conflicts, waits for more measures to decide for data association (Vermaak et al., 2005; Gelgon et al., 2005), and is then less sensible to false alarms. However, this exhaustive enumeration of hypotheses makes this approach very slow and costly. Probabilistic multiple hypothesis tracker (PMHT) (Gauvrit et al., 1997) supposes a target generates zero or several measures, and, contrary to the previous approaches, makes the posterior estimation of the state once all the measures are available, making it inappropriate for online tracking applications.

Usually, association effectiveness is based on prior information and available measures. In cases of lack of prior information, insufficient knowledge about the dynamics, large interval of time between two available observations, deformable or non distinguishable target (i.e. same color or same shapes), the association becomes difficult. Likewise, when only dealing with target positions, and if a measure is equidistant from several targets, all target association probabilities get close values, also making difficult the data association. No association method can efficiently handle all the cases previously illustrated.

In this paper, we propose a novel method for data association based on a geometric criteria, called *energetic association filter* (EAF). The idea is to define a term *E*, inspired by the evolution of the target dynamical model, from which, we deduce the probability of association between a given target and all available measures. The main advantages are it requires few parameters and prior knowledges: only information about target positions in the image plan are used. Besides, it handles the problem of association if a measure falls within the validation region of several targets and is equidistant from them. The outline of this paper is as follows. In Section 2, we introduce the notations used throughout the paper. In Section 3, we expose our approach for data association (EAF), derive its geometrical representation and its mathematical formulation. The proposed method is then tested on several sequences in Section 5. Finally, concluding remarks and perspective works are given in Section 6.

2. Definitions and notations

In this paper, k = 1, ..., K is the discrete time, l = 1, ..., L is the index associated to a measure and m = 1, ..., M the index associated to a target. The set of targets at time k is given by $\{x_k^1, ..., x_k^M\}, x_k^m \subset \mathcal{X}$, the state space. We mainly deal with predictions, because we focus on a missing data problem without knowledges about the real dynamical model: we then work into an optimal filtering framework, requiring prediction and correction steps: the prediction of target x_k^m is denoted \hat{x}_k^m . At each time k, a sensor delivers zero or one observation containing at least one measure, giving the set $\{y_k^1, ..., y_k^L\}, y_k^I \subset \mathcal{Y}$, the measure space. Observations are supposed to be i.i.d. samples of a Gaussian distribution. A target is characterized by its own dynamic model. We then dispose M models one for each target, plus one model for false alarms, and then have M + 1 tracks. In this paper, we also make the following assumptions (Bar-Shalom and Fortman, 1988):

- 1. A measure y_k^l is associated to a track or considered as false alarm (false-track);
- 2. A track contains at most one measure at time *k*.

The global complexity for data association is $\mathcal{O}(L \times (M + 1))$. To reduce the computation time, some algorithms introduce a validation region (Bar-Shalom and Fortman, 1988), that only contains measures with strong association probabilities, and exclude the others. This produces an ellipsoid (Cox, 1993) whose center is the predicted measure, and size and direction of axes are respectively given by the eigenvalues and by their associated eigenvectors of the covariance matrix. Then y_k^l belongs to the validation region \mathcal{V}_k^m of predicted target \hat{x}_k^m if:

$$\left[\boldsymbol{y}_{k}^{l}-\hat{\boldsymbol{y}}_{k}^{m}\right]^{T}\left(\boldsymbol{\Sigma}_{k}^{m}\right)^{-1}\left[\boldsymbol{y}_{k}^{l}-\hat{\boldsymbol{y}}_{k}^{m}\right] \leqslant \gamma^{2} \tag{1}$$

where \hat{y}_k^m is the predicted measure, defined by the projection of the predicted state \hat{x}_k^m into the measure space (see Eq. (3)), $(y_k^l - \hat{y}_k^m)$ is the estimation error between the predicted measure \hat{y}_k^m and the current measure y_k^l , and Σ_k^m its covariance matrix (Therrien, 1989). Fig. 1(a) shows the predicted measures \hat{y}_k^m (black circles), after their projection into the measure space, and their validation regions (ellipses). The measures y_k^l , with $l = 1, \dots, 4$ (gray squares), that belong to these regions are candidate to data association.

3. Energy association filter (EAF)

Generally, an effective data association method is based on measures. A sensor provides various measures such as shapes, colors, positions, directions, *etc.* Furthermore, if the measure is limited to the position, falls inside the validation region of several targets and is equidistant from them, an association ambiguity will occur if we use NNSF or JPDAF. Moreover, targets can be quite similar: even if information about the color or shape is available, the association task is difficult, even impossible in case of complex dynamics.

In this section, we describe our contribution for data association restricted to one category of measures: positions. Furthermore, it only needs few prior information concerning targets: exclusively the two anterior predicted positions are used as input for our algorithm. We first explain the concept of our approach before giving its mathematical formulation.

3.1. Concept

Prediction is made using a dynamic model which parameters are generally known or learned from a training sequence to represent plausible motions, such as constant velocity or damped oscillations (Blake and Isard, 1998; North et al., 2000). For complex dynamics, such as non-constant velocities or non-periodic oscillations, the choice of the parameters for the dynamic model is difficult. Furthermore, the learning step becomes impossible in cases of missing data (dynamics between two successive measures are unknown). For these reasons, the parameters of our dynamic model are set in an adaptive and automated way once measures are available (El Abed et al., 2006).

