Multi-target tracking with occlusion management in a mean field framework

C. Medrano, R. Igual, J. Martinez, C. Orrite

Multi-target tracking with occlusion management in a mean field framework

C. Medrano R. Igual J. Martínez
C. Orrite
Computer Vision Lab, Aragón Institute for Engineering Research
María de Luna 1, 50018, Zaragoza, Spain.

Abstract

In this paper we consider the problem of tracking multiple targets. Following a mean field approach, we obtain a cost function that depends on the means and covariances of each object. A simplification of this cost function leads to self consistent equations for the means, while the covariances are obtained using a normal Kalman filter. The iteration of the self consistent equations allows refining the solution and including the effect of occlusions. An implicit assumption of our model is that we can deduce the depth order from the joint state and that we can approximate the posterior of each object by a Gaussian. We show how this simple approach reduces the number of tracking failures in two sequences. The first one is a synthetic sequence of tennis balls moving on a background. The second one is a sequence of a football match taken from the VS-PETS data base.

1. Introduction

Tracking is an important task of Computer Vision with many applications in surveillance, analysis in sport environments etc. In particular, tracking similar objects is a challenging problem that has been managed typically with two kinds of algorithms. In the first kind we include JPDAF (Joint Probabilistic Data Association) and MHT (Multiple Hypothesis Tracker) [2]. A key point is that multiple measurements are assumed to be available and the proposed methods deal with the problems of data association, data fusion or multiple hypothesis generation. A second kind of algorithms includes an MRF (Markov Random Field). This MRF introduces pairwise potentials that prevent two trackers from latching onto the same target. The resulting model is not exactly solvable so that methods such as MCMC (Monte Carlo Markov Chain) [6] or SMFMC (Sequential Mean Field Monte Carlo) [16] have been used. MCMC samples the joint state space efficiently, while SMFMC performs an approximate inference over the associated graphical model. SMFMC needs an initial solution, for instance that obtained by independent trackers. Then, the solution is refined iteratively. Pairwise potentials play the main role in this iteration.

There also some other approaches that manage the problem of sampling in high dimensional state spaces and that cope with the systematic derivation of the observation density. For instance, MacCormick and Blake [8] incorporate an exclusion principle by which any edge feature can correspond to at most one target boundary. In addition an efficient sampling, partitioned sampling, is proposed for a particle filter algorithm.

However, in the present paper we argue that some of the problems associated with multi-target tracking can be solved if an occlusion reasoning is possible. Lanz [7] also argued in favour of an appearance likelihood that implements a physically-based model of the occlusion process. Several papers deal with the occlusion challenge. Rasmussen and Hager [12] considered a joint image likelihood given an object depth order, where occluded portions of objects are masked. Pixels predicted to be obstructed are ignored and those predicted to be visible are matched normally. Only the most likely depth order is considered in their data association filter. In [15] a variable is associated to the order of the objects. This hidden variable is also inferred at each time step after the execution of a particle filter algorithm. Sigal [13] introduced binary hidden variables to deal with occlusions between limbs in a pose estimation algorithm. These hidden variables represent the visibility of a given body part at a given pixel. The occlusions can be modelled as constraints and included in a non parametric belief propagation algorithm (PAMPAS) to perform inference in the graphical model of the human body. An analytical approximation is needed in order to manage the exponentially large number of occlusion masks [13, 14]. The Hybrid Joint Separable filter (HJS) [7] includes interaction and an explicit occlusion reasoning. In this filter the state must include information to deduce which object is responsible for the rendering function at a given pixel. Such a state could be a point in the 3D world or a state containing depth

information through a target scale. An approximate solution to obtain the posterior for each object can be embedded into a particle filter. McKenna et al. [9] use colour information to disam- biguate occlusion and to provide qualitative estimates of depth ordering and position for tracking groups of people.

Our goal is to incorporate an occlusion reasoning in a mean field approach to cope with the problem of tracking multiple targets. We assume that the object depth order can be deduced from the joint state and we approximate the posteriors by Gaussians. The philosophy is in some way similar to [7], but we work with Gaussian posteriors so that only means and covariances have to be updated at each time step. In our approach, we first assume independent and Gaussian measurement generation process for each object to obtain an approximate solution. Then, we refine the solution using mean field equations where occlusions are taken into account. These steps have a high parallelism with SMFMC algorithm as explained above [16, 10].

