Soft Computing, (2011), 15 (9), 1845-1864.
DOI: http://www.dx.doi.org/10.1007/s00500-011-0698-z

Fuzzy region assignment for visual tracking

Jesus Garcia, Miguel A. Patricio, Antonio Berlanga, Jose M. Molina

J. Garcia (✉) · M. A. Patricio · A. Berlanga · J. M. Molina GIAA-Departamento de Informatica, Universidad Carlos III de Madrid,
Avda, Universidad Carlos III, 22, 28270 Colmenarejo, Spain
e-mail: jgherrer@inf.uc3m.es

Abstract In this work we propose a new approach based on fuzzy concepts and heuristic reasoning to deal with the visual data association problem in real time, considering the particular conditions of the visual data segmented from images, and the integration of higher-level information in the tracking process such as trajectory smoothness, consistency of information, and protection against predictable interactions such as overlap/occlusion, etc. The objects' features are estimated from the segmented images using a Bayesian formulation, and the regions assigned to update the tracks are computed through a fuzzy system to integrate all the information. The algorithm is scalable, requiring linear computing resources with respect to the complexity of scenarios, and shows competitive performance with respect to other classical methods in which the number of evaluated alternatives grows exponentially with the number of objects.

Keywords: Machine vision; Video data association; Fuzzy system design

1 Introduction

The research on video processing algorithms to track and analyze the objects moving in a scene is one of the most demanding areas of computer vision, and has been receiving intensive attention in the recent years. These algorithms must solve the detection, recognition and tracking ofinteresting objects in the video sequence with satisfactory performance, usually having available multiple cameras and computation resources networked to cover an extended area (Moeslund et al. 2006). Among the applications, the visual surveillance systems are especially relevant nowadays, given the very demanding requirements and expectations for monitoring safety conditions in protected areas (Cucchiara et al. 2004; Javed and Shah 2002; Medioni et al. 2001; Ferryman et al. 2000; Leuven et al. 2001; Brodsky et al. 2001; Greenhill et al. 2002). Other relevant applications are advanced visual interfaces for context-aware applications (Koller et al. 1997; Krumm et al. 2000) and video mining systems to retrieve and understand situations for statistical analysis of, for example, sports, physical performance of players, semantic analysis, etc. (Xu et al. 2004; Liu et al. 2009; Joo et al. 2007). A fundamental requirement for these systems is detection, labeling and tracking of objects. Another requirement is the capability to track and maintain identity of all detected objects continuously over time.

Motion correspondence in video analysis basically requires from robust data association methods, an area which has started to receive attention also from the computer vision community in recent years. The data association problem (also named data correlation) consists in the appropriate correspondence among observations extracted from each frame to the objects extracted in the previous ones, a necessary step before to the estimation of the individual targets states. Objects should be tracked without interruption even in the case where the low-level detection algorithms fail to segment them in the images. This correspondence among sequential observations is hard for different reasons. The predictions are done accordingly to previous estimations and must be corresponded to current measurements. Ambiguity rises when predictions are not supported by measurements, there are unexpected measurements or several observations may match with some predictions. This problem occurs in a variety of diverse domains in which observations arrive in time, including computer vision, surveillance, air traffic control, defense, robotics and target tracking, whose community has coined the term "data association" in this specific sense.

So, the problem of data association addressed in this work has a specific meaning in this context of processing a stream of sensor data. It differs from the general association problem in data mining, which refers to the search of semantic linkages between attributes of data instances, such as the relations discovered in basket analysis. The a priori algorithm (Agrawal 1996) is one of the original association methods in data mining, from which many efforts have continued to develop association methods with capabilities to generalize with uncertainty conditions and integrate high-level knowledge (Novak et al. 2008).

A number of statistical and alternatives techniques have been developed for the sensor data association problem, where a typical step toward assignment of measurements is the computation of likelihood of measurements generated by different hypotheses of correspondences with predictions. Different proposals range from simple and suboptimal, such as Nearest Neighbor (NN), to other more complex and close to optimal approaches, whose cost is usually excessive when the number of targets and measurements increase, such as Multiple Hypotheses Tracking (MHT), and Joint Probabilistic Data Association Filter (JPDAF). Fuzzy systems have been traditionally applied to data association (Fuzzy Data Association, FDA) with sensor sources such as radar positioning, infrared and Doppler measurements, taking advantage of the flexibility of fuzzy logic to model uncertainty coming from heterogeneous sources and natural ability to handle different types of information (Chen 2000; Han et al. 2003; Ermin et al. 2000; Gad et al. 2002; Aziz et al. 1999, 2007). A more detailed analysis of approaches to data association based on this paradigm is presented in the next section. In the same way as this work, a research line in data mining extends the semantics of associations to deal with uncertainty and imprecision, including knowledge representation with generalized concepts and linguistic expressions (Novak et al. 2008).

The application of a soft-computing paradigm to video tracking and motion correspondence is much scarcer, outstanding the previous work by the authors (Garcia 2002; Garcia et al. 2005), and application to image segmentation in order to approximate conditional probability densities at pixel level (Cho et al. 2007). In this paper we present a robust method for visual data association based on the integration of visual information at several levels of granularity: low-level image segmentation operations, medium-level smoothness criteria on target features and high-level constraints on tracking continuity. It is based on a rule-based system with fuzzy sets to represent the concepts at different levels, employing heuristic and geometrical reasoning in the tracking process. The approach presented extends previous system (Garcia et al. 2005) to provide a complete and modular solution of the video association problem, formulated as the decision of the foreground image regions to update each active track independently of the subsequent specific tracker applied. With a Bayesian foundation of a data association algorithm to maximize likelihoods, the proposed method integrates concepts at several levels to take the decisions of assigning the image regions corresponding to each object. The fundamental goal is to find an efficient solution to the association problem in the presence of splits and merges, one with robustness to find good solutions and avoid system instabilities. It avoids the combinatorial analysis of region subsets; every blob is compared to every track only once to compose the assigned synthetic regions. This allows a strict linear complexity, differing from most conventional approaches requiring an exponential number of operations. The main contributions are highlighted below:

- The concepts proposed for the fuzzy model are appropriate for reasoning on video tracking, extending previous approaches applied to other sensors based only on point tracking, and on innovation residuals analogous to the Mahalanobis distance (Cox 1993). The concepts are specifically considered to solve the split/merge problems (the current challenge of video tracker in hoard conditions) in the most efficient way. The concepts proposed use structural information with a geometrical analysis of shape and size.
- The usual one-to-one constraints of other applications of JPDA, NN or MHT are relaxed to take into account the merging/splitting effects which appear in realistic situations. The problem is dealt with at the unconstrained level, as the correspondence of multiple blobs to multiple tracks considers multiple fragmented or merged blobs to update each track. Groups of blobs (pseudo blobs) are created by aggregation of confident blobs, and assigned to update tracks.
- The result of fuzzy evaluation (confidence between each blob-track pair) is used to make a composition of foreground image regions and generate the final synthetic segmented image to update the track. This approach weights the measurements but, instead of deriving a centroid, as in the usual, earlier PDAF approaches, the geometrical analysis allows reasoning with shape parameters considered as intermediate concepts. The confidence level is used to update the

track attributes, since they depend on the final group to be associated to the track.

The multitarget situation is explicitly considered through a concept which assesses the degree of conflict. A conservative approach is used to block degradation due to merging of data coming from different targets through geometrical analysis. Differently from other applications of fuzzy logic, the exponential effect on the number of rules is avoided by limiting the conflict analysis to the worst case of the most overlapped track. Furthermore, the situation of track coalescence with merging and occlusion is handled by limiting the deformation of tracks through limitations in the sets of blobs. This is a limitation equivalent to that of hypotheses implying simultaneous groups of conflicting blobs shared by different tracks. In those cases, the regions to update the tracks are conditioned by prior track shapes in order to avoid severe deformations and loss of previous estimation (Mori et al. 2005).

The rest of paper is organized in six sections. Section 2 reviews the sensor data association problem and the families of approaches to solve it, highlighting the most relevant soft-computing approaches applied before, such as FDA. Section 3 presents the video data association problem using Bayesian formalism, and the terminology used. Section 4 details the proposed algorithm for video data association based on fuzzy region assignment (FRA), including geometrical heuristics to represent the tracking situations, the creation of stable regions to update the tracks, and an overview of the algorithm complexity with respect to classical approaches and to FDA. Section 5 contains the results of the performance of the proposed algorithm in relation to well-known visual tracking algorithms, and the conclusions are summarized in Sect. 6.

2 Sensor data association and soft-computing approaches

The inference of the real state at a certain environment, based on the information coming as sensor observations, is usually addressed as an estimation problem. It consists in estimating the number of objects in a scene, together with their dynamic state (location, speed, attitude, size, etc.), based on the available observations.

The Kalman filter is the most popular estimation technique to estimate the track state vector at frame k, combining the information in the current observation with the prediction from previous frame at k-1 with equations for obtaining the optimal solution under linear-Gaussian assumptions. However, the Kalman filter provides the solution for the particular problem of single state vector updated with a single measurement at each frame; that is, it assumes that there is a single object in the environment and it is the source of all measurements. The problem is that the correspondences among observations and objects are unknown; they must be estimated from the observed data. For that reason, tracking multiple objects is a much more difficult problem, it deals with an unknown number of active objects as sources of measurements, and the statistical model requires both continuous variables to describe each target state and discrete variables to describe the correspondences between objects and observations. The multitarget tracking problem is divided into two problems: data association and state estimation. Data association decides the correspondences to pair objects and observations. Then, once the association is decided, one applies Kalman filter to estimate each target's state conditioned to this decision. Figure 1 illustrates an example in which there are four objects whose trajectories get so close that the noisy measurements are mixed.

The families of algorithms for data association are usually classified in two groups, algorithmic, and non-algorithmic (Singh et al. 1997). Algorithm (or classic) methods are further subdivided in two approaches, combinatorial (or non-Bayesian), based on nearest neighbor with single-hypothesis techniques, and Bayesian techniques, such as multiple-hypothesis tracking and joint probabilistic data association. Non-algorithmic (approximate) methods include knowledge-based systems, fuzzy logic and neural networks.

As mentioned, a powerful and general method is the joint probabilistic data association filter, also including the phases of data association and state estimation. Data association assigns measurement to targets to prepare the

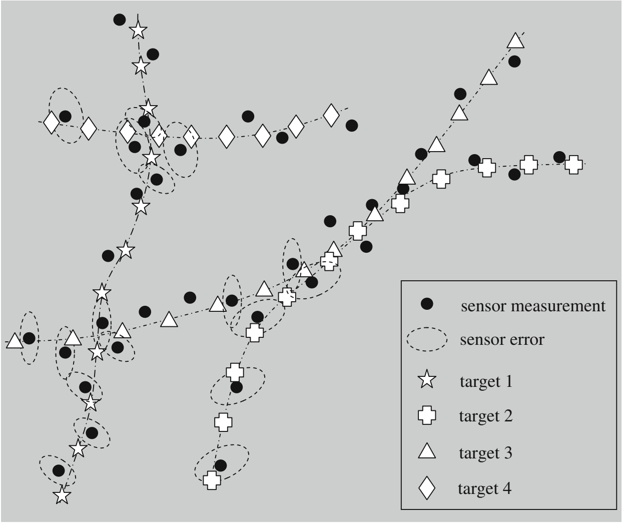


Fig. 1 Sensor data association with four trajectories

state estimation phase. The basic characteristic of JPDAF is the estimation of association probabilities, from the joint likelihood functions corresponding to hypotheses associating observations to objects. Then, the update of target states is computed with the weighs corresponding to the JPDAF probabilities.

Soft-computing approaches to data association are inspired in the human ability to reason by simultaneously integrating information at different levels of abstraction. Thus, non-algorithmic approaches such as artificial neural networks, fuzzy systems and genetic algorithms can be applied to data association problems, isolated or in conjunction with classical formulations. Methods based on fuzzy systems and artificial neural networks have been used to compute the association probabilities in JPDAF, to take the best decisions in the association process in different conditions, accordingly to the characteristics of objects and available sensors (Turkmen et al. 2004; Sengupta et al. 1989; Chen et al. 2001). Genetic Algorithms, with a recognized capability to address hard search problems, have been previously applied in the data association problem in radar data processing by Angus et al. (1993) and by Hillis (1997) to deal with the mono and multiscan data association problems, respectively. The authors have also proposed the use of evolutionary computation in visual data association (Patricio 2008).