Predicted state \hat{x}_{k+1}^m of target *m* and its corresponding measure (i.e. after projection into the measure space) \hat{y}_{k+1}^m at instant k + 1 are given by:

$$\hat{x}_{k+1}^m = f_{k+1}\left(\hat{x}_k^m\right) + V_{k+1} \tag{2}$$

$$\hat{y}_{k+1}^m = h(\hat{x}_{k+1}^m) + W_{k+1} \tag{3}$$

where f_{k+1} models the dynamics of the state at time *t* and *h* is the projection of the state into the measure space. V_{k+1} and W_{k+1} are respectively the state and measure noises. Consider now an



Fig. 1. (a) Validation regions. Projection of three predicted states into the measure space to give $\{\hat{y}_k^m\}_{m=1,2,3}$ (black circles) and their validation regions (ellipses). The measures $\{y_k^i\}_{i=1,...,4}$ (gray squares), that belong to these regions are candidate to data association. (b) The two computed predicted measures: $(\hat{y}_{k+1}^m)^{\#1}$ is obtained by using the initial dynamic model (prediction "without association"), and $(\hat{y}_{k+1}^m)^{\#2}$ is obtained by using the updated dynamic model (prediction "after association"). Dashed lines correspond to the estimated tracks without and after association.

available observation at instant k, containing one measure y_k^l , equidistant from several predicted measures. We suggest to determine the association probability at instant k by predicting at instant k+1 the measure associated with predicted target \hat{x}_{k+1}^m in two different ways.

Prediction "without association" (#1). The new state is predicted into the state space without taking into account any available measure (i.e. no measure comes from the target: we use Eq. (2)). The prediction of the state $(\hat{x}_{k+1}^m)^{\#1}$, and its projection into the measure space $(\hat{y}_{k+1}^m)^{\#1}$ (i.e. predicted measure) are then given by:

$$(\hat{\mathbf{x}}_{k+1}^m)^{\#1} = f_{k+1}(\hat{\mathbf{x}}_k^m) + \mathbf{V}_{k+1} \tag{4}$$

$$\left(\hat{y}_{k+1}^{m}\right)^{\#1} = h\left(\left(\hat{x}_{k+1}^{m}\right)\right) + W_{k+1} \tag{5}$$

Prediction "after association" (#2). The new measure $(\hat{y}_{k+1}^m)^{\#2}$ is predicted into the measure space by associating the measure y_k^l to the previous predicted measure (i.e. by taking into account the available measure). This prediction is given into the measure space by:

$$\left(\hat{y}_{k+1}^{m}\right)^{\#2} = f_{k+1}^{\#2} \left(y_{k}^{l}\right) + V_{k+1}^{\#2} \tag{6}$$

 $f_{k+1}^{\#2}$ is the dynamic model into the measure space, and $V_{k+1}^{\#2}$ a Gaussian noise. This adaptive modeling will be explained in next subsection.

Our goal is to analyze the effect of an adaptive parametrization of the dynamic model of the state on its future prediction. The comparison of predictions "without association" and "after association" gives the influence of the available measure on the predicted track. In Fig. 1(b), we show an example of the predicted measure obtained with these two models at different times. At times k - 1and k - 2, the same dynamic model (the initial one f_{k+1}) is used to predict the state of the target (see Eq. (4)). At time k + 1, two predictions are made: one without association, and one after association of measure y_{k}^{l} with the target it comes from. During the interval]k, k + 1[, we can see the parametrization of the dynamic model (i.e. use of the current measure) significantly changes the predicted track. Our approach then consists in comparing these two trajectories to determine the measure-target association probability. In next subsections, we derive the mathematical model of EAF.

3.2. Data association algorithm

We define a term inspired from the evolution of the target's dynamic model, described in terms of displacements in the state space. This term provides the measure-target association probability, and requires the predicted measures at times k - 1 and k - 2. We suppose only one sensor observes the dynamic scene, and can deliver at most one observation at each time, containing at least one measure. Each measure can be associated with a target, or be a false alarm. The score for the association of predicted measure \hat{y}_k^m with measure y_k^l is given by a linear combination of three terms:

$$E(\hat{y}_{k}^{m}, y_{k}^{l}) = \frac{1}{3} \sum_{i=1}^{3} \alpha_{i} E^{i}(\hat{y}_{k}^{m}, y_{k}^{l})$$
(7)

 α_i is a weighting factor introduced to emphasize the relative importance attached to each term E^i and is given by:

$$\alpha_i = \frac{1}{\sum\limits_{m=1}^{M} E^i(\hat{y}_k^m, y_k^l)}$$
(8)

Each term deals with one category of movement into the observation space. First term E^1 only considers linear translations cases and corresponds to the Mahalanobis distance: it is inappropriate in cases of complex dynamics (non linear displacements, oscillatory motions, non-constant velocities, *etc.*). To remedy, we add a second term E^2 which measures the geometric accuracy between two dynamic models (i.e. "without association" and "after association"), and indicates their closeness. Finally, to distinguish some cases, such as occlusions between targets, we add a third term E^3 , indicating the proximity of evolution. The global term *E* is only computed for a measure that falls within several validation regions (measures not included in any validation region of targets are considered as clutters). In our case, the validation region corresponds to the one described in Section 2. We now detail subsequently the three parts of *E*.

• Distance term E¹ (case of linear translations).

 E^1 corresponds to the Mahalanobis distance between the measure and its prediction at instant k, and is given by:

$$E^{1}(\hat{y}_{k}^{m}, y_{k}^{l}) = \sqrt{\left(y_{k}^{l} - \hat{y}_{k}^{m}\right)^{T} \left(\Sigma_{k}^{m}\right)^{-1} \left(y_{k}^{l} - \hat{y}_{k}^{m}\right)}$$
(9)

where $\hat{y}_k^m = h(\hat{x}_k^m) + W_k$, and Σ_k^m is the covariance matrix. This distance leads to association errors if measures and predictions are equidistant, or in cases of crossed trajectories.