The contributions of this paper are three. Firstly, the development of a mean field cost function that includes complex likelihood function. Secondly, an approximation of it that allows obtaining a simple equation for the mean state of the objects. Finally, the third contribution is the experimental work based on a self-generated synthetic sequence and a sequence of a public data base. The experiments show that the proposed algorithm helps reduce the number of tracking failures. In the future we plan to deal with multimodality by means of a mixture of Gaussians.

The rest of the paper is organized as follows. In section 2 we briefly review the basis of mean field tracking and we show how it can be used with occlusions. In section 3 we show the experimental results. Finally, we sum up our contributions in section 4.

2. Theoretical background

We consider the usual assumption of a first order Markov Process in a Bayesian tracking:

$$p(X_t, Z_t) = \phi(Z_t|X_t) \int p(X_t|X_{t-1}) p(X_{t-1}|Z_{t-1}) \quad (1)$$

where X_t is the joint state at time t, $X_t = \{x_{i,t}, i = 1..M\}$, Z_t the measurement at time t. M is the number of targets.

The joint likelihood is denoted as $\phi(Z_t|X_t)$.

In addition, we consider that the propagation equation is:

$$p(X_t|X_{t-1}) = \prod_i p(x_{i,t}|x_{i,t-1}) \quad (2)$$

and that the dynamics is linear:

$$x_{i,t} = D x_{i,t-1} + v_{i,t} \quad (3)$$

where $v_{i,t}$ is a Gaussian state noise with zero mean and covariance matrix $V_{i,t}$. The matrix D relates the state at time t-1 to the state at time t.

As we explain below, we approximate the posterior by the product of independent probabilities. Then, the propagated density is the product of individual propagated densities:

$$p_{t|t-1}(X_t) = \int p(X_t|X_{t-1}) p(X_{t-1}|Z_{t-1}) = \prod_i p_{i,t|t-1}(x_{i,t}) \quad (4)$$

where $p_{i,t|t-1} = \int p(x_{i,t}|x_{i,t-1}) p(x_{i,t-1})$.

In tracking, the goal is to find the posterior $p(X_t|Z_t)$, which is usually an intractable function. Therefore, we look for another function $Q(X_t)$ to approximate it. $Q(X_t)$ is found by minimizing a cost function defined as [16, 4]:

$$J_t(Q) = KL\left(Q(X_t) \parallel p(X_t|Z_t)\right) - log\, p(Z_t) \quad (5)$$

where KL is the Kullback-Leibler divergence.

In a mean field framework the posterior is approximated by a product of independent probability densities [16, 4]:

$$p(X_t|Z_t) \approx Q_t(X_t) = \prod_i Q_{i,t}(x_{i,t}) \quad (6)$$

so that the cost function is rewritten as:

$$J_t(Q) = KL\left(\prod_i Q_{i,t}(x_{i,t}) \parallel p(X_t|Z_t)\right) - log\, p(Z_t) = \int \prod_i Q_{i,t}(x_{i,t}) log \frac{\prod_i Q_{i,t}(x_{i,t})}{p(X_t|Z_t)} dX_t - log\, p(Z_t) \quad (7)$$

Taking into account that $p(X_t|Z_t) = p(X_t, Z_t)/p(Z_t)$ and substituting $p(X_t, Z_t)$ using equations 1 and 4, we arrive to:

$$J_t(Q) = \sum_i \int_i Q_{i,t}(x_{i,t}) log\, Q_{i,t}(x_{i,t}) + \sum_i \int_i Q_{i,t}(x_{i,t}) log\, p_{i,t|t-1}(x_{i,t}) + + \int Q_t(X_t) log\, \phi(Z_t|X_t) \quad (8)$$

So far we have presented standard material. In the next section we present our contribution for managing joint likelihoods.

2.1. Complex likelihoods in mean field

An important point of our development is the likelihood function. We assume that we can obtain two different likelihoods:

- Firstly, a Gaussian approximation of the likelihood for each object, $\phi(\tilde{z}_{i,t}|x_{i,t})$, which assumes independent targets. Each one has a measurement vector $\tilde{z}_{i,t}$ related to the state by a linear relation:

$$\tilde{z}_{i,t} = Hx_{i,t} + w_{i,t} \tag{9}$$

  where $w_{i,t}$ is a Gaussian measurement noise with zero mean and covariance matrix denoted as $W_{i,t}$, and H is the matrix that relates state to measurement.