Neural networks have been applied to estimate the association probabilities in JPDA filters (Sengupta et al. 1989; Turkmen et al. 2004; Zhu et al. 1994; Shams 1996), representing the measured residuals between observation and tracks as inputs for the network. These approaches have proved capable of handling complex scenes with radar data, although the implementations require a large number of neurons and the preparation of large training data sets to have a reliable system. The usual attributes are based on the way humans perform visual grouping, using principles such as proximity, common paths or directions, similarity of shape, color, size, closure of boundaries and continuation of contours and edges that extend smoothly. For instance, Bogner et al. (1998) evaluate the association of radar plots with Over Horizon Radar, a typical problem in which propagation through different ionospheric layers produces up to four replicas of plots forming repeated tracks for the same target.

Fuzzy systems are one of the most outstanding non-algorithmic approaches used in the data association problem, a general strategy called FDA. They provide approximate solutions which are simple, robust and efficient, joining high-level reasoning with numerical computation. They have been applied mostly with radar positioning and doppler measurements and fusion of radar with other sensors such as infrared cameras (Singh et al. 1997, Chen and Huang 2000, Han et al. 2003, Gad et al. 2002; Ermin et al. 2000). In practically all cases, targets and measurements are presented as "point-type" detections, representing the statistical behavior with multidimensional Gaussian distribution. The fuzzy membership function is used to characterize the degree of belonging to the linguistic concepts with respect to the degree of association between each observation and track. Basically the input variables in all cases are the residuals between estimated position and velocity of centroids with respect to measurements extracted from processed data. The level of detail is given by the number of fuzzy linguistic elements, which are as many as the desired granularity. The usual application of fuzzy logic to data association has four basic elements: (1) fuzzifzy numeric inputs into fuzzy variables, (2) express a knowledge base containing a set of "IF THEN" rules, (3) fuzzy inference which emulates expert decision processes to generate output, and (4) defuzzify fuzzy output variables into numeric variables. For instance, Han et al. (2003) propose a number of fuzzy rules for data fusion and convert the data into fuzzy sets with the values {NB, NS, ZO, PS, PB} (negative big, negative small, zero, positive small, and positive big). The output variables in FDA are usually the degree of correlation between observations and tracks, so that the maximum values are searched for as solutions to the association problem. As an alternative to rule-based fuzzy systems, Aziz et al. (1999, 2007) propose the application of fuzzy clustering means (FCM) algorithm. Their iterative algorithm applies FCM over an active set of measurements and tracks, identifies the pairs with highest membership and removes them to reduce the size of the problem for the next iteration until all measurements are assigned to a track. However, application of FCM to data association supposes, from our point of view, renouncing to the ability to inject expert knowledge to solve the problem. This is especially important when the attributes have different magnitudes and heterogeneous semantic meaning, and in the domain considered here, which is that of visual tracking of image attributes where the relationships between attributes and the target variable, the association matrix, cannot be clearly identified.

As far as we know there are no extensions of fuzzy data association algorithms to the video tracking problem, apart from the previous work by authors (Garcia et al. 2005), and application to image segmentation in order to approximate conditional probability densities at pixel level (Cho et al. 2007). The extension for data association in the video domain must cover the modeling of image attributes so that the information can be injected through the likelihood definition, $p(Z|X)$, going further than the approaches based on "point" residuals, the formulation with multiblob-to-multitrack assignments (covering merging and splitting situations), and an efficient solution to multitrack situations avoids increasing the input space (among previous works only in (Gad et al. 2002) is the practical problem of developing rules with a high number of targets considered).

3 Visual-data-association problem statement

In the case of video data association, the objective is mainly the same as other multitarget tracking systems: objects should be robustly tracked in time, even though the image processing algorithms fail in some intervals to segment them as single foreground regions (blobs). Problems with object segmentation often occur (Genovesio 2004; Kumar et al. 2006): when another region occludes the object (a fixed object in the scene or other moving object), when the object image is split into fragments during image segmentation, or when the images from different objects get merged because of their closed or overlapped projection on the camera plane. Besides, extraneous elements in the scene such as waving trees, smoke, clouds, etc., may originate false detected regions interacting with the real objects of interest but they should not degrade their continuity. A frequent problem with merged regions is to correctly recover the original trajectories when the objects ''reappear'' after the time interval of occlusion (Tao 2002; Haritaoglu et al. 1998, 2000).

Classical data association techniques, previously explored in other fields of sensors and target tracking, have been recently adopted and extended by computer vision researchers. The JPDA filter has been applied to 3-D vision reconstruction (Chang 1991; Kan et al. 1996). Cox and Hingorani (1996) proposed the first adaptation of Reid's MHT (Reid 1979) to visual data association problems, although objects are simplified to points without considering the split/merge problem. Medioni et al. (2001) proposed an approach based on graph theory for tracking multiple targets which was similar to Reid's MHT. Their algorithm considered splits only, and they used gray level correlation between objects and segmented blobs to detect and handle splits. Other methods based on graphs for data association have been proposed by Chen et al. (2001), but using a one-to-one assumption. In recent approaches (Khan et al. 2005; Liu et al. 2009, Cai et al. 2006), a Markov Chain Monte Carlo (MCMC) strategy is applied to explore the data association space in order to estimate the MAP joint distribution of multiple targets by means of a MCMC method. Other recent approaches (Fleuret 2008) are based on discretized occupancy maps in the real world onto which the objects are projected. The association and estimation are solved through the computation probabilities of occupancies for the sequence of discretized locations of objects in the discrete space, making use of Hidden Markov Models.

The visual tracking problem consists in the estimation of the number of objects in a scene, together with their instantaneous location, cinematic state and additional attributes (size, shape, color, identification, etc.). In this sense, environment E is defined for each time instant $t[k]$ as a set of $N[k]$ objects, $E[k] = \{O_1[k],\ldots, O_{N[k]}[k]\}$, where each object is defined by a set of characteristics in this instant. The description of the objects is expressed in a vector state space, $\bar{x}_i \in \Re^d$ For instance a common simplified representation of objects in a 2D camera plane contains the position of a centroid object, together with bounds (width and length) and their velocity and scale derivatives:

$$\bar{x}_i = [\,x_i \quad y_i \quad wx_i \quad wy_i \quad \dot{x}_i \quad \dot{y}_i \quad \dot{w}x_i \quad \dot{w}y_i]^t \tag{1}$$

The image preprocessing step acquires characteristics of the objects disturbed by the measurement process. In this work we will consider the preprocessing phase to be the background subtraction and thresholding to detect moving objects in monocular images (Stauffer and Grimson 1999; Fleuret 2008). After these processes we have a binary image where a detected object is observed through a set of compact regions (blobs), formed by adjacent binary detected pixels in this instant:

$$Z^i[k] = \{b_1^i[k], \ldots, b_{M^i}^i[k]\} \tag{2}$$

where M^i is the number of blobs that are due to the i-*th* object. The problem is that both $N[k]$ and superscript i are hidden so they must be estimated from the observed data. The only observable amount is the global set of blobs appearing in the whole foreground image: $Z[k] = \{b_1[k],\ldots b_{M[k]}[k]\}$. Thus, the basic problem in video data association is the re-connection of blobs and assignments to update the tracks, searching the subsets of blobs corresponding to each track $\bar{x}_i[k], Z^i[k]$.

A Bayesian framework to determine the best estimation, $X[k]$, inferred from available measurements, $Z[k]$, is the one targeted at obtaining the maximum a posteriori probability of the estimated state, conditioned to the whole set of observations:

$$\hat{X}[k] = \underset{X[k]}{\arg\max}\, P(X[k]|Z[k], Z[k-1], \ldots, Z[0]) \tag{3}$$

where $\hat{X}[k]$ denotes both the number of targets and their state in the scene at time instant $t[k]$, $\hat{X}[k] = \hat{x}_{1\ldots N_k}[k] = \{\hat{x}_1[k], \ldots, \hat{x}_{N_k}[k]\}$, where $\hat{x}^i[k] \in \Re^d$, in our case $d = 8$ as indicated above.

The classical inference formulation applies Bayes' theorem to rearrange the problem in a recursive formulation:

$$\begin{aligned} &P(X[k]|Z[k], Z[k-1], \ldots, Z[0]) = \\ &\quad \frac{1}{c} P(Z[k]|X[k]) \int [P(X[k]|X[k-1]) \\ &\quad P(X[k-1]|Z[k-1], \ldots, Z[0])] \mathrm{d}X[k-1] \end{aligned} \tag{4}$$

where the integral in the joint problem would extend over the whole space of the predicted state, $P(X[k]|X[k-1])$ and c is the normalization constant to guarantee that the result is a probability density. In this formulation, and dropping the time index for simplicity, $P(Z|X)$ is the likelihood function, i.e. the probability of observing a particular image Z given a certain current state X. As mentioned above, in our case we particularize the observation process to the analysis of the binarized image resulting from the background subtraction and thresholding, $Z[k] = \{b_1[k],\ldots, b_{Mk}[k]\}$.

The association problem can be considered as part of the maximization of the a posteriori likelihood of observations, considering the sequential series of data assignments:

$$P(Z[k]|X[k]) = P(Z[k]|\mathbf{A}[k], Z[\mathrm{k}-1], \mathbf{A}[k-1], Z[\mathrm{k}-2], \ldots, \mathbf{A}[0], Z[0]) \tag{5}$$

where the assignment matrix $\mathbf{A}[k] = \{a_{ij}[k]\}$ is defined as $a_{ij}[k] = 1$ if blob $b_i[k]$ is assigned to track $\hat{x}_j[k]$; and $a_{ij}[k] = 0$ otherwise. In the k-th frame there are $M[k]$ blobs extracted to be assigned, $b[k] = \{\mathrm{b}_1[k],\ldots, b_{Mk}[k]\}$, and the objects tracked up to them (the last assignment of blobs was at frame $k-1$) are: $X[k-1] = \{O_1[k-1],\ldots, O_{Nk-1}[k-1]\}$.

Classical combinatorial methods are characterized by hard associations of measurements to tracks, based on a certain cost criterion, and then tracking propagates the decisions taken at every instant as if they were right:

$$\underset{A[k]}{\mathit{maximize}}\, f(A[k]) = P(Z[k]|A[k], Z[k-1], A[k-1], Z[k-2], \ldots, A[0], b[0]) \tag{6}$$

So, the optimal estimation under this formulation is equivalent to finding the sequence of association matrices to correspond observations and tracks to apply estimation algorithms. However, this joint optimization of the whole sequence of association matrices is not possible, its complexity increases at an exponential rate with time. A practical approach (single-hypothesis optimization) is the sequential optimization of association decisions, where decision at frame $k-1$ is propagated for time k, and the search space reduces to the size of matrix $A[k]$ for each processed frame. The association decision at time k, $A[k]$, is computed to maximize the likelihood of current detections, conditioned on the given chain of previous assignments, $A[k-1]$, $A[k-2]$,…, $A[0]$. This likelihood of current observations conditioned on all previous assignments, $\mathbf{A}[m]$, $m = 0,\ldots, k-1$, can be recursively defined with previous tracking states, $\hat{x}_j[k-1]$, $j = 1,\ldots,N[k-1]$. Thus, the previous expression can be approximated by:

$$\underset{\mathbf{A}[k]}{\mathrm{maximize}}\, f(\mathbf{A}[k]) \approx P(Z[\mathrm{k}]|\mathbf{A}[k], \hat{x}_{1,\ldots,N_{k-1}}[k-1]) \tag{7}$$

with $\hat{x}_{1,\ldots,N_{[k-1]}}[k-1] = \{\hat{x}_1[k-1], \ldots, \hat{x}_{N[k-1]}[k-1]\}$ and $\hat{x}^i[k-1] \in \Re^d$. These vectors, containing state information on objects, are recursively updated with the sequence of assigned observations, using motion and observation models, by means of a Kalman filter, and predicted to the k-*th* frame of current observations.

In order to reduce the search space for the assignment matrix $A[k]$, a gating criterion is usually defined, discarding in this way the farthest pairs of tracks and measurements: $\Omega[k] = \begin{bmatrix} \omega_{11}[k] & \cdots & \omega_{1n}[k] \\ \vdots & \ddots & \vdots \\ \omega_{m1}[k] & \cdots & \omega_{mn}[k] \end{bmatrix}$, being $\omega_{ij}[k] = 1$ if distance (att($b_i[k]$), $\hat{x}_j[k]$) <=Th.