• Evolution term E² (case of complex dynamics)

To consider the case of complex dynamics, such as oscillatory motions or non-constant velocities, we add the evolution term $E^2(\hat{y}_k^m, y_k^l)$. It introduces the notion of geometric accuracy between two sets of features whose evolutions are different. These dynamics are:

- 1. The current dynamic model, that predicts the measure $(\hat{y}_{k+1}^m)^{\#1}$ without considering the new available measure y_k^l , i.e. without updating the parameters of the dynamics (Eq. (5));
- 2. The updated dynamic model, that considers the measure y_k^l is generated by *m*th target and updates the parameters of its dynamic model to predict the new measure $(\hat{y}_{k+1}^m)^{\#2}$ of the target (Eq. (6)).

 $E^2(\hat{y}_k^m, y_k^l)$ measures the geometric closeness between the trajectories of two dynamic models. Our idea is to evaluate the parameters of the dynamic model in two cases: if the measure y_k^l comes from this target or not. We compute predicted measures $(\hat{y}_{k+1}^m)^{\#1}$ and $(\hat{y}_{k+1}^m)^{\#2}$, and then determine S_1 , intersection surface between the two circumscribed circles of triangles $(\hat{y}_{k-2}^m, \hat{y}_{k-1}^m, \hat{y}_k^m)$ and $(\hat{y}_{k-1}^m, \hat{y}_k^m, (\hat{y}_{k+1}^m)^{\#1})$, and S_2 , intersection surface between the two circumscribed circles of triangles $(\hat{y}_{k-2}^m, \hat{y}_{k-1}^m, y_k^m)$ and $(\hat{y}_{k-1}^m, y_k^l, (\hat{y}_{k+1}^m)^{\#2})$, see Fig. 2(a). $E^2(\hat{y}_k^m, y_k^l)$ is given by:

$$E^{2}(\hat{y}_{k}^{m}, y_{k}^{l}) = |S_{1} - S_{2}|$$
(10)

By comparing these two sets we measure the ratio of similarity, defined by $R_s = 1 - \alpha_2 E^2(\hat{y}_k^m, y_k^l)$, between the predictions at k + 1 given by two different dynamic models for *m*th target. The similarity between dynamic models is then maximal if $E^2(\hat{y}_k^m, y_k^l)$ is minimal (i.e. surfaces S_1 and S_2 are similar). Decreasing

 $E^2(\hat{y}_k^m, y_k^l)$ means that prediction \hat{y}_k^m and measure y_k^l are close, so are the two predictions $(\hat{y}_{k+1}^m)^{\#1}$ and $(\hat{y}_{k+1}^m)^{\#2}$ and their associated dynamic models, showing y_k^l is generated by target $x_k^m \cdot E^2(\hat{y}_k^m, y_k^l)$ is considered reliable if \hat{y}_k^m and y_k^l are on the same side comparing to axis $(\hat{y}_{k-2}^m, \hat{y}_{k-1}^m)$ (see left part of Fig. 2(b)). If \hat{y}_k^m and y_k^l are diametrically opposite or if their positions are in opposite side comparing to the axis $(\hat{y}_{k-2}^m, \hat{y}_{k-1}^m)$, both surfaces S_1 and S_2 are quite similar, which implies $E^2(\hat{y}_k^m, y_k^l)$ to be close to zero (see right part of Fig. 2(b)). In such cases, corresponding to trajectory crosses, $E^2(\hat{y}_k^m, y_k^l)$ is not sufficient to solve the association problem. We then incorporate a third term $E^3(\hat{y}_k^m, y_k^l)$.

• Evolution of proximity term E³

The evolution of proximity term, $E^3(\hat{y}_k^m, y_k^l)$, is the inverse of the surface *S* defined by the common area between triangles $(\hat{y}_{k-2}^m, \hat{y}_{k-1}^m, y_k^l)$ and $(\hat{y}_{k-2}^m, \hat{y}_{k-1}^m, \hat{y}_k^m)$ (see Fig. 2(c), dashed area). This term evaluates the absolute accuracy between prediction \hat{y}_k^m and measure y_k^l at instant *k*, and is given by:

$$E^{3}(\hat{y}_{k}^{m}, y_{k}^{l}) = \frac{1}{S}$$
(11)

Increasing *S* means that \hat{y}_k^m and y_k^l are close at instant *k* (see Fig. 2(c)). In the right part of this figure, we now have two predictions at instant *k*, $(\hat{y}_{k+1}^m)^{\#1}$ and $(\hat{y}_{k+1}^m)^{\#2}$, both equidistant from the



Fig. 2. Geometrical representation of *E* terms. (a) Intersection surfaces S_1 and S_2 ; (b) Differences between surfaces S_1 and S_2 extracted from the two dynamic models. (c) Left: intersection surface *S* (dashed area); right: case of two predicted measures $(\hat{y}_{k+1}^m)^{\#1}$ and $(\hat{y}_{k+1}^m)^{\#2}$ equidistant from y_k^l .

observation y_k^l . If we only use the distance to compute the proximity term, we get that both models have the same degree of similarity with the one defined by the dynamic model of points $(\hat{y}_{k-2}^m, \hat{y}_{k-1}^m, y_k^l)$. This leads to a contradiction with the reality, that can be explained by the fact that if they have both the same degree of similarity with the second dynamic model, their corresponding targets have then the same dynamics. For this reason, we have chosen to evaluate the similarity by measuring the intersection surface between triangles extracted from previous prediction, to add an information about the previous similarity of the models.