- Secondly, the 'true' likelihood, $\phi(Z_t|X_t)$, which considers a joint state and takes into account occlusions.

Thus, we can set the following equation:

$$\phi(Z_t|X_t) = \phi^{'}(Z_t|X_t) \prod_i \phi(\tilde{z}_{i,t}|x_{i,t}) \tag{10}$$

where $\phi^{'}(Z_t|X_t)$ measures the deviation of the likelihood from the Gaussian and independent case. In fact, equation 10 is a definition of $\phi^{'}(Z_t|X_t)$ from two known quantities. It will also allow us to write the final equations in a more elegant way. If we approximate $\phi^{'}(Z_t|X_t) \approx 1$, all the probability densities are Gaussians and therefore a Kalman solution for each object can be obtained. Let us refer to the mean of the Kalman solution as $\mu_{K,i,t}$ and to the covariance matrix as $\Sigma_{K,i,t}$.

Taking into account equation 10 in equation 8 we easily arrive to:

$$\begin{aligned} J_t(Q) = \sum_i \int_i Q_{i,t}(x_{i,t}) log\, Q_{i,t}(x_{i,t}) + \\ \sum_i \int_i Q_{i,t}(x_{i,t}) log(\phi(\tilde{z}_{i,t}|x_{i,t}) p_{i,t|t-1}(x_{i,t})) + \\ + \int Q_t(X_t) log\, \phi^{'}(Z_t|X_t) \end{aligned} \tag{11}$$

We go one step further by assuming that $Q_{i,t}(x_{i,t})$ is Gaussian, in short $\mathcal{N}_{i,t}$, with mean $\mu_{i,t}$ and covariance $\Sigma_{i,t}$. In addition, the product $\phi(\tilde{z}_{i,t}|x_{i,t})p_{i,t|t-1}(x_{i,t})$ is proportional to the Kalman posterior. In this case, some of the integrals can be done analytically [11], since the first one is minus the entropy and the second is, up to a constant, the integral of a Gaussian times the product of a quadratic form:

$$\begin{aligned} log(\phi(\tilde{z}_{i,t}|x_{i,t}) p_{i,t|t-1}(x_{i,t})) = constant + \\ -\frac{1}{2}(x_{i,t} - \mu_{K,i,t})^T \Sigma^{-1}_{K,i,t} (x_{i,t} - \mu_{K,i,t}) \end{aligned} \tag{12}$$

A very similar calculation can be found in [10] for interactions modeled as Markov Random Fields. The result is easily adapted to the present case:

$$\begin{aligned} J_t(Q) = constant - \frac{1}{2} \sum_i log|\Sigma_{i,t}| + \\ + \frac{1}{2} \sum_i Tr(\Sigma^{-1}_{K,i,t} \Sigma_{i,t}) + \\ + \frac{1}{2} \sum_{i,t} (\mu_{i,t} - \mu_{K,i,t})^T \Sigma^{-1}_{K,i,t} (\mu_{i,t} - \mu_{K,i,t}) + \\ - \int log\, \phi^{'}(Z_t|X_t) \prod_i \mathcal{N}_{i,t} \end{aligned} \tag{13}$$

Finding the means and the covariances by minimizing equation 13 is also very similar to the calculation done in [10]. Here we focus on an approximation of $log\, \phi^{'}(Z_t|X_t)$ by a linear term around the value at the mean state:

$$\begin{aligned} log\, \phi^{'}(Z_t|X_t) \approx log\, \phi^{'}(Z_t|\mu_t) + \\ + \sum_i \frac{\partial log\, \phi^{'}(Z_t|X_t)}{\partial x_{i,t}} (x_{i,t} - \mu_{i,t}) \end{aligned} \tag{14}$$

where $\mu_t = \{\mu_{i,t}, i = 1..M\}$. Under this approximation the integral in equation 13 is :

$$\int log\, \phi^{'}(Z_t|X_t) \prod_i \mathcal{N}_{i,t} \approx log\, \phi^{'}(Z_t|\mu_t) \tag{15}$$