Then, after the gating phase we can identify a set of measurements compatible with each track, those which could potentially be assigned to track $\hat{x}_j$ under any hypothesis: $W_j = \bigcup_{i \in \{1,\ldots,M[k]\}} \{b_i[k]|\omega_{ij}[k] = 1\}$

After this gating process, represented by matrix Ω, the association problem could be defined as the search for the optimal assignation of measurements to tracks, bounded by these conditions. Ω can be considered as a set of constraints over the hypotheses such that all hypotheses in the search space must satisfy $a_{ij} \leq \omega_{ij}$. The set of blobs finally assigned for each track $\hat{x}_j[k]$ is defined as:

$$Z_j = \bigcup_{i \in \{1,\ldots,M[k]\}} \{b_i[k]|a_{ij}[k] = 1\} \tag{8}$$

so that $Z_j \subseteq W_j$. A general combinatorial algorithm for video data association and tracking can be formalized with the steps indicated in Fig. 2:

In classical data association problems, a typical constraint is the one-to-one assignment: each observation comes from at least one object, and each object produces a maximum of one observation:

$$\sum_{i=1}^{M[k]} a_{ij}[k] \leq 1; \quad \sum_{j=1}^{N[k]} a_{ij}[k] \leq 1 \tag{9}$$

This one-to-one correspondence between observations and objects is due to the conditions of traditional wide-area, low-resolution sensors such as radar. This limitation was systematically assumed in the first applications to visual data association (Cox 1993; Cox et al. 1995), but it can be too restrictive for video processing under situations of occlusions and image splitting. Recent approaches have identified the problem and proposed the extension of previous algorithms to take into account the splitting/merging effects for visual data association (Kumar et al. 2006; Genovesio and Olivo-Marin 2004; Liu et al. 2009; Rasmussen et al. 2001;

Fig. 2 General Data Association and Tracking Algorithm

```
1. Gating phase. Compute matrix $\Omega[k] = \{\omega_{ij}[k]\}$;   $i = 1,\ldots M[k]$, $j = 1,\ldots N[k]$
    1.1. $\omega_{ij}[k] = 1$ if distance(att($b_i[k]$, $\hat{x}_j[k]$) $\le$ Th ) ; otherwise $\omega_{ij}[k] = 0$
2. Assignment phase. Search for the optimum subset of blobs to be assigned for each track
    2.1. Search matrix A [k] to optimize maximize likelihood (for all N predicted tracks)
            $A[k] = \underset{A[k]}{\arg\max}\, P(Z[k] \mid A[k], \hat{x}_{1,\ldots,N_{k-1}}[k-1])$
    2.2. For each predicted track $\hat{x}_j[k \mid k-1]$,   $j \in \{1,\ldots N[k]\}$
        2.2.1. Group assigned blobs: $Z_j = \bigcup_{i \in \{1,\ldots,M[k]\}} \{b_i[k] \mid a_{ij}[k] = 1\}$
        2.2.2. Update $\hat{x}_j[k \mid k-1]$ with attributes extracted from $Z_j$ to estimate $\hat{x}_j[k \mid k]$
        2.2.3. Predict for next iteration: $\hat{x}_j[k+1 \mid k]$
```

Sheikh et al. 2008). The detected blobs corresponding to each target must be re-connected before they are used to update each track (Genovesio 2004).

In any case, one of the keys for jointly tracking multiple objects is forcing exclusion constraints to avoid several tracks being coalesced into the same observations. In the case of multiple objects which may overlap, the likelihood of the image cannot be simply decomposed in the likelihoods of each individual object; instead, a joint likelihood of the whole image, given all objects, needs to be constructed. In this way, the JPDA enumerates the association alternatives in order to first mark and remove those with several tracks merged with common measurements. Then, as a probabilistic solution, JPDA keeps the "average hypothesis," weighting all feasible hypotheses remaining after the discarding process (this is the big difference with respect to simple PDA which simply weights all hypotheses). In a more complex parallel process, the MHT matches a variable number of extracted points with tracks, allowing for assignments, missed observations (not-updated tracks) and false observations (discarded measurements), keeping in memory alternative hypotheses, each one containing a collection of tracks updated with mutually exclusive sets of measurements.

The main problem with combinatorial association techniques, even with the most efficient ones such as MHT or JPDA, is the exponential increase in computation resources as the complexity of situation increases, even more if the one-to-one constraint is removed. Moreover, the constraints on assignment decisions are sometimes insufficient to avoid failures with persistent complex situations such as long occlusions, noise from active objects, large shadows, etc., and a higher-level reasoning dealing explicitly with occlusion or other contextualized events needs to be included to avoid tracking failures (Malik and Russell 1996; S´anchez et al. 2008). This is the main reason why the soft-computing techniques mentioned in Sect. 2 are appropriate for dealing with these types of problems

and compute the association likelihood avoiding combinatorial analysis. Another interesting aspect is the computation of the likelihood function $P(Z[\mathrm{k}]|\mathbf{A}[k], \hat{x}_{1,\ldots,N_{k-1}}[k-1])$. Most previous approaches have been based on statistical distances between centroids of the region of interest, but in the case of video a more complex analysis could be carried out to integrate more available information such as size, shape or orientation. In this work some heuristics are defined to represent geometrical conditions of data association and a fuzzy rule system is proposed to represent the relationships within this heuristic using the assignment matrix.

4 Proposed algorithm for video data association: fuzzy region assignment

In this section we present our proposal for visual data association. It is an extension of previous methodology defined by FDA, considering the specific problems appearing in video data, and the requirements to provide a competitive alternative which is efficient and strictly linear in resources in order to work in real time. The basic input information are the detected regions (blobs in the binary foreground image) so a geometric reasoning is used for data association. The traditional one-to-one constraint is removed, allowing multiple regions to be associated to multiple tracks, with the ability to deal with the usual split and merge situations in video scenarios, thus extending conventional approaches based on simplified point or centroid representation of targets.

The input variables are several attributes proposed to define the rules for fuzzy assignment. They are heuristics which allow the appropriate semantic granularity in the reasoning process, using simultaneously low and high-level information. Then, the output variable (confidence level) is used to compose a synthesized measurement (pseudoblob) which is finally assigned to the track. This composition allows again the integration of low-level information (detected regions) with high-level tracks in order to avoid the instability or degradation of tracks through the composed measurement. This is equivalent to the joint multi-target analysis performed in classical combinatorial systems, but also allows the definition of rules by analyzing the causes of observations: dynamic and static occlusions, presence of maneuvers, changes of shape/orientation, etc. As mentioned before, even an exhaustive combinatorial search may not be enough to guarantee that merging situations with conflicting tracks are dealt with appropriately, since additional elements are needed: a higher-level reasoning process and use of additional information, not naturally included in a pure Bayesian framework working with observations, estimates, priors and likelihoods.

5 Video tracking based on segmented regions (blobs)

The detected regions are represented, as in other typical approaches, with a rectangular box, $b[k] = [x[k], y[k], wx[k], wy[k]]^t$, while the tracks contain this estimated information and its time derivatives for the targets, extrapolated from last update (T seconds) by means of a first-order approximation:

$$\begin{bmatrix} \hat{x}_j[k|k-1] \\ \hat{y}_j[k|k-1] \\ \hat{w}x_j[k|k-1] \\ \hat{w}y_j[k|k-1] \\ \hat{\dot{x}}_j[k|k-1] \\ \hat{\dot{y}}_j[k|k-1] \\ \hat{\dot{w}}x_j[k|k-1] \\ \hat{\dot{w}}y_j[k|k-1] \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\begin{bmatrix} \hat{x}_j[k-1|k-1] \\ \hat{y}_j[k-1|k-1] \\ \hat{w}x_j[k-1|k-1] \\ \hat{w}y_j[k-1|k-1] \\ \hat{\dot{x}}_j[k-1|k-1] \\ \hat{\dot{y}}_j[k-1|k-1] \\ \hat{\dot{w}}x_j[k-1|k-1] \\ \hat{\dot{w}}y_j[k-1|k-1] \end{bmatrix} + \begin{bmatrix} 0.5T^2 \\ 0.5T^2 \\ 0 \\ 0 \\ T \\ T \\ 0 \\ 0 \end{bmatrix} n_m[k] + \begin{bmatrix} 0 \\ 0 \\ 0.5T^2 \\ 0.5T^2 \\ 0 \\ 0 \\ T \\ T \end{bmatrix} n_s[k] \tag{10}$$

Notation “$k|k-1$” represents prediction (estimation at time k conditioned on observations up to time k-1), and “$k|k$” is filtering (estimation at time k conditioned on observations up to time k). Variables $n_m[k]$ $n_s[k]$ are plant-noise processes considered in the estimation algorithm, such as a Kalman Filter. The observation model relating measurements with vector states is $[x_{mj}[k] \quad y_{mj}[k] \quad wx_{mj}[k] wy_{mj}[k]]^t = h(\hat{x}^j[k]) = att(Z_j)$. These attributes are computed from Z_j, defined as the set of blobs associated to j-th track, $\mathrm{Z_j} = \bigcup_{i \in \{1,\ldots,M[k]\}} \{b_i[k] | a_{ij}[k] = 1\}$. Thus, the result of association is directly used in the measurement process of the estimation algorithm to update the track states corresponding to each object.

6 Input heuristics

The fuzzy system integrates different heuristics computed from gated blobs and target tracks to compute “confidence levels” that are used to weight each gated blob’s contribution to update the target track and its rectangular dimensions. The heuristics proposed to represent the situation for every blob are presented now, extracted from geometric analysis of blobs and predicted tracks. All of them consider the evaluation of a particular blob b_i potentially assigned to a particular track $\hat{x}_j$:

- *Overlap:* this heuristic evaluates the fact than the object originating the detected region is the same as the one represented by the (predicted) track $\hat{x}_j$. It is defined as:

$$\mathrm{Overlap}(\hat{x}_j, b_i) = \frac{\mathrm{Area}(\hat{x}_j \cap b_i)}{\min\{\mathrm{Area}(\hat{x}_j), \mathrm{Area}(b_i)\}} \tag{11}$$

  Its geometrical meaning is illustrated in Fig. 3. It is always in the interval [0,1]. The denominator to normalize is the minimum of areas so that the maximum value is obtained when same blob and tracks are the same object, both in the situation of splitting, $\mathrm{Area}(b_i) < \mathrm{Area}(\hat{x}_j)$, where the blob is included in the track, and the situation of merging, $\mathrm{Area}(\hat{x}_j) < \mathrm{Area}(b_i)$, where the track is contained within the blob.
- *Deformation.* This heuristic evaluates the deformation of the track when updated by the blob, thus, it assesses the possibility of the blob containing sources which are extraneous to the real object. The assimilation of blob to track would define a new area contained by the union

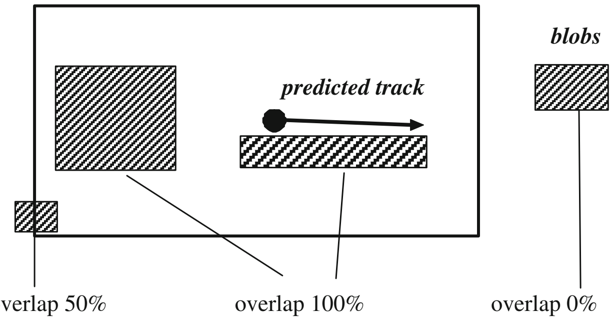


Fig. 3 Overlapping degree heuristic

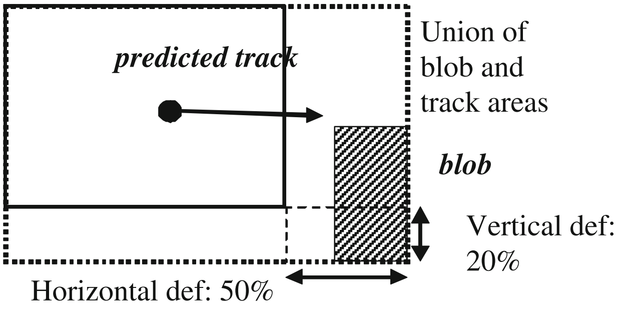


Fig. 4 Deformation degree heuristic

of both boxes, and the deformation is the difference of the new bound with respect to track bounds (see Fig. 4). It is computed by adding the deformation in the horizontal and vertical axis:

$$\text{Deform}(\hat{x}_j, b_i) = \frac{\text{Length}(\hat{x}_j \cup b_i) - \text{Length}(\hat{x}_i)}{\text{Length}(\hat{x}_j)} + \frac{\text{Width}(\hat{x}_j \cup b_j) - \text{Width}(\hat{x}_j)}{\text{Width}(\hat{x}_j)} \qquad (12)$$

Deformation is in the range $[0, \infty)$. It is easy to check that the deformation is zero if and only if the overlap is maximum, 100%

- *Density:* this heuristic evaluates whether the area defined by the union of blob and track comprises a motion area, through the ratio of detected regions to the total area. Analogously to previous variables, it assesses the presence of extraneous sources to the track, but this time directly through the detected image. Taking $I(x,y)$ as the binary foreground image (see Fig. 5), its value is computed as:

$$\text{Density}(\hat{x}_j, b_i) = \frac{\sum\limits_{x,y \in (\hat{x}_j \cup b_i)} I(x,y)}{\text{Area}(\hat{x}_j \cup b_i)} \qquad (13)$$

- *Conflict:* this component evaluates the situation of the blob being in conflict with other tracks (see Fig. 6). This problem appears when target trajectories are so close that track gates overlap and share the blob. The evaluation of blob conflict degree is done through the overlapping with the other existing tracks. In the case that more than one track is in conflict, the maximum overlapping degree is selected.

$$\text{Conflict}(\hat{x}_j, b_i) = \max_{k \in \{1, \ldots N\}, k \neq j} Ov(\hat{x}_k, b_i) \qquad (14)$$

Thus, the number of evaluations is constant, proportional to the product of tracks and blobs, since the conflict variable is evaluated only once for each pair, independently of the number of tracks involved in the conflict.