3.3. Measure-target posterior association probability

The posterior association probability between the available measure y_k^l and the predicted measure \hat{y}_k^m for target state x_k^m is given by:

$$\beta_{ml} = 1 - E(\hat{y}_k^m, y_k^l) \tag{12}$$

The posterior association probability that target state x_k^m is undetected is given by:

$$\beta_{m0} = 1 - \frac{1}{L} \sum_{l=1}^{L} \beta_{ml}$$
(13)

EAF algorithm is given in Algorithm 1.

```
Algorithm 1: Algorithm of Energetic Association Filter (EAF)
 1 Input: \hat{x}_k^m, k = 0, 1, 2, m = 1, \dots, M (offline preprocessing)
 2 for k = 2 to K do
          foreach predicted target \hat{x}_{i}^{m}, m = 1, \ldots, M do
 3
                 Compute predicted measure \hat{y}_k^m of target:
 4
                                                       \hat{y}_k^m = h(\hat{x}_k^m) + W_k
                 Compute state prediction "without association" (Eq. (4)):
 5
                                             (\hat{x}_{k+1}^m)^{\#1} = f_{k+1} \left( \hat{x}_k^m \right) + V_{k+1}
                 Compute predicted measures "without association" (Eq. (5)) and "after
 6
                 association" (Eq. (6)), respectively by:
                                             \begin{array}{lll} (\hat{y}_{k+1}^m)^{\#1} & = & h((\hat{x}_{k+1}^m)) + W_{k+1} \\ (\hat{y}_{k+1}^m)^{\#2} & = & f_{k+1}^{\#2} \left( y_k^l \right) + V_{k+1}^{\#2} \end{array}
                 for each valid measure \hat{y}_k^l (see Eq. (1)), l = 1, \ldots, L do
                       Compute distance term
 8
                       E^{1}(\hat{y}_{k}^{m}, y_{k}^{l}) = \sqrt{\left(y_{k}^{l} - \hat{y}_{k}^{m}\right)^{T} (\Sigma_{k}^{m})^{-1} \left(y_{k}^{l} - \hat{y}_{k}^{m}\right)}
                       Compute evolution term E^2(\hat{y}_k^m, y_k^l) = |S_1 - S_2|
9
                       Compute evolution of proximity term E^3(\hat{y}_k^m, y_k^l) = \frac{1}{S}
10
                       Compute global score:
11
                      E(\hat{y}_k^m, y_k^l) \ = \ \frac{1}{3} \sum_{i=1}^3 \alpha_i E^i(\hat{y}_k^m, y_k^l) \quad \text{with} \quad \alpha_i = \frac{1}{\sum_{m=1}^M E^i(\hat{y}_k^m, y_k^l)}
                       Compute posterior association probability of data association
12
                       between measures y_k^l and target x^m: \beta_{ml} = 1 - E(\hat{y}_k^m, y_k^l)
13
                 \hat{x}_{k-2}^m = \hat{x}_{k-1}^m, \ \hat{x}_{k-1}^m = \hat{x}_k^m
                 \hat{y}_{k-2}^m = \hat{y}_{k-1}^m, \ \hat{y}_{k-1}^m = \hat{y}_k^m
14
                 \beta_{ml^{\bullet}} = \max_{l} \beta_{ml}
15
                 y_{l}^{l^{\bullet}}
                      is associated with predicted target \hat{x}_{L}^{m}
16
```

We have described a novel approach for data association based on a geometric criteria to define the global score $E(\hat{y}_k^m, y_k^l)$ between an available measure y_k^l and the predicted measure \hat{y}_k^m of target state x_k^m . The three components of *E* are extracted from geometrical representations (areas and distances) constructed with measures, previous states and prediction positions. Once this score is computed, we deduce the posterior association probability for all possible targets to measure associations. Finally, EAF associates measure *l* to target *m* if the posterior association probability β_{ml} is maximal. The purpose of choosing a geometrical definition for these terms refers to:

- Show the geometrical continuity of the system between predictions and previous states using two different dynamic models;
- Measure the similarity between predictions, at a particular time for the same target, using two different dynamic models, that logically must be quite similar because they represent the same system.

4. Problem modeling for applications

For position prediction, we usually use a dynamic model describing the state process over time. For regular motions, the parameters of this model are often learned (North et al., 2000). For complex motions or online applications, the choice of parameters is difficult, especially in a missing data context. For this reason, an adaptive parametrization of the dynamic model is important when a new observation is available.

We consider a rectangular and planar observation region $\mathcal{R} \subset \mathbb{R}^2$. The state vector of target state x_k^m contains coordinates in the image domain: $x_k^m = [r_k^m \ c_k^m]^T$. Dynamical models f_k and $f_k^{\#2}$ (see respectively Eq. (4) and (6)) are cubic B-splines that extrapolate new predictions, fitted using previous predictions as control points (see Fig. 1(b)), allowing for example to deal with direction changes or non constant accelerations. \hat{x}_k^m is the predicted state of target *m* at instant *k*

Measures are also positions, available regularly or not, noisy and detected according to a probability p < 1: they can then also be false alarms, uniformly distributed in \mathcal{R} . The observation model is given by:

$$\hat{y}_{k}^{l} = \begin{cases} h(\hat{x}_{k}^{m}) + W_{k} & \text{if measure } l \text{ comes from target}m\\ u_{k} & \text{otherwise} \end{cases}$$
(14)

with W_k a Gaussian noise and u_k the random process of the false alarms. Because both predicted state and measure are positions, we have $h(\hat{x}_k^m) = \hat{x}_k^m$. We fixed $\gamma = 5.99$ as the threshold for the Mahalanobis distance (see Eq. 1) (Cox, 1993). We call δ_k the time between two successive observations.