because the integral of the linear term times the Gaussian is zero. Since equation 15 does not depend on $\Sigma_{i,t}$, the solution for the covariance is the same as in the Kalman case, which could also be verified by taking the derivative of equation 13 with respect to $\Sigma_{i,t}$. Therefore, the only relevant terms in equation 13 are:

$$\begin{aligned} \frac{1}{2} \sum_i (\mu_{i,t} - \mu_{K,i,t})^T \Sigma^{-1}_{K,i,t} (\mu_{i,t} - \mu_{K,i,t}) + \\ - log \phi^{'}(Z_t|\mu_t) \end{aligned} \tag{16}$$

The minimization of 16 with respect to $\mu_{i,t}$ leads to [11]:

$$\mu_{i,t} = \mu_{K,i,t} + \Sigma_{K,i,t} \frac{\partial log\, \phi^{'}(Z_t|\mu_t)}{\partial \mu_{i,t}} \tag{17}$$

The gradient is in fact found from equation 10 as [11]:

$$\frac{\partial log\,\phi^{'}(Z_t|X_t)}{\partial x_{i,t}} = \frac{\partial log\,\phi(Z_t|X_t)}{\partial x_{i,t}} - H^T W_{i,t}(\tilde{z}_{i,t} - Hx_{i,t}) \tag{18}$$

Equation 17 can be easily interpreted as a correction of the Kalman solution that comes from a likelihood that is not the product of independent Gaussians. Since the gradient is taken at $\mu_{i,t}$, this mean enters in both the left and the right side of equation 17. Thus equations 17 for $i = 1..M$ form a set of nonlinear equations that can be solved by any standard method. The simplest is using equation 17 as a fixed point equation, but this approach did not converge in our experiments. The convergence probably depends on each particular case. Thus, we transformed equation 17 using an analog of the 'chord method' [3] to get the equivalent equation:

$$\mu_{i,t} = \mu_{i,t} - \beta(\mu_{i,t} - (\mu_{K,i,t} + \Sigma_{K,i,t}\frac{\partial log\,\phi^{'}(Z_t|\mu_t)}{\partial \mu_{i,t}})) \tag{19}$$

where β is a constant. Setting it between 0.2 and 0.5 we obtained convergence when iterating the fixed point equation 19.

We refer to the resulting algorithm as Approximated Gaussian Sequential Mean Field (AGSMF), figure 1. We experimentally found that three of five iterations were enough to get convergence in agreement with [16].

Obtain mean and covariance matrix at time t, $\mu_{i,t}$, $\Sigma_{i,t}$ from mean and covariance at time t-1, $\mu_{i,t-1}$, $\Sigma_{i,t-1}$ $i = 1..M$, M=number of targets.
(1). Run a Kalman filter for each target to obtain $\mu_{K,i,t}$ and $\Sigma_{K,i,t}$, $i = 1..M$. Accept the covariances.
(2). Initialize $p \leftarrow 1$, $\mu_{i,t,p} = \mu_{K,i,t}$
(3). Iteration:
3.a. Obtain numerically the gradient $\frac{\partial log\,\phi(Z_t|X_t)}{\partial x_{i,t}}$, and then use equation 18 to obtain $\frac{\partial log\,\phi^{'}(Z_t|X_t)}{\partial x_{i,t}}$. The gradients have to be calculated at the point $X_t = \mu_{t,p} = (\mu_{1,t,p}, \mu_{2,t,p}, ..., \mu_{M,t,p})$.
3.b. Use equations 17 or 19 to obtain $\mu_{i,t,p+1}$
3.c. Repeat 3.a and 3.b for $i = 1..M$
3.d. Iteration: $p \leftarrow p + 1$, iterate until convergence
(4) Result: $\mu_{i,t} \leftarrow \mu_{i,t,p}$, $\Sigma_{i,t} \leftarrow \Sigma_{K,i,t}$

Figure 1. AGSMF Approximated Gaussian Sequential Mean Field.

Figure 2. Synthetic sequence. Left: input image; Right: segmented image.

3. Experiments

3.1. Synthetic sequence

We made a synthetic sequence of 1000 images where five tennis balls are bouncing on the limits of a background image of 352x288 pixels. The speed was limited to an absolute value of 10 pixels per frame and a noise was added to the velocity at each frame. The radius of the ball was 14 pixels.