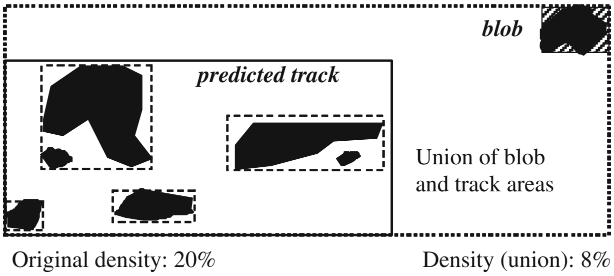


Fig. 5 Group density after blob re-connection

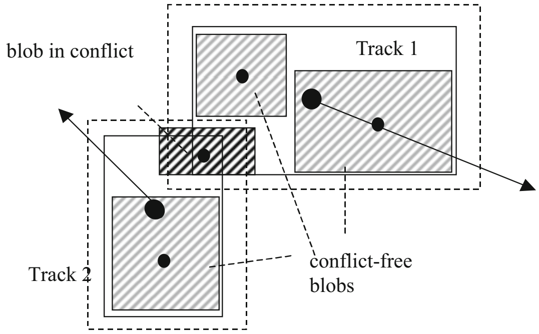


Fig. 6 Blob in assignment conflict with two tracks

7 Synthesis of state-update regions

The heuristics presented above provide input information to describe the situation and compute the correlation level of the *i*-th blob with respect to the *j*-th track, $\mu_j(b_i)$. They are computed for the set of blobs gated by the *j*-th track a time *k*, W_j: $W_j = \bigcup_{i \in \{1,\ldots,M[k]\}} \{b_i[k] | \omega_{ij}[k] = 1\}$.

The FRA method analyzes the situation represented by the four heuristics and computes the output to build a synthetic region, the pseudoblob, which contains the union of regions finally assigned to update each track, each one with an impact according to its reliability. The resulting update is not a direct weight of positions (centroids) like other previous approaches, because the structural information about target size and shape would be missed, and the track stability must be kept in complex situations. The resulting confidence output is used to define the group envelop, with a soft gradation between "reliable" conditions and "non-reliable" (due to conflict, noise, clutter, etc.), with special emphasis in avoiding corruption by multitarget merging.

The criterion to use the confidence level of every blob to update every track is: a blob with maximum confidence, 1, will directly update the track, while a blob with minimum confidence, 0, updates the track only with its intersection, if the intersection is not null. In any other case of confidence level, $\mu_j(b_i)$ the area assigned to the track is an intermediate region between the two extreme cases. The advantage of this fuzzy assignment scheme is twofold:

- All the interaction with the tracker is done at this level so the assignment can be combined with any estimation algorithm for tracking. For instance, in the case of zero-confidence, the track is extrapolated, but updated only with the overlapped area of detected regions.
- Each blob can be treated individually independently of the rest, avoiding the combinatorial problem of analyzing sets of blobs. Finally, the union of all computed regions is used to update the track.

Figure 7 illustrates the process with a single assigned blob and different values of confidence level.

Only when the confidence is maximum ($\mu = 1$) is the total area of the blob used to update the track. Otherwise, the area is reduced to the minimum case, consisting in the intersection between blob and track, $\hat{x}_j \cap b_i$. The expression to indicate this operation is:

$$b_i^{x_j}[k] = \mu\, b_i[k] + (1-\mu)\left(\hat{x}_j \cap b_i[k]\right) \qquad (15)$$

If the parameters to estimate shape are simply the position and enclosing box (x, y, wx, wy), the previous operation is applied to the four parameters to compute the assigned region to update the track. In the case that the overlap is null, no contribution at all is assigned and the track is kept unassigned. Notice that this scheme allows the spatial properties of objects to be maintained while the conflict lasts, the overlap between the merged area and predicted track is kept, and there is no loss since the assignments are kept within the area with overlap. Our region assignment proposal avoids complex interdependencies with the estimation phase, since it is only concerned with solving the association, with the gradation between track overlap (confidence zero) and total confidence, and total assignment. Then, the set of assigned blobs, with their corresponding confidences, is joined to form the synthesized pseudoblob finally used to update the track, as indicated in Fig. 8.

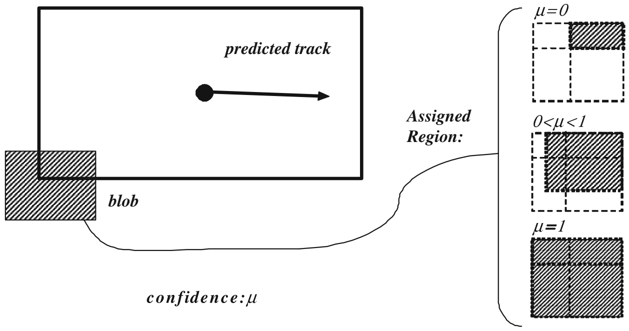


Fig. 7 Calculus of assigned region (single blob)

8 Algorithm overview and complexity analysis

Finally, the information variables expressed with the heuristics extracted from image operations are combined to define the appropriate actions, applying high- and low-level knowledge. The aggregation allows a soft approximation to the likelihood function which at the same time considers intuitive closeness criteria and exclusion constraints, equivalent to those defined with hard decisions, in order to track continuity. The algorithm for fuzzy region assignment is formalized in Fig. 9:

So, the input–output relationships are not computed with analytical or statistical formula, but through a set of rules used to synthesize the appropriate output for each situation. The general idea is that the result of fuzzy assignment contains the proper action to take under a set of particular extreme conditions to guarantee track continuity. An example is:

IF overlap IS <*LOW*> AND deformation IS <*MEDIUM*> THEN confidence is <*MEDIUM*> IF conflict IS NOT <*LOW*> THEN confidence IS <*ZERO*>

The system has been built based on the analyzed behavior of the tracking system observing the defined input and output variables, using human expertise and adjustments done by direct inspection and available input–output data. The heuristics proposed present the input variables with linguistic labels (small, medium, big), identifying the regions in which the universe of discourse is partitioned to build the approximated relationship among the variables.

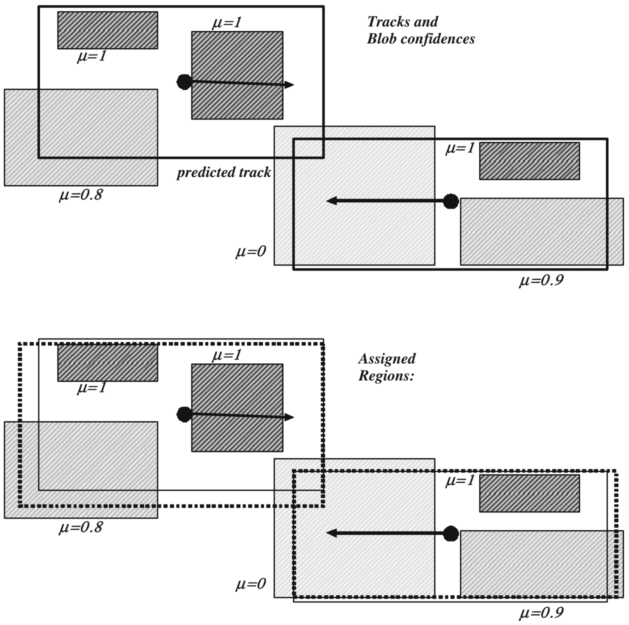


Fig. 8 Calculus of assigned regions (multiple blobs)

Fig. 9 Fuzzy Region Assignment Algorithm

```
1.    Gating phase. Compute matrix $\Omega[k] = \{\omega_{ij}[k]\}; \quad i = 1, \ldots M[k], j = 1, \ldots N[k]$
 1.1.  $\omega_{ij}[k] = 1$ if $\text{distance}(\text{att}(b_i[k], \hat{x}_j[k]) \leq Th)$ ; otherwise $\omega_{ij}[k] = 0$
2.    Assignment phase. Compute the assigned region for each track
 2.1.  For each predicted track $\hat{x}_j[k \mid k-1], \quad j \in \{1, \ldots N[k]\}$
      2.1.1. $b_{update} = \emptyset$
      2.1.2. For each overlapped blob $b_i[k]$ in $\tilde{W}_j = \{b_i[k] \mid \omega_{ij}[k] = 1 \wedge \hat{x}_j \cap b_i \neq \emptyset\}$
          2.1.2.1.    $Ov = \text{Overlap}(\hat{x}_j, b_i)$
          2.1.2.2.    $Df = \text{Deformation}(\hat{x}_j, b_i)$
          2.1.2.3.    $De = \text{Density}(\hat{x}_j, b_i)$
          2.1.2.4.    $Cf = \text{Conflict}(\hat{x}_j, b_i)$
          2.1.2.5.    $\mu_j(b_i) = \text{FIS}(Ov, Df, De, Cf)$
          2.1.2.6.    $b_i^{x_j}[k] = \mu_j(b_i)\, b_i[k] + (1 - \mu_j(b_i))\left(\hat{x}_j \cap b_i[k]\right)$
          2.1.2.7.    $b_{update} = b_{update} \cup b_i^{x_j}[k]$
      2.1.3. Update $\hat{x}_j[k \mid k-1]$ with $b_{update}$ to estimate $\hat{x}_j[k \mid k]$
      2.1.4. Predict for next iteration: $\hat{x}_j[k+1 \mid k]$
```

The starting point was the set of heuristic rules applied in order to allow a general-purpose tracker to work with visual data.

The modeling of knowledge may be done in different ways depending on the type of information available. There are systems that have available plenty of both data input and outputs, as there is a historical data scenario. In this situation, using automatic learning techniques, the system can be modeled and adjusted, in a similar way as the works based on neural networks (Zhu 1994) or neurofuzzy systems (Turkmen et al. 2004) to approximate the assignment probabilities. There are other situations in which these data are not available or are very partial. For example, cases involving big risks are very rare (such as nuclear accidents, air crashes,…). In this situation, the application of automatic learning techniques fail because it is difficult to establish patterns or correlations that model the abnormal regime, and the system must be modeled only with the relationship that experts of the problem can establish.

The determination of the fuzzy membership function is crucial when applying a fuzzy system to a certain problem. There is not a general method available to take this decision, but membership functions are determined in many cases manually. The grade of membership of the linguistic variables is the key element in reasoning, and they are developed following diverse criteria depending on the application (heuristic determination, theoretical analysis, model of human concepts, etc.). In the case of sensor-based multitarget tracking, these techniques usually apply statistical inputs so that the membership function estimation can be based on statistical analysis such as the possibility/probability principle developed by Singh et al. for optimal membership generation in sensor data processing (Singh et al. 1997). Other authors, such as (Aziz et al. 1999) propose the use of fuzzy clustering methods to avoid the development of rules, although from our point of view this means disregarding useful knowledge to solve the problem.