5. Experimental results

Algorithms have been implemented in Matlab on a PC with a 2.2 GHz Intel processor. We first validate our approach on a synthetic example in Section 5.1. Tests presented in Sections 5.2 and 5.3 concern the comparison of our filter (EAF, see Section 3) with PDAF and JPDAF (see Section 1) on real video sequences. For these tests, performance tools must be defined. The complexity of the data association problem depends on two parameters: (i) the false alarm probability p_f , and (ii) the detection probability p. Increasing the probability of false alarm p_f , increasing the number of targets to track, or decreasing the probability of detection makes the association problem more complex, especially if targets move close from each other and if their trajectories cross. Performances are evaluated using the following terms:

1. The correct association probability (CAP):

$$CAP = \frac{\#correct associations}{\#possible associations}$$

(



Fig. 3. Tested sequences. (a) Synthetic test: crossed lines represent the trajectories of targets $T_1 = x_k^1$ and $T_2 = x_k^2$. The full square is the measure y_k . Dotted squares and blue stars are the predictions (see Eq. (14)) of T_2 and T_1 at instants k and k + 1 without taking into account the measure. Dotted and full circles are the prediction of T_1 and T_2 if we consider that the measure is associated with both targets. "Tennis" sequence. (b) Some frames; (c) Whole set of measures coming from the targets (crosses) or false alarms (triangles) during a test. Real trajectories of the ball and the racquet are respectively in blue and green lines. "Ant" sequence. (d) Frames at 10, 25, 35, 45; (e) Real trajectories of the six ants F_i . (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

2. The ratio of incorrect association probability to correct association probability (IAP/CAP):

$$IAP/CAP = \frac{\#incorrect associations}{\#correct associations}$$
.

An association filter should be more accurate if IAP/CAP \rightarrow 0. 3. The run time.

For each test, we propose to make three experiments:

- *Experiment I.* Variation of the false alarm probability *p*_f from 0.1 to 0.9.
- *Experiment II.* Variation of the detection probability *p* from 0.1 (target missed) to 0.9 (detected target). In these experiments, all targets are supposed to have the same detection probability.

 Experiment III. Variation of δ_k between two observations. Increasing δ_k makes the data association more difficult, especially in case of erratic motions.

5.1. Synthetic test

We first consider the 1D synthetic example of Fig. 3(a), that explores the case of oscillatory motion with a constant phase. We have two targets $T_1 = x_k^1$ and $T_2 = x_k^2$ whose real dynamic models are known and defined by $x_k^1 = \sin(k)$ and $x_k^2 = \sin(2k) + \frac{1}{2}$. We consider a specific instant k, when the measure y_k (full square in Fig. 3(a)), is equidistant from both predictions \hat{y}_k^1 and \hat{y}_k^2 of respectively targets T_1 and T_2 and falls into their validation regions. In such case, both targets are candidates to be associated with this measure. We compute the three terms for each target (see Table 1)

Table 1

Terms computed for the two targets T_1 and T_2 . The measure y_k is associated with T_1 because $E(\hat{y}_k^1, y_k)$ is minimal.

т	$\alpha_1 E^1\left(\hat{y}_k^m,y_k\right)$	$\alpha_2 E^2\left(\hat{y}_k^m,y_k\right)$	$\alpha_3 E^3\left(\hat{y}_k^m, y_k\right)$	$E(\hat{y}_k^m, y_k)$
$\hat{y}_k^1 \\ \hat{y}_k^2$	0.5	0.0001	0.4821	0.3274
	0.5	0.9999	0.5179	0.6724

and obtain $E(\hat{y}_k^1, y_k) < E(\hat{y}_k^2, y_k)$: the observation is correctly associated with T_1 .

5.2. Aperiodic oscillatory motion: "Tennis" sequence

In this test, we have M = 2 targets (the ball and the racquet), whose initial positions are known. Their real trajectories are given



Fig. 4. "Tennis" sequence: comparative tests between EAF, PDAF et JPDAF. Left column, **Experiment I**; Right column, **Experiment II**; From top to bottom: CAP depending on p_f and p, IAP/CAP ratio and run time in seconds, depending on p_f and p.

Table 2

Comparative results of online performances for the three tested filters, depending on δ_k (interval between two observation acquisitions), with p = 0.8 and $p_f = 0.3$. Best values are written in bold font.

	$\delta_k = 0.1$			$\delta_k = 0.2$			$\delta_k = 0.5$		
	EAF	JPDAF	PDAF	EAF	JPDAF	PDAF	EAF	JPDAF	PDAF
CAP	0.92	0.8	0.7	0.91	0.75	0.63	0.91	0.73	0.59
Times (s.)	0.09 26	0.25 25.2	0.429 9	0.1 17	0.33 13	0.59 7	0.1 14	0.37 9	0.69 6

on Fig. 3(c). Their motion is complex, non linear and undergoes vertical and horizontal oscillations with several periods coupled with translation in two directions: it is difficult to learn the motion from a training set. Moreover, centers of the targets are very close in the region [150,160] × [50,70]. For the experiments, $\mathcal{R} = [132, 175] \times [20, 180]$ is the observation region. The state noise is Gaussian, and its covariance matrix is $\begin{pmatrix} 0.5^2 & 0 \\ 0 & 0.1^2 \end{pmatrix}$. Measures are available at regular times, and $\delta_k = 0.1$. The covariance matrix of the measure noise is $\begin{pmatrix} 0.5^2 & 0 \\ 0 & 0.5^2 \end{pmatrix}$. We suppose each observation is made of four measures, from which at least two are false alarms. Fig. 3(c) shows the measures coming from the targets (crosses) and the false alarms (triangles) during all the test.