The ball colour has been modeled in RGB space by a Gaussian, which allows obtaining a binary image using a threshold of the Mahalanobis distance in the input image, see figure 2. Therefore we have an experimental classification of whether a pixel belongs to a tennis ball or to the background.

The state of each tennis ball is a four dimensional vector $(y, x, \dot{y}, \dot{x})$ and we used a constant velocity motion model with diagonal noise covariance. In this experiment, we assumed a given depth order of the balls since we had no other cues. When a ball occludes another one, there is no colour model change in the occluded pixels since all the objects have a single colour model and the same form. However, the proposed algorithm can still improve tracking by means of a joint measurement process. It corrects the computationally faster independent measurement processes that we show below.

3.1.1 Independent object likelihood

To obtain a vector measurement $\tilde{z}_{i,t}$ we first calculate the predicted position of the ball, $\mu_{i,t|t-1}$. Then, we set a grid around it. We can associate a weight $w(y, x)$ with each point (y, x) of the grid:

$$log\,w(y, x) = \alpha \sum_{u \in ball} p_u \tag{20}$$

where the sum extends over the pixels u inside a ball centered at (y, x), and p_u is $+1$ if the pixel is experimentally classified as a ball or -1 otherwise.

Afterwards, the weights of all the grid points are normalized. α is a constant that is set to have a suitable covariance

Table 1. Results on the synthetic sequence

Algorithm	Tracker coalescence	Label interchange	Total	Time per frame (s)
IKF	9	1	10	0.101
AGSMF	1	0	1	0.104

of the weights. Once the weights are normalized, we can find the mean position and a covariance matrix. We use the mean as the measurement $\tilde{z}_{i,t}$, and the covariance matrix as the covariance of the measurement noise, $W_{i,t}$ in equation 9.

3.1.2 Joint likelihood

For $\phi(Z_t|X_t)$ we adopt this definition that is also based in the segmentation of the image:

$$log\,\phi(Z_t|X_t) = \alpha \sum_u q_u \tag{21}$$

where q_u is $+1$ if the expected classification of a pixel agrees with the measured classification and -1 otherwise. A pixel is expected to belong to a ball if it is inside of any of the circles centered at the tennis ball positions. The reader should be aware that equation 21 depends on the joint state X_t while equation 20 depends on only one object state $x_{i,t}$. We also point out that, in fact, we need the gradient of $log\,\phi(Z_t|X_t)$ with respect to an object state $x_{i,t}$ and that this is evaluated through a difference $log\,\phi(Z_t|X_t+\delta x_{i,t}) - log\,\phi(Z_t|X_t)$. Thus, for practical purposes the sum only needs to extend over those pixels that have changed their expected classification when comparing two nearby states.

3.1.3 Results

We run our algorithm with this synthetic sequence. Whenever a ball was more than 40 pixels away from its ground truth position, an error was raised and the tracking of all the objects resumed. We also run our algorithm without correction, using only the Kalman solution. The algorithms were coded combining Scipy [5] and C. We used C for the most time consuming operations that are related to the measurement process. The results are shown in table 1, where IKF means Independent Kalman Filters.

It is clear from that table that the mean field correction allows decreasing the number of errors. The increase of the computation time corresponds to the correction of the Kalman solution, which needs a few ms in C. With an optimized code in C we could easily achieve a higher rate of frames per second.

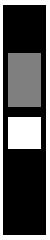

Figure 3. Mask attached to a 'Fullham' player. Each gray level is one of three colour models: background, t-shirt or shorts.

3.2. Football sequence

This second experiment was performed on 700 frames of a football sequence of the VS-PETS2003 data base [1]. Specifically we used 300 frames from the directory TRAINING/CAMERA3 and 400 from TESTING/CAMERA3. When a new player entered (exited) the scene, we artificially added (removed) an object to our tracker since we have not implemented a detection algorithm. This experiment is more complex because a single Gaussian is not enough to model the colour. In addition, there are two types of objects, 'Fullham' and 'Liverpool' players.