Besides the process of fuzzy partitioning of input heuristic and building the rules with expert knowledge, the authors have also extended the methodology to include optimization of rules and sets with NEFCLASS (Garcia et al. 2005). Usually the transformation of human knowledge into the fuzzy system is a first approximation whose parameters can be determined optimally with a learning process. However, results in (Garcia et al. 2005) showed the advantage was moderate in comparison with the performance observed with the initial system.

The fuzzy inference system is sketched in Fig. 10, and analyzed with the MATLAB fuzzy logic toolbox. In Fig. 11 we can see the membership functions for the input and output variables. The four input variables have three linguistic labels, whose fuzzy sets are specified as usual with trapezoidal membership functions. The case of the output variable, confidence, is more peculiar. The particular case of confidence NULL is considered, with singleton for this subset, in order to avoid sudden degradation of tracks at the moment when two tracks are mixed, since even a small contamination of a track with regions from another one ends in the merging effect.

With respect to algorithm complexity, we can compare the cost of a general data association algorithm (pseudocode in Fig. 2) based on hypotheses enumeration with the proposed FRA algorithm. In the general association process, considering a situation at time *k* in which we have *M*

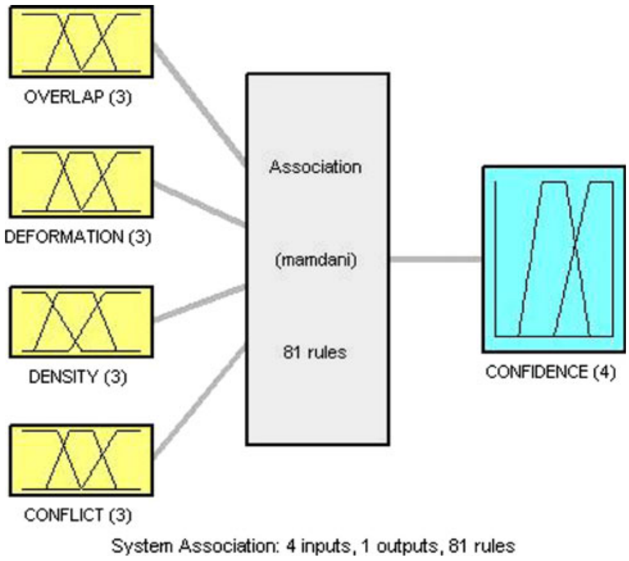


Fig. 10 Structure of the fuzzy inference system

blobs to be assigned to N tracks in the assignment matrix a_{ij} [k], we would have to enumerate all hypotheses in this search space of binary variables.

With the constraints of one-to-one assignment (Eq. 9), each observation comes, at least from one object, and each object produces, at maximum one observation), the number of possible assignments of M measurements to N tracks is given by:

$$N_H^{\text{one-to-one}} = \frac{\max(N+1, M)!}{|N+1-M|!} \tag{16}$$

where 1 is added to $N[k]$ because of the possibility of labeling each blob as not belong to any track, "null track," $\hat{x}_0$. However, in video applications this condition is not realistic, it seems much more reasonable to drop the first constraint and allow that multiple blobs can be assigned to the same track (split effect of bad segmentation). In this case, and keeping the first constraint that every blob only can be assigned to one track (considering the null track for extraneous sources), the size of search space is increased to:

$$N_H^{\text{many-to-one}} = (1+N)^M \tag{17}$$

Finally, in the case that occlusions and overlap appear and are considered in association, the merging effect could result in the same blob being assigned to more than one tracks (opening the problem of dividing it). In this case, the more general, the assignment matrix has not any constraint and the size of search space is the total number of combinations form binary matrix A (extended again with the null track):

$$\mathrm{N}_H^{\text{many-to-many}} = 2^{(1+N)M} \tag{18}$$

This last is the most general case, it means a total search of potential combinations of values in the binary matrix A. The graphical representation of the search is in Fig. 12, it would directly consider the assignment matrix, containing $N(1 + M)$ bits:

Classical algorithms which enumerate hypotheses to compute the optimum, such as MHT of JPDA, suffer from this exponential complexity on the number of objects, although it is usually bounded by a maximum number of hypotheses searched. The use of evolutionary algorithms in association (Angus et al. 1993) (Patricio 2008) allows a more efficient search in the assignment space, and it is usual also the definition of a maximum number of evaluations. Finally, suboptimal approaches which assign individually the closest observation to each track, such as Nearest Neighbor, or group first the image regions with a connected components analysis (Silva 2005) allow a linear dependence on the problem but are more vulnerable to failures under complex situations.

If we turn to the general FDA algorithm, with respect to the number of evaluated rules in a general case, if there are L input attributes, $\xi_1, \ldots, \xi_L$, for each k-th attribute with fuzzy domain of labels $\{L_1, \ldots, L_{N_k}\}$, the total number of rules is:

$$N_{\text{Rules}} = \prod_{k=1}^{L} N_k \tag{19}$$

In the example of (Singh et al. 1997) to correlate observations to objects in a situation of available data for position and speed, the input attributes for the fuzzy data association are Position Error (***PE***) and Speed Error (***SE***), defined as residuals between the measured values in radar plots and the estimated attributes of targets. They compare two examples: first, a single maneuvering target; and second, a situation of two targets crossing themselves at a short distance.

In the first case, there are two input variables, ***PE*** and ***SE***, taking values on five linguistic labels (Negative big, Negative small, Zero, Positive small, Positive big), and one output variable, correlation, with three linguistic variables (Low, Medium, High). Examples of rules are:

IF *PE* IS <*NB*> AND *SE* IS <*PS*> THEN *Correlation* is <*MED*>

where labels NB, PS and MED are linguistic labels with associated fuzzy sets which are previously defined. The total number of rules in this case is 5 × 5 = 25, corresponding to all combinations of input attributes, and the authors implement a system with the 25 rules. However, the second example, with only two targets and two observations, implies a much more complex situation. In this case, we have four independent measurements for the

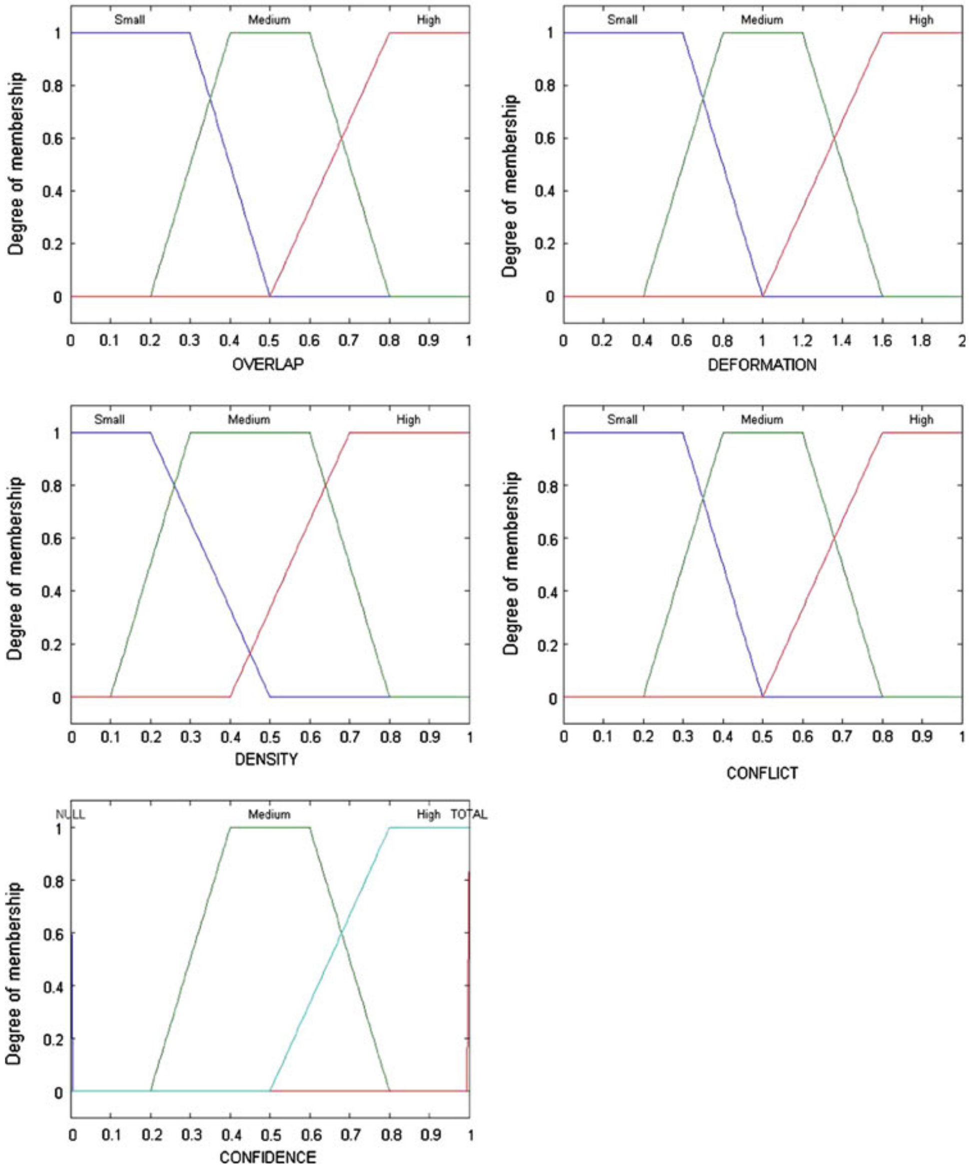


Fig. 11 Membership functions for input and output variables

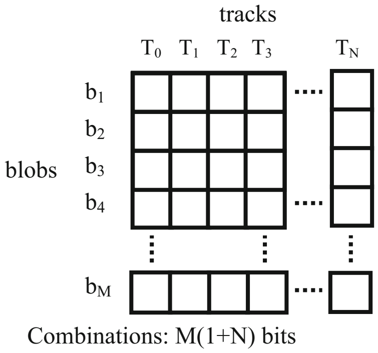


Fig. 12 Direct search space for data association matrix A

two target positions and speeds, resulting in eight evaluated errors forming the association cost matrices: {$\boldsymbol{PE_{11}}$, $\boldsymbol{PE_{12}}$, $\boldsymbol{PE_{21}}$, $\boldsymbol{PE_{22}}$}, {$\boldsymbol{SE_{11}}$, $\boldsymbol{SE_{12}}$, $\boldsymbol{SE_{21}}$, $\boldsymbol{SE_{22}}$}. In this case, we have a fuzzy inference system with eight input attributes and four output variables: {$\boldsymbol{Corr_{11}}$, $\boldsymbol{Corr_{12}}$, $\boldsymbol{Corr_{21}}$, $\boldsymbol{Corr_{22}}$} where $\boldsymbol{Corr_{ij}}$ is the degree of matching between the i-th observation to the j-*th* track. Authors of (Singh et al. 1997) explain that although it seems the number of rules also increases exponentially with the size of problem, as in classic combinatorial techniques, fuzzy techniques are very flexible when it comes to creating rules, with options to reduce the rule base or include fewer terms where imprecision can be tolerated.

On the other hand, FRA complexity shows a strict linear dependence on the problem size (number of blobs and tracks), with a constant number of evaluations per pair. It computes the input heuristics for all pairs in the assignment matrix, $M \times N$, and uses them to determine the update confidence through the fuzzy system. This linear dependence allows scalability to track a large number of objects in real time, no combinatorial analysis is done to build the sets of blobs with respect to set of tracks. Even for a conflict situation, where more than one track is in conflict with a set of blobs, the maximum operator is used to compute the conflict heuristic (Eq. 14), and it is not required the evaluation of different combinations of tracks and blobs (equivalent in complexity to enumerate alternative association hypotheses). The conflict degree is obtained from the overlap heuristic computed for every pair.

In the specific case developed, there are four input variables with three linguistic values, so the number of rules is $3^4 = 81$ rules, which are evaluated for the $N \times M$ blob-track pairs. This overcomes the typical limitation of FDA pointed out by other authors (Aziz et al. 2007; Singh et al. 1997) given by the exponential increase in the number of rules generated to cover a dense target environment. FRA uses the typical input space partitioning by antecedents of rules. But the strict limitation to $N \times M$ evaluations is another clear advantage with respect to other FDA approaches generating rules for all combinations of all blobs and track attributes in the association matrix.