- ★ Experiment I. For this test, p = 0.9 for the ball, and p = 0.7 for the racquet: targets are not systematically detected. Left column of Fig. 4 shows performances of the three tested filters.
 - 1. For $p_f = 0.1$. CAP > 0.8 for all approaches. JPDAF value is greater than PDAF value, because this filter uses all measures to update the state of targets. EAF value is the greatest one, almost equal to 1. This is due to the use of an adaptive dynamic model, that permits to deal with the aperiodic oscillatory movements of the targets to give a better data association probability.
 - 2. For $p_f \rightarrow 0.9$. If the ambiguity about measure associations increases, performances of PDAF and JPDAF decrease faster than EAF's one, that keeps good results even for high values of p_f (CAP \in [0.75, 1] for EAF, and <0.4 for the two other filters).
 - 3. About ratio IAP/CAP. For small p_f values, IAP/CAP < 0.3. On the other hand, for greater values, IAP/CAP curves of PDAF and IPDAF increase very quickly: for example IAP/CAP > 1

for $p_f = 0.75$. However, EAF better detects false alarms: IAP/CAP increases slowly, and stays above 0.4 even for high values of p_f .

- 4. About run time. Due to its simplicity, PDAF is the less time consuming. EAF requires more computation time because it updates the parameters of the dynamic model for each available measure. This time is however acceptable, varying between 10 and 22 s for this experiment.
- ★ Experiment II. We suppose the probability for a target to be detected is the same for both targets ($p_f = 0.3$). The other parameters keep the same values than for Experiment I. Right column of Fig. 4 shows the performances of the three filters as a function of p.
 - 1. For p = 0.1. CAP < 0.2 for PDAF and JPDAF, because these filters associate false alarms to targets. For EAF, we have CAP \approx 0.8, showing our filter better deals with false alarms.
 - 2. For $p \rightarrow 0.9$. The ambiguity decreases with the number of false alarms. When the maximal value is reached (p = 0.9), CAP value for JPDAF is greater than the one of PDAF. But EAF value it the greatest one, with CAP ≈ 0.92 : the performance of our filter is better even when targets are well detected.
 - 3. About ratio IAP/CAP. For p = 0.9, *IAP/CAP* > 0.5: the minimal value for this ratio is obtained with our filter. When p decreases, *IAP/CAP* increases quickly for PDAF and JPDAF, contrary to EAF. For a small value of p, *IAP/CAP* is always lower than 0.5. We obtain a better performance with our filter because it better takes into account the movement of targets, providing a better association probability.
 - 4. About run time. PDAF is the fastest filter, because it requires less computations to associate measures to targets. EAF and JPDAF give similar computation times.
- ★ Experiment III. We test three values for δ_k (0.1, 0.2 and 0.5). For both targets, p = 0.8 and $p_f = 0.3$. Table 2 shows comparative

Table 3

"Ant" sequence. Normalized evolution energies $\alpha_1 E^1(\hat{y}_k^w, y_k^l), \alpha_2 E^2(\hat{y}_k^w, y_k^l), \alpha_3 E^3(\hat{y}_k^w, y_k^l)$ and global energy $E(\hat{y}_k^w, y_k^l)$ computation. Best values are written in bold font.

		(= =) (=		(= =)			
		y_k^1	y_k^2	y_k^3	y_k^4	y_k^5	y_k^6
\hat{y}_k^1	$\alpha_1 E^1(\hat{y}^m_{\nu}, y^l_{\nu})$	0.07	0.47	0.26	0.44	0.49	0.46
	$\alpha_2 E^2(\hat{y}^m_k, y^l_k)$	0.01	0.02	0.01	0.01	0.11	0.02
	$\alpha_3 E^3 (\hat{y}^m_{\mu}, y^l_{\mu})$	0.01	0.45	0.01	0.14	0.47	0.42
	$E(\hat{y}_k^m, y_k^l)$	0.03	0.31	0.09	0.2	0.36	0.3
\hat{y}_k^2	$\alpha_1 E^1(\hat{y}_k^m, y_k^l)$	0.23	0.06	0.21	0.06	0.03	0.12
	$\alpha_2 E^2(\hat{y}_k^m, y_k^l)$	0.03	0.01	0.04	0.01	0.54	0.02
	$\alpha_3 E^3(\hat{y}_k^m, y_k^l)$	0.07	0.01	0.24	0.02	0.15	0.14
	$E(\hat{y}_k^m, y_k^l)$	0.12	0.03	0.16	0.04	0.24	0.09
\hat{y}_k^3	$\alpha_1 E^1(\hat{y}_k^m, y_k^l)$	0.15	0.25	0.04	0.18	0.24	0.2
ĸ	$\alpha_2 E^2(\hat{y}_k^m, y_k^l)$	0.14	0.01	0.01	0.03	0.01	0.01
	$\alpha_3 E^3 (\hat{y}^m_{\mu}, y^l_{\mu})$	0.22	0.08	0.02	0.06	0.05	0.25
	$E(\hat{y}_k^m, y_k^l)$	0.17	0.11	0.02	0.09	0.1	0.15
\hat{y}_k^4	$\alpha_1 E^1(\hat{y}_k^m, y_k^l)$	0.21	0.02	0.22	0.09	0.05	0.10
	$\alpha_2 E^2(\hat{y}_k^m, y_k^l)$	0.07	0.02	0.03	0.01	0.24	0.02
	$\alpha_3 E^3(\hat{y}_k^m, y_k^l)$	0.39	0.15	0.28	0.01	0.18	0.12
	$E(\hat{y}_k^m, y_k^l)$	0.22	0.06	0.18	0.03	0.16	0.08
\hat{y}_k^5	$\alpha_1 E^1(\hat{y}_k^m, y_k^l)$	0.18	0.06	0.18	0.12	0.09	0.06
	$\alpha_2 E^2(\hat{y}_k^m, y_k^l)$	0.74	0.93	0.9	0.93	0.07	0.92
	$\alpha_3 E^3(\hat{y}_k^m, y_k^l)$	0.33	0.4	0.4	0.36	0.06	0.35
	$E(\hat{y}_k^m, y_k^l)$	0.33	0.4	0.4	0.36	0.07	0.35
\hat{y}_k^6	$\alpha_1 E^1(\hat{y}_k^m, y_k^l)$	0.16	0.14	0.09	0.11	0.10	0.06
	$\alpha_2 E^2(\hat{y}_k^m, y_k^l)$	0.01	0.01	0.01	0.01	0.02	0.01
	$\alpha_3 E^3(\hat{y}_k^m, y_k^l)$	0.23	0.2	0.32	0.75	0.11	0.01
	$E(\hat{y}_k^m, y_k^l)$	0.13	0.12	0.14	0.29	0.08	0.03