9 Experimental results

In this section, we present a performance analysis and a comparison of the proposed fuzzy region assignment algorithm (**FRA**) described in previous sections with respect to other well-known real-time tracking multiple video targets, among them:

Particle Filtering algorithm (**PF**) is one of the most powerful algorithms in visual tracking (Isard 1998; Arulampalam et al. 2002; Ristic et al. 2004; Pérez et al. 2004; Xu and Li 2005; Loza et al. 2008) and relies on sample-based reconstruction of probability density functions of tracks.

A combinatorial data association method (Patricio 2008), which can be characterized as a ‘‘hard’’ association of the sequence of measurements to all tracks, based on certain cost criterion, processing the update stage. In our case, we have implemented an algorithm from the Estimation Distribution Algorithms (EDAs) family, specifically the Univariate Marginal Distribution Algorithm (**UMDA**) (Mühlenbein 1997).

A Connected Components (**CC**) tracking algorithm (Silva 2005), which uses a nearest neighbor strategy to determine the blob-to-track assignment.

The system described in this work has been implemented in Microsoft Visual C++, based on the “visual surveillance” algorithms incorporated in the Open Source Computer Vision Library (OpenCV). The system was tested on a DELL PE1950 Quad-Core Xeon E5310 1.6 GHz/ 2 × 4 MB 1066FSB.

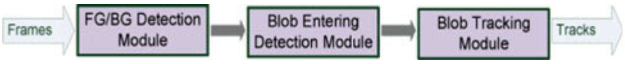


Fig. 13 The OpenCV ‘visual surveillance’ algorithms

The OpenCV ‘‘visual surveillance’’ algorithms use the pipeline structure depicted in Fig. 13. The input data for the pipeline is the image of current frame and the output data is the information on track position and size. The “FG/BG Detection” module performs foreground/background segmentation for each pixel; the “Blob Entering Detection” module uses the result of the “FG/BG Detection” module to detect new blob objects which entered the scene in each frame; and the “Blob Tracking” module is initialized by the “Blob Entering Detection” results and it tracks each newly entered blob. This pipeline structure allows us to exchange easily the four different algorithms described above for the “Blob Tracking” module, and to maintain the same execution conditions, i.e. using the same “FG/BG Detection” and “Blob Entering Detection” modules. All the tracking methods are initialised by the automatically detected blobs, however, some crucial parameters, such as a number of particles ($N = 100$) and the covariance of the random walk model for the PF, had to be predetermined manually for a wide range of video sequences. We have fixed the same ‘‘FG/BG Detection” module for every test that we have carried out. The selected module was the OpenCV implementation of the adaptive background mixture models for real-time tracking (Stauffer 1999).

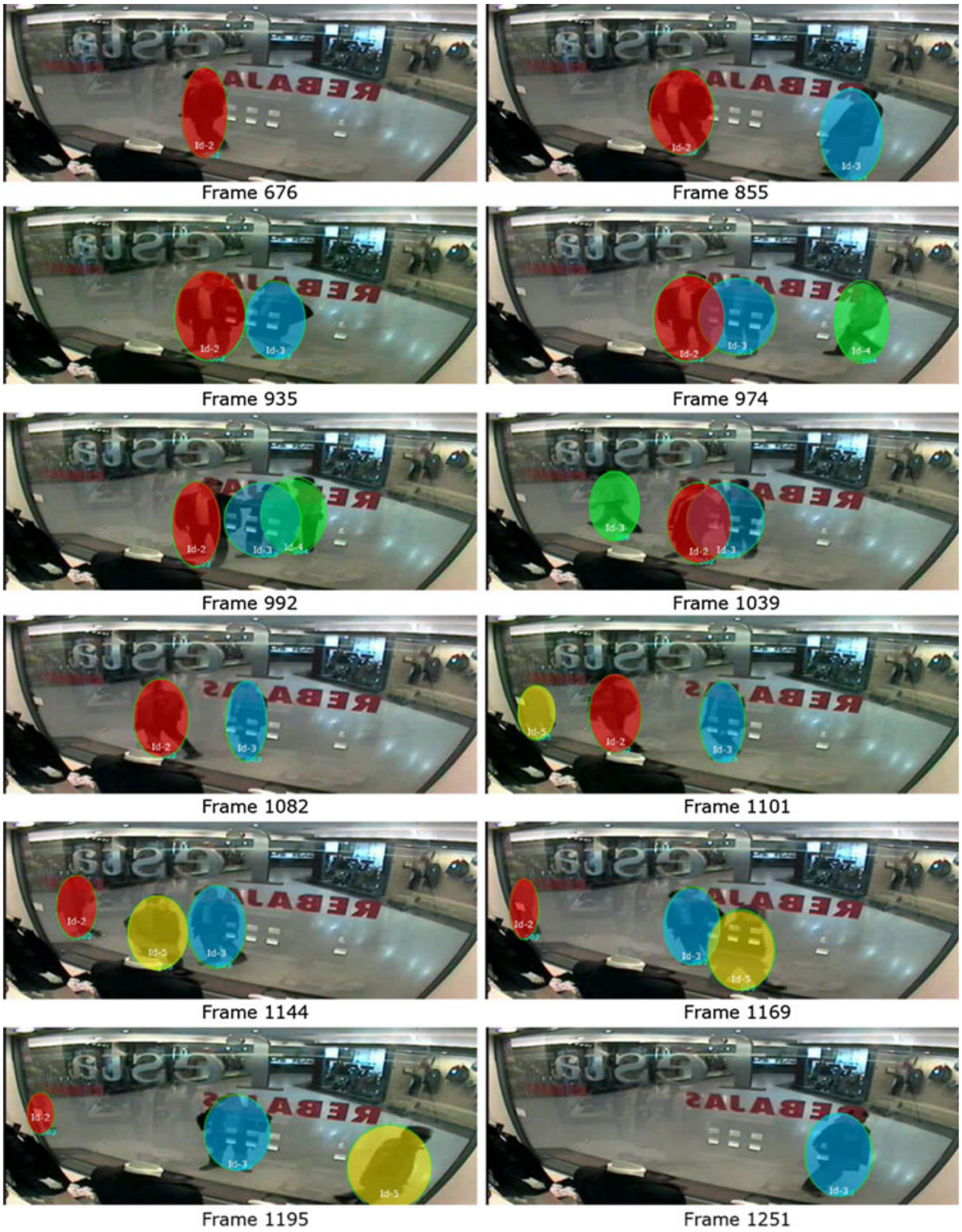
The performance of the four algorithms was evaluated with two well-known datasets:

1. The Performance Evaluation of Tracking and Surveillance (PETS) dataset (PETS 2002). Among numerous scenarios available through PETS, we have chosen a minute-long sequence from the PETS2002 workshop, where the underlying task was to track pedestrians in indoor video sequences of a shopping mall. The sequence contains multiple closely-spaced objects

Table 1 Quality measures of the algorithms applied to PETS2002

	mean TPF	std TPF	LTP	FPS
FRA	1.7505	0.8804	0.0019	10.39
PF	1.8626	1.1681	0.0037	2.58
UMDA	1.5981	0.8642	0.0093	6.56
CC	1.4916	0.7839	0.0131	9.39

Fig. 14 Performance of our proposed FRA tracker in a complex dataset from PETS2002



Frame 676

Frame 855

Frame 935

Frame 974

Frame 992

Frame 1039

Frame 1082

Frame 1101

Frame 1144

Frame 1169

Frame 1195

Frame 1251

Table 2 Quality measures of the algorithms applied to SQUASH

	mean TPF (ideal = 2)	std TPF	LTP	FPS
FRA	2,0050	0.2926	0.0009	11.67
UMDA	1.8258	0.3490	0.0016	12.40
CC	2.1688	0.6251	0.0027	10.80
PF	2,3559	0.7585	0.0032	2.74

(pedestrians), similar in shape and color to each other and to some elements present in the background. The challenges specific to the PETS2002 dataset result from the pedestrians appearing at a wide range of distances and angles with respect to the camera, thus introducing shape scaling and distortions. Moreover, the scene is recorded from behind a shop-window, which partially reflects the objects of interest.

2. The Computer Vision Based Analysis in Sport Environments (CVBASE) dataset. The CVBASE 2006 dataset (Machine Vision Group 2001) was filmed during a tournament of recreational players. The videos were recorded in S-VHS video-recorder, using a bird’s eye view with wide-angle lens. The videos were digitized to digital video format with 25 fps, with a resolution of 384 × 576 and M-JPEG compression. The selected video is a zenithal record of two players playing squash (SQUASH). They are in close proximity to each other, they are dressed similarly and are

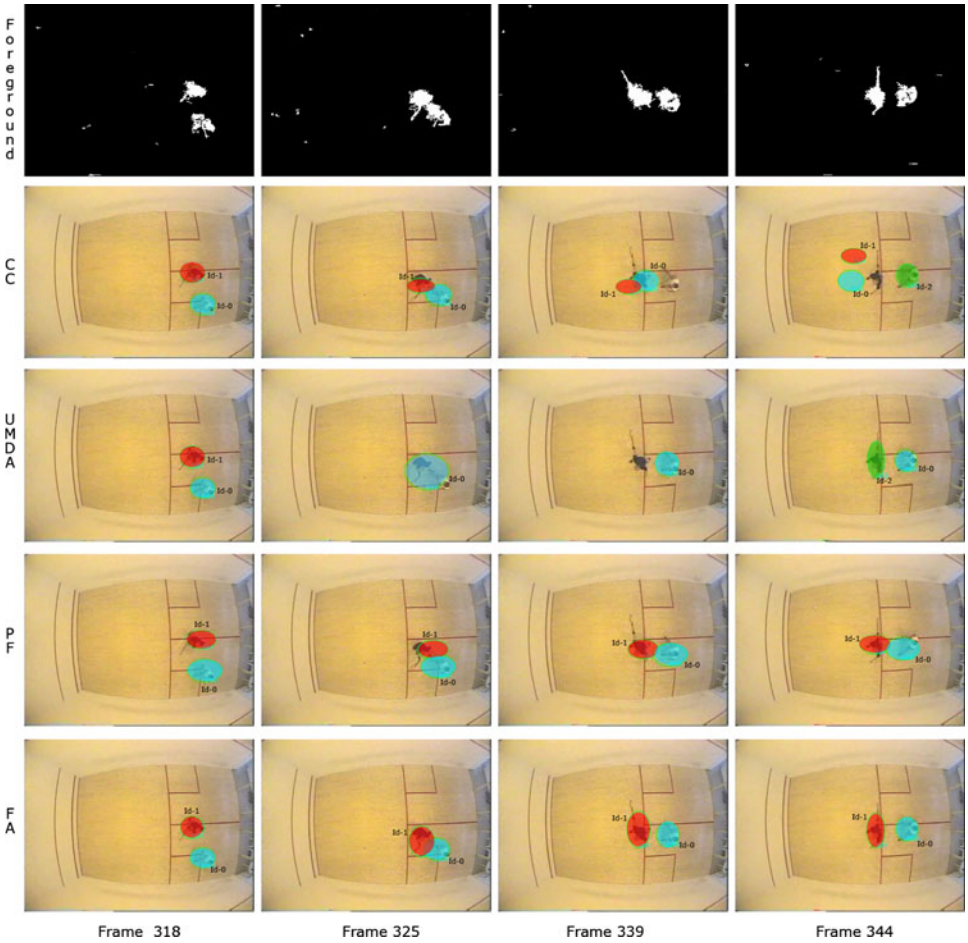


Fig. 15 Different tracking results for frames 318, 325, 339 and 344. First row depicts the blobs detected for each frame. They are the input for the four tracking methods. The tracking results are shown along the last four rows using CC, UMDA, PF and FRA algorithms, respectively

moving quickly, and there are constant crossings between players, which make for a challenging sequence.

10 Evaluation metrics

Tracking methods can be evaluated on the basis of whether they generate correct mobile object trajectories. A qualitative comparison of tracking algorithms can be based on the ability to maintain the number of targets during the sequence video and to provide an optimal solution to the cost function minimization problem used for establishing correspondence [Yilmaz et al. 2006]. Therefore, the metrics that allow us to provide formal comparisons among the algorithms tested are:

Tracks per Frame (**TPF, std TPF**): evaluates the continuity of the tracks. An optimal tracker results in the TPF referred to as an ‘‘ideal’’ and a low standard deviation. The TPF below the ‘‘ideal’’ indicates that the tracker lost the continuity of the tracks (merge effect) and, conversely, higher than ‘‘ideal’’ TPF indicates that the tracker had an excess of tracks (split effect). In datasets where the number of tracks is known and fixed in time (for instance, SQUASH), this metric should approach its known ideal value (two players in SQUASH). This metric is the summary of the tracking algorithm performance over a representative number of frames. Thus, an algorithm performs better the process of tracking the more closely to its ideal value and a standard deviation lower.

Lost Track Probability (**LTP**): determines the probability of losing a track in a given frame. Note that this measure has also been used in (Kan et al. 1996), among others).

Frames per Second (**FPS**): the rate of processed images by the tracking algorithms; high values imply that an algorithm is less computationally demanding.

11 Results and discussion

In the following tables the quality measurements of the CC (Connected Components), UMDA (Univariate Marginal Distribution Algorithm), PF (Particle Filtering) and our proposed FRA (Fuzzy Region Assignment) algorithm

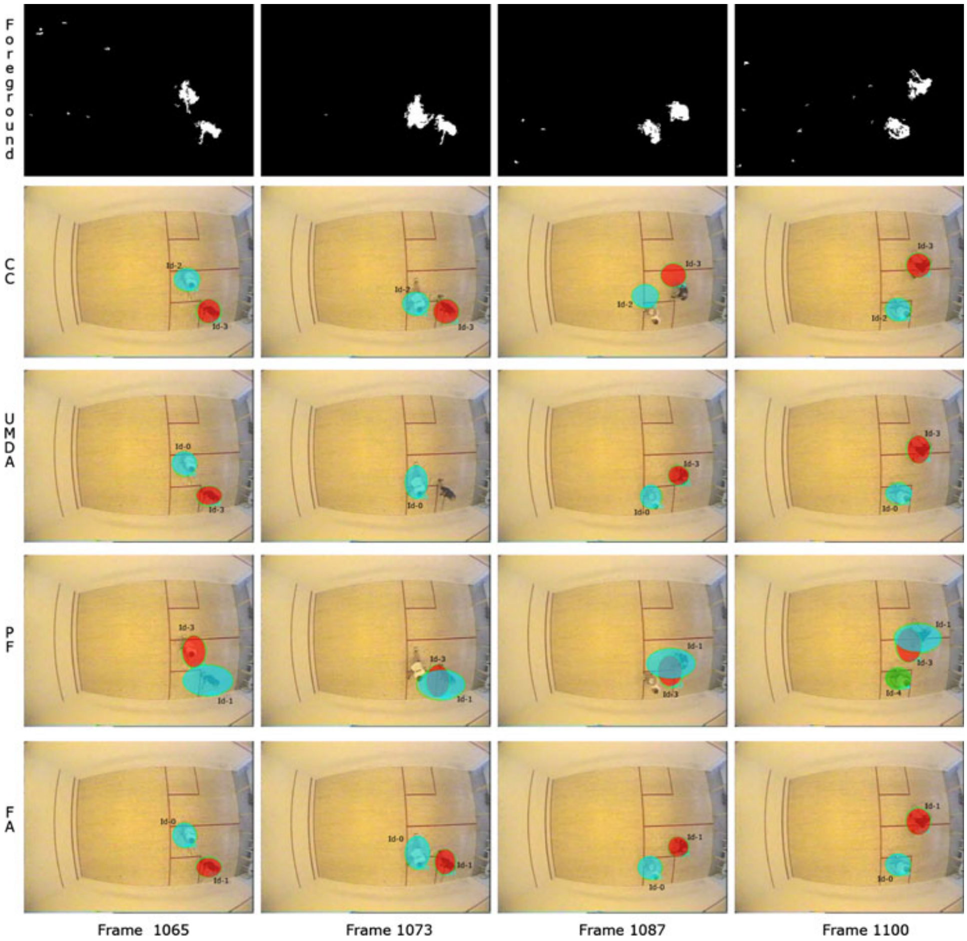


Fig. 16 Different tracking results for frames 1065, 1073, 1087 and 1100. First row depicts the blobs detected for each frame. They are the input for the four tracking methods. The tracking results are shown along the last four rows using CC, UMDA, PF and FRA algorithms, respectively

applied to the PETS2002 and SQUASH sequences are presented. Additionally, some videos showing the performance of these algorithms can be downloaded from our website (http://ww.giaa.inf.uc3m.es/softcomputing-2008).

11.1 PETS2002

The results of the numerical tracking evaluation for PETS2002 are summarized in Table 1.

The tracking methods in the table are score-sorted according to greater similarity to mean TPF. It has been determined a priori that the average “ideal” TPF in the sequence used is 1.79. According to the TPF, all methods perform comparably, with our proposed FRA algorithm being the most accurate. Note that since TPF is a “blind” measure, i.e. it does not validate the tracks, the TPF scores should be taken into account along with their standard deviation, std TPF, and the LTP measure. The LTP metrics result in the same ranking as TPF: in order of decreasing quality: FRA, PF, UMDA, CC; LTP for CC being almost seven times higher than LTP for FRA. It should be noted, however, that the three best methods, FRA, PF and UMDA, have larger std TPF than poorly performing CC.

An important metric in real-time video tracking is the speed at which the images are processed by the algorithms, measured by the FPS metrics. According to the FPS results shown in Table 1, the PF tracker entails more computation load, which in turn reduces its processing capacity. To further illustrate this, the processing time needed for one track has been measured. In our simulations, the average of 386.916 and 2.003 ms was obtained for the PF and the FRA algorithms, respectively.

Thus, it can be concluded that our proposed FRA tracker is more precise than the powerful PF algorithm and has a greater capacity to process large amount of tracks in the multitarget scenario. This is because the PETS2002 presents a complex scenario analysis. A complex scenario is that in which at least one of the following situations occur: dynamic background, several objects, objects enter and/or leave the scene, objects interact with each other producing occlusions, cross, merge and split effects, etc. An instance of the performance of our proposed FRA algorithm in a complex scenario is depicted in Fig. 14, where four people are tracked from frame 676 to frame 1251. In this sequence, tracks with id-2 (red ellipse) and id-3 (blue ellipse) enter the scene and stop together in the middle of

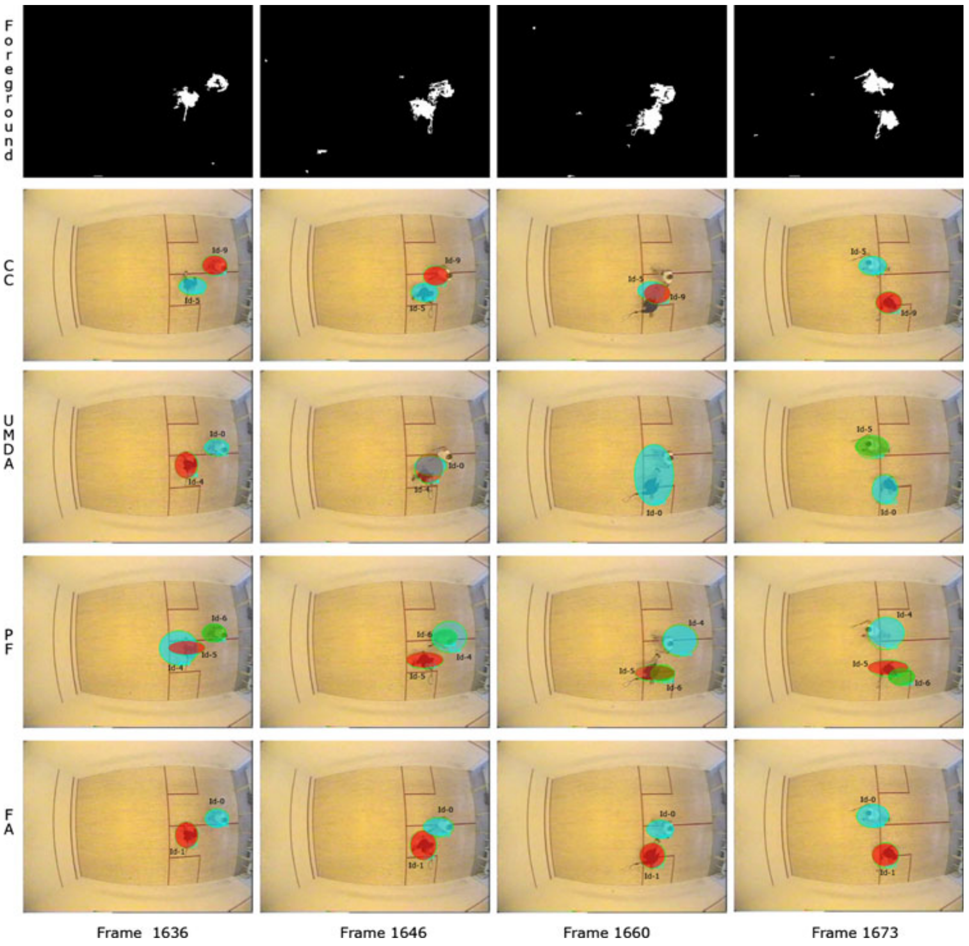


Fig. 17 Different tracking results for frames 1636, 1646, 1660 and 1673. First row depicts the blobs detected for each frame. They are the input for the four tracking methods. The tracking results are shown along the last four rows using CC, UMDA, PF and FRA algorithms, respectively

the window (frame 935). In frame 974, the track with id-4 (green ellipse) enters the scene and goes from right to left (frame 1039). Track with id-2 (red ellipse) starts to walk to the left side (frame 1082) meanwhile track with id-5 (yellow ellipse) comes into the scene and goes from left to right (frames 1011, 1144, 1169 and 1195). Finally, track with id-3 (blue ellipse) keeps still in the scene (frame 1251). We can observe that there have been several interactions among people (grouping and crossing events) and none have lost their identity.

11.2 SQUASH dataset

In the following test video, SQUASH, given the normal dynamics of game, there are many situations in which the players move very close to each other, making abrupt movements. This makes the tracking problem harder. The results of the quality measures are shown in Table 2.

We can see that we obtain results similar to those in the PETS2002 dataset. The most appreciable difference is the poor precision measures (TPF and LTP) of the PF algorithm. This is mainly due to two factors. First, the SQUASH dataset presents a challenging scenario where the players are very close and are moving quite quickly and constantly switching places. These continuous changes of movement and accelerations cause PF to fail in the estimation of the tracks. On the other hand, when the PF algorithm loses a track, these track turns into a ‘‘ghost track’’ (see Fig. 16 frames 1073, 1087 and 1100, and Fig. 17 all frames for the performance of PF algorithm). Some heuristic should be included in the PF algorithm described in (Loza 2008) to erase ‘‘ghost tracks’’ and to avoid this malfunction. These are the reasons why PF receives the highest value for TPF and std TPF.

In most of the situations, algorithms perform acceptably and provide similar results, with our proposed Fuzzy data Association algorithm being the most precise (TPF of 2.0050 and std TPF of 0.2926). Regarding the capacity of process, PF means greater computational requirements (2.74 frames per second). The main differences appear when players are close together making quick movements. In order to illustrate the performance of the four algorithms behind this condition, we show the behavior of the tracking algorithms with three sequences of the SQUASH dataset (see Figs. 15, 16, 17). First row shows the output of the detection algorithm for each frame. The following four rows are the tracking result of the CC, UMDA, PF, and FRA algorithms, respectively. In first s equence (Fig. 15), we can observe how the PF and FRA algorithms track both players well while CC and UMDA algorithms fail in the tracking process. In the next two sequences, our proposed Fuzzy data Association (FRA) is the only algorithm that is able to track both players without any problem.

12 Conclusions

The proposed visual tracker based on fuzzy region assignment extends previous approaches of fuzzy data association to the problematic area of video data, using a representation which allows manipulation of concepts at different levels and a geometrical reasoning based on expert experience. The main contribution consists in the representation of variables with an important semantic effect to represent the visual data association process and drive the tracker with the appropriate decisions.

The system allows improving the ratio performance/resources with respect to representative visual tracking systems, and a significant reduction in the number of rules with respect to previous approaches by making use of fuzzy systems for data association with other sensor tracking applications. It has shown competitive results and efficiency when working in real conditions after detailed evaluation in representative situations.