Fig. 5. "Ant" sequence. Comparative tests between EAF, PDAF and JPDAF. Left column, **Experiment I**; Right column, **Experiment I**; From top to bottom: CAP depending on p_f and p, IAP/CAP ratio and run time in seconds, depending on p_f and p.

results of CAP, IAP/CAP and run times. EAF performs better: the increase of δ_k does not affect CAP, IAP/CAP and run time. The other filters quickly fail, mainly because of the aperiodic oscillatory motion of the two targets and their proximity into the region [150, 160] × [60, 80] (see Fig. 3(c)).

5.3. Interaction between targets and crossed trajectories, "Ant" sequence

This sequence shows six ants moving on a flat surface. Each ant has a non linear and erratic movement: translation while walking and deformations while moving its head, abdomen, or antennas (see trajectories in Fig. 3(e)). Ants frequently interact, sometimes four of them are close from each others and are quite similar even non-distinguishable, characterized by the same gray level distribution and shape: only their motion can be useful for the data association.

We first test the robustness of EAF. Images of Fig. 3(d) are observations at instants] k - 2, k - 1, k and k + 1, and respectively

represent frames 10, 25, 35 and 45 (ants are labeled to show their real positions from a frame to another). There is a considerable interval of time between frames k - 1 and k + 1 ($\delta_k = 0.1$).

Table 3 gives the values of components $\alpha_1 E^1(\hat{y}_k^m, y_k^l), \alpha_2 E^2(\hat{y}_k^m, y_k^l)$ and $\alpha_3 E^3(\hat{y}_k^m, y_k^l)$ computed between measures y_k^l and the predicted measure \hat{y}_k^m of six predicted target state \hat{x}_k^m : the first row represents the measure number and the first column the target number $(\hat{y}_k^l$ really coming from measure y_k^l). If we associate measures to targets only using the Mahalanobis distance minimization, we associate y_k^1 to \hat{y}_k^1, y_k^2 to \hat{y}_k^4, y_k^3 to \hat{y}_k^3, y_k^4 to \hat{y}_k^2, y_k^5 to y_k^2 and y_k^6 to y_k^5 . This leads to two problems: only two measures are well associated, and two measures are associated to the same target, that is not possible under our assumptions. Global energies are also given in Table 3: we observe each measure is well associated with its corresponding target, that show the interest of our approach.

In this second part of our tests, the sensor, at instant k, provides an observation containing five measures (from targets or false alarm, see Eq. (14)): at least one ant is not detected (missing data).

- ★ Experiment I For this experiment, we have p = 0.8, $\delta_k = 0.1$. Left column of Fig. 5, shows, from top to bottom, the CAP, IAP/CAP ratio and run time curves obtained for the three filters.
 - 1. For $p_f = 0.1$. CAP values for filters PDAF (=0.7) and JPDAF (=0.72) are smaller than the one of EAF (=0.98): PDAF et JPDAF give similar results but EAF is 26% better. Because of the crossed trajectories and the proximity of the ants (particularly in region [80,300] × [80,200], see Fig. 3(d)), the other filters do not provide good associations while our filter is not affected.
 - 2. For $p_f \rightarrow 0.9$. EAF is the less affected with the increasing of p_f , comparing to PDAF and JPDAF: its CAP is always >0.8.
 - 3. About ratio IAP/CAP. For $p_f = 0.1$, IAP/CAP ≈ 0 for EAF, and ≈ 0.4 for the other filters: EAF is then 40% better. When p_f increases IAP/CAP ratio also increases for the three filters, but always stays <0.2 with our filter.
 - 4. About run time. As previously, PDAF requires less computation time than the other filters. Computation time of EAF is smaller than the one of JPDA for a small value of $p_f (\leq 0.22)$, and becomes greater after.
- ★ Experiment II In this experiment, we suppose all ants have the same detection probability $p_f = 0.3$. The other parameters keep same values than in the previous experiment. Right column of Fig. 5, shows, from top to bottom, the CAP, IAP/CAP ratio and run time curves obtained for the three filters along the sequence.
 - 1. For p = 0.1. In this case, targets are not systematically detected. CAP value for PDAF and JPDAF < 0.4: they correctly associate only 60% of the measures, when EAF correctly associates 80% of them.
 - 2. For $p \rightarrow 0.9$. The correct association probability of the filters increases with p: the maximum for PDAF and JPDAF is 0.7, while the one of EAF is 0.95. Note that for PDAF and JPDAF, CAP does not increase for p > 0.35, worst, it decreases for PDAF.
 - 3. About ratio IAP/CAP. This ratio increases for the three filters with the decreasing of *p*. Ir is <1 for JPDAF and PDAF with p < 0.2, and ≈ 0.3 for EAF.
 - 4. About run time. PDAF remains the fastest filter. Mean values of the computation time for JPDAF and EAF are equivalent, but EAF necessitates less computation time for $0.35 \le p \le 0.75$.