Acknowledgments Research supported by projects CICYT TIN2008-06742-C02-02/TSI, CICYT TEC2008-06732-C02-02/TEC, SINPROB and CAM MADRINET S-0505/TIC/0255.

References

Agrawal R, Imielinski T, Swarmi A (1993) Mining association rules between sets of items in large databases. In: Proceedings of the ACM SIGMOD international conference on management of data, Washington, DC, pp 207–216

Angus J, Zhou H, Bea C, Becket-Lemus L, Klose J, Tubbs S, (1993) Genetic algorithms in passive tracking. Claremont Graduate School, Math Clinic Report, May 1993

Arulampalam M, Maskell S, Gordon N, Clapp T et al (2002) A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans Signal Proc 50(2):174–188

Aziz AM, Tummala M, Cristi R et al (1999) Fuzzy logic data correlation approach in multisensor-multitarget tracking systems. Signal Process 76(2):195–209

Aziz AM, Elkobba K et al (2007) Fuzzy track-to-track association and track fusion approach in distributed multisensor-multitarget multiple-attribute environment. Signal Process 87(6):1474–1492

Bogner RE, Bouzerdoum A, Pope KJ, Zhu J et al (1998) Association of tracks from over the horizon radar. IEEE Aerosp Electron Syst Mag 13(9):31–35

Brodsky T, Cohen R, Cohen-Solal E, Gutta S, Lyons D, Philomin V, Trajkovic M (2001) Visual surveillance in retail stores and in the home. In: Advanced video-based surveillance systems, Chap 4. Kluwer, Boston, pp 50–61

Cai Y, de Freitas N, Little J (2006) Robust visual tracking for multiple targets. In: European conference on computer vision 2006, pp 107–118

Chang YL, Aggawal JK (1991) 3d structure reconstruction from an ego motion sequence using statistical estimation and detection theory. In: Proc. IEEE workshop on visual motion, pp 268–273

Chen YM, Huang HC (2000) Fuzzy logic approach to multisensor data association. Math Comput Simul 52(5–6):399–412

Chen HT, Lin HH, Liu TL (2001) Multi-object tracking using dynamical graph matching. Proc IEEE Conf Vis Pattern Recognit 11:210–217

Cho J-S, Yun B-J, Yun-Ho Ko Y-H et al (2007) Intelligent video tracking based on fuzzy-reasoning segmentation. Neurocomputing 70(4–6):657–664

Cox IJ (1993) A review of statistical data association techniques for motion correspondence. Int J Comput Vis 10(1):53–66

Cox IJ, Hingorani SL (1996) An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. IEEE Trans Pattern Anal Mach Intell 18(2):138–150

Cox IJ, Miller ML et al (1995) On finding ranked assignments with application to MultiTarget tracking and motion correspondence. IEEE Trans Aerosp Electron Syst 32(1):486–489

Cucchiara R, Grana C, Patri A, Tardini G, Vezzani G (2004) Using computer vision techniques for dangerous situation detection in domotic applications. In: Proc. IEEE workshop on intelligent distributed surveillance systems, London, pp 1–5

da Silva Pires D, Cesar R, Vieira M, Velho L (2005) Tracking and matching connected components from 3d video. In: 18th Brazilian symposium on computer graphics and image processing, 2005. SIBGRAPI 2005, 9–12 Oct 2005, pp 257–264

Ermin S, Sundararajan N, Saratchandran P (2000) Performance evaluation of a fuzzy data association algorithm for multitarget tracking (MTT). In: Proceedings of the IEEE 2000 national aerospace and electronics conference, 2000. NAECON Dayton, OH, pp 716–722

Ferryman JM, Maybank SJ, Worrall AD (2000) Visual surveillance for moving vehicles. Int J Comput Vis 37(2):187–197

Fleuret F, Berclaz J, Lengagne R, Fua P (2008) Multicamera People tracking with a probabilistic occupancy map. IEEE Trans Pattern Anal Mach Intell 30(2):267–282

Gad A, Majdi F, Farooq M (2002) A comparison of data association techniques for target tracking in clutter. In: Proceedings of the fifth international conference on information fusion, vol 2, pp 1126–1133

Garcia J, Besada JA, Molina JM, Portillo J, de Miguel G (2002) Fuzzy data association for image-based tracking in dense scenarios. In: IEEE international conference on fuzzy systems, Honolulu, Hawaii, May 2002

Garcia J, Molina JM, Besada JA, Portillo JI (2005) A multitarget tracking video system based on fuzzy and neuro-fuzzy techniques. EURASIP J Appl Signal Process (Special Issue on Advances in Intelligent Vision Systems: Methods and Applications, no. 14):2341–2358

Genovesio A, Olivo-Marin JC (2004) Split and merge data association filter for dense multi-target tracking. In: 17th int. conf. on pattern recognition, vol 4, pp 677–680

Greenhill D, Remagnino P, Jones GA (2002) VIGILANT: content querying of video surveillance streams. In: Remagnino P, Jones GA, Paragios N, Regazzoni CS (eds) Video-based surveillance systems. Kluwer, Boston, pp 193–205

Han H, Ran C, Zhu H, Wen R (2003) Multi-target tracking based on multi-sensor information fusion with fuzzy inference In: Proceedings of the sixth international conference of information fusion, vol 2, pp 1421–1425

Haritaoglu I, Harwood D, Davis L (1998) W4: who, when, where, what: a real time system for detecting and tracking people. In: Proceedings of the third international conference on automatic face and gesture recognition (FG'98), April 1998, pp 222–227

Haritaoglu I, Harwood D, Davis LS (2000) W4: real-time surveillance of people and their activities. IEEE Trans Pattern Anal Mach Intell 22(8):809–830

Hillis DB (1997) Using a genetic algorithm for multi-hypothesis tracking. In: 9th int. conf. on tools with artificial intelligence, Newport Beach, CA, USA

Isard M, Blake A (1998) Condensation—conditional density propagation for visual tracking. Int J Comput Vis 28(1):5–28

Javed O, Shah M (2002) Tracking and object classification for automated surveillance. In: European conference on computer vision, p IV:343 ff

Joo S-W, Chellappa R et al (2007) A multiple-hypothesis approach for multiobject visual tracking. IEEE Trans Image Process 16(11):2849–2854

Kan W, Krogmeier J et al (1996) A generalization of the pda target tracking algorithm using hypothesis clustering. Signals Syst Comput 2:878–882

Khan Z, Balch T, Dellaert F et al (2005) Multitarget tracking with split and merged measurements. Proc IEEE Conf Vis Pattern Recognit 1:605–610

Koller D, Klinker G, Rose E et al. (1997) Real-time vision-based camera tracking for augmented reality applications. In: ACM symposium on virtual reality software and technology, Lausanne, Switzerland

Krumm J, Harris S, Meyers B, Brumit B, Hale M, Shafer S (2000) Multi-camera multi-person tracking for easy living. In: Third IEEE int. workshop on visual surveillance, Ireland, pp 8–11

Kumar P, Ranganath S, Sengupta K, Weimin H et al (2006) Cooperative multitarget tracking with efficient split and merge handling. IEEE Trans Circuits Syst Video Technol 16(12):1477–1490

Leuven J, Leeuwen M, Groen F (2001) Real-time vehicle trakcing in image sequenes. IEEE Instrumentation and Measurement Technology Conference, Budapest, Hungary, May 2001

Liu J, Tong X, Li W, Wang T, Zhang Y, Wang H (2009) Automatic player detection, labeling and tracking in broadcast soccer video. Pattern Recognit Lett 30:103–113. doi:10.1016/j.patrec.2008.02.011

Loza A, Patricio MA, Garcia J, Molina JM (2008) Advanced algorithms for real-time video tracking with multiple targets. In: 10th international conference on control, automation, robotics and vision, ICARCV 2008, Hanoi, Vietnam, 17–20 Dec 2008

Machine Vision Group, U. o. L. (2001) Cvbase '06 workshop on computer vision based analysis in sport environments. http://vision.fe.uni-lj.si/cvbase06/. Accessed in 2007

Malik J, Russell S (1996) Final report for traffic surveillance and detection technology development. New traffic sensor technol-ogy. University of California

Medioni G, Cohen I, Bremond F, Hongeng S, Nevatia R (2001) Event detection and analysis from video streams. IEEE Trans Pattern Anal Mach Intell 23(8):873–889

Moeslund TB, Hilton A, Kru¨ger V et al (2006) A survey of advances in vision-based human motion capture and analysis. Comput Vis Image Underst 104(2):90–126

Mori G, Belongie S, Malik J (2005) Efficient shape matching using shape contexts. IEEE Trans Pattern Anal Mach Intell 27(11):1832–1837

Mühlenbein H (1997) The equation for response to selection and its use for prediction. Evol Comput 5:303–346

Novak V, Perfilieva I, Dvovrak A, Chen G, Wei Q, Yan P et al (2008) Mining pure linguistic associations from numerical data. Int J Approx Reason 48(2008):4–22

OpenCV, http://www.intel.com/technology/computing/opencv/index.htm. Accessed in 2006

Patricio M, Garcia J, Berlanga A, Molina JM (2008) Solving video-association problem with explicit evaluation of hypothesis using EDAS. In: 2008 IEEE congress on evolutionary computation (IEEE CEC 2008) within 2008 IEEE world congress on computational intelligence (WCCI 2008). Hong Kong, June 2008

Pe´rez P, Vermaak J, Blake A (2004) Data fusion for tracking with particles. Proc IEEE 92(3):495–513

PETS (2002) In: 3rd IEEE international workshop on performance evaluation of tracking and surveillance, (PETS'2002). pets2002.visualsurveillance.org. Accessed in 2007

Rasmussen C, Hager GD et al (2001) Probabilistic data association methods for tracking complex visual objects. IEEE Trans Pattern Anal Mach Intell 23:560–576

Reid DB (1979) An algorithm for tracking multiple targets. IEEE Trans Autom Control 24(6):843–854

Ristic B, Arulampalam S, Gordon N (2004) Beyond the Kalman filterparticle filters for tracking applications. Artech House, Boston

Sa´nchez AM, Patricio MA, Garcı´a J, Molina JM (2008) Occlusion management using a context-based tracking system. In: 3rd workshop on artificial intelligence techniques for ambient intelligence (AITAmI '08) special session on vision-based reasoning co-located event of European conference on artificial intelligence, Patras, Greece, 21–22 July 2008

Sengupta D, Iltis R et al (1989) Neural solution to the multiple target tracking data association problem. IEEE Trans Aerosp Electron Syst 25:96–108

Shams S (1996) Neural network optimization for multi-target multi-sensor passive tracking. Proc IEEE 84(10):1442–1457

Sheikh YA, Shah M et al (2008) Trajectory association across multiple airborne cameras. IEEE Trans Pattern Anal Mach Intell 30(2):361–367

Singh R-NP, Bailey WH et al (1997) Fuzzy logic applications to multisensor-multitarget correlation. IEEE Trans Aerosp Electron Syst 33:752–769

Stauffer C (1999) Adaptive background mixture models for real-time tracking. In: Proc. IEEE conf. on computer vision and pattern recognition, pp. 246–252

Stauffer C, Grimson W (1999). Adaptive background mixture models for real-time tracking. In: IEEE computer society conference on computer vision and pattern recognition, vol 2, Los Alamitos, CA. IEEE Computer Society

Tao H, Sawhney HS, Kumar R (2002) Object tracking with Bayesian estimation of dynamic layer representations. IEEE Trans Pattern Anal Mach Intell 24(1):75–89

Turkmen I, Guney K et al (2004) Cheap joint probabilistic data association with adaptive neuro-fuzzy inference system state filter for tracking multiple targets in cluttered environment. Int J Electron Commun 58:349–357

Xu X, Li B (2005) Particle filter for tracking with application in visual surveillance. In: 2nd joint IEEE international workshop on visual surveillance and performance evaluation of tracking and surveillance, Breckenridge, Colorado, USA

Xu M, Lowey L, Orwell J (2004) Architecture and algorithms for tracking football players with multiple cameras. In: Proc. IEEE workshop on intelligent distributed surveillance systems, London, pp 51–56

Yilmaz A, Javed O, Shah M (2006) Object tracking: a survey. ACM Comput Surv 38(4), article 13

Zhu J, Bogner R, Bouzerdoum A, Southcott M (1994) Application of neural network to track association in over the horizon radar. Proc SPIE 2233:224–235