6. Conclusion

This work proposes a new method for data association based on a geometric criteria. The developed approach can handle complex motions and highly non-linear systems, and deals with the lack of prior knowledge, by only using few parameters. The geometric illustration of energy components allows to measure the accuracy between two dynamic models and to define their degree of similarity. The three components are extracted from geometrical representations (areas and distances) constructed with measures, previous states and prediction positions. The purpose of choosing a geometrical definition for these terms refers measure the similarity between predictions, at a particular time for the same target, using two different dynamic models, that logically must be quite similar because they represent the same system.

As a perspective for this work, we suggest to integrate the energetic association filter within the classical particle filter to build a new framework for multiple target tracking. Moreover, since we consider erratic motions that cannot be learned from training sequences, we suggest to use an adaptive and automated way to set parameters of the dynamic model of the filter. Then, we could track targets under the restriction of the missing of prior information and especially if similar targets are evolving in the scene.

References

- Bar-Shalom, Y., Fortman, T., 1988. Tracking and Data Association. Academic Press. Bar-Shalom, Y., Li, X., 1995. Multitarget Multisensor Tracking: Principles and Techniques. YBS Publishing.
- Blake, A., Isard, M., 1998. Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion. Springer-Verlag, New York, Inc., Secaucus, NJ, USA.
- Chen, Y., Rui, Y., Huang, T.S., 2001. JPDAF based HMM for real-time contour tracking. In: IEEE Conf. on Computer Vision and Pattern Recognition. pp. 543–550.
- Cox, I., 1993. A review of statistical data association techniques for motion correspondence. Internat. J. Comput. Vision 41 (1), 53–65.
- El Abed, A., Dubuisson, S., Béréziat, D., 2006. Comparison of statistical and shapebased approaches for non-rigid motion tracking with missing data using a particle filter. In: Advanced Concepts for Intelligent Vision Systems. Antwerp, Belgium, pp. 185–196.
- Fortmann, T., Bar-Shalom, Y., Scheffe, M., 1983. Sonar tracking of multiple targets using joint probabilistic data association. In: IEEE Journal of Oceanic Engineering, vol. 8. pp. 173–194.
- Gauvrit, H., Cadre, J.-P.L., Jauffret, C., 1997. A formulation of multitarget tracking as an incomplete data problem. IEEE T. Aerosp. Electron. Syst. 33, 1242–1257.
- Gelgon, M., Bouthemy, P., Le Cadre, J.-P., 2005. Recovery of the trajectories of multiple moving objects in an image sequence with a PMHT approach. Image Vision Comput. J. 23 (1), 19–31.
- Jaward, M., Mihaylova, L., Canagarajah, N., Bull, D., 2006. A data association algorithm for multiple object tracking in video sequences. In: IEE Seminar on Target Tracking: Algorithms and Applications. pp. 135–142.
- Konstantinova, P., Udvarev, A., Semerdjiev, T., 2003. A study of a target tracking algorithm using global nearest neighbor approach. In: CompSysTech '03: Proc. of the 4th Internat. Confer. on Computer Systems and Technologies, ACM, New York, NY, USA, pp. 290–295.
- North, B., Blake, A., Isard, M., Rittscher, J., 2000. Learning and classification of complex dynamics. In: IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22. pp. 333–339.
- Rasmussen, C., Hager, G.D., 2000. Probabilistic data association methods for tracking multiple and compound visual objects. IEEE Trans. Pattern Anal. Machine Intell. 23, 560–576.
- Rasmussen, C., Hager, G., 2001. Probabilistic data association methods for tracking complex visual objects. In: IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 23. pp. 560–576.
- Reid, D., 1979. An algorithm for tracking multiple targets. IEEE T. Automat. Contr. 24 (6), 1016–1034.
- Rong, L., Bar-Shalom, Y., 1996. Tracking in clutter with nearest neighbor filter: Analysis and performance. In: IEEE Trans. on Aerospace and Electronic Systems, vol. 3. pp. 995–1010.
- Singer, R., Sea, R., Housewright, K., 1974. Derivation and evaluation of improved tracking filters for use in dense multitarget environments. IEEE Trans. Inform. Theory 20 (4), 423–432.
- Song, T.L., Lee, D.G., Ryu, J., 2005. A probabilistic nearest neighbor filter algorithm for tracking in a clutter environment. Signal Process. 85 (10), 2044–2053.
- Therrien, C., 1989. Decision Estimation and Classification: An Introduction to Pattern Recognition and Related Topics. Wiley, New York.
- Vermaak, J., Godsill, S., Pérez, P., 2005. Monte carlo filtering for multi-target tracking and data association. In: IEEE Trans. on Aerospace and Electronic Systems, vol. 33. pp. 309–332.
- Yilmaz, A., Javed, O., Shah, M., 2006. Object tracking: A survey. ACM Comput. Surveys 38 (4), 1–45.