The state of each player is given by a four dimensional vector $(y, x, \dot{y}, \dot{x})$, where the images coordinates are the coordinates of the player's feet. The y-axis points downwards. In the view of the pitch we used, it is reasonable to assume that object A occludes object B if $y_A > y_B$ and if some pixels of the two objects overlap. The physical height of the player is assumed to be the same for all the players and the height in the image is adjusted depending on the y coordinate using an experimental fitted function.

We fitted a Gaussian in RGB space to four colours in the image: black (b), white (w), red (r) and green (g). Attached to each player position there is a mask in a bounding box that encodes the colour model of each pixel, which we allowed to be one of the following options: background (-1), 'Fullham' player t-shirt (0) , 'Fullham' player shorts (1), and 'Liverpool' player(2). Background is supposed to be also outside the bounding box. The size of the mask is proportional to the player height in the image. There is no one to one mapping between colour Gaussians and colour models. For instance, a 'Fullham' player t-shirt is mainly white, bus has black numbers in the back. A representation of the 'Fullham' player mask is shown in figure 3. This is a rather crude approximation of the player shape.

We adopt a likelihood similar to that used in [15], adapting it to colour appearance and to mixtures of colours. Thus, we used the following pixel likelihood $log\phi_u(c|m)$ depending on the colour model of the pixel (c is the colour of pixel

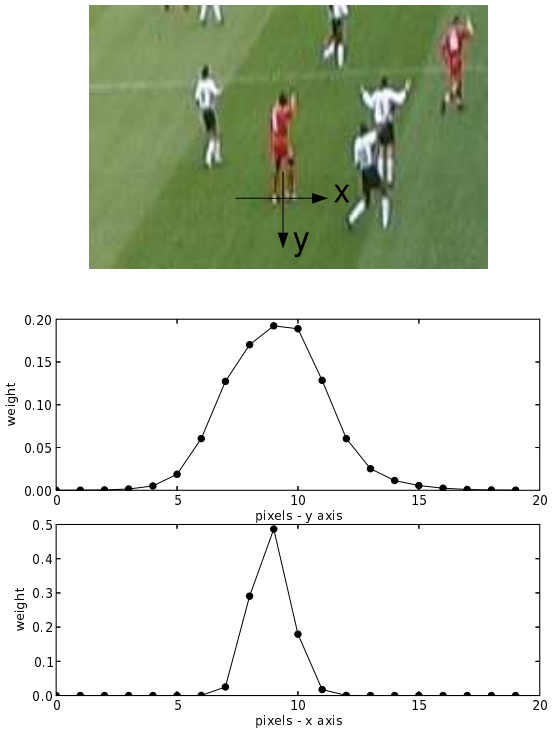

Figure 4. Weights around the true ground position of a red player.

u and m the model):

$$-log\phi_u(c|m) = \gamma_{m,g} d_g^2(c) + \gamma_{m,w} d_w^2(c) + \\ +\gamma_{m,r} d_r^2(c) + \gamma_{m,b} d_b^2(c) \qquad (22)$$

where u is a pixel, $d_n(c)$ is the Mahalanobis distance of pixel colour value c from colour Gaussian $n = red(r), white(w), black(b)$ or $green(g)$, and $\gamma_{m,n}$ are positive multiplicative constants for model $m = -1, 0, 1, 2$ and Gaussian $n = red(r), white(w), black(b)$ or $green(g)$. For instance, 'Fullham' player shorts are mainly black, so $\gamma_{1,n} = 0$ except for $n = black(b)$, while the other models include some mixture. The proportionality constants $\gamma_{m,n}$ were estimated subjectively. With the selected values, the above equation works reasonably well and allow any tracking algorithm to focus on the players, giving also a reasonably spread of the likelihood. In figure 4 we show the normalized likelihood obtained by scanning in the y and x axis from the position of a 'Liverpool' player. The height of the player is about 50 pixels.

3.2.1 Independent object likelihood

Given the above colour pixel likelihoods, we can obtain a measurement vector $\tilde{z}_{i,t}$ as follows. We first set a grid around the predicted position of a given player, $(y_{i,t|t-1}, x_{i,t|t-1})$. Then we loop over the grid points, (y, x), place the player mask at that point and then compute a weight as:

$$log\, w(y,x) = \sum_u \log \phi_u \qquad (23)$$

where in principle the sum extends over all the image pixels. ϕ_u is given by equation 22. Weights are then normalized. After this step, we can calculate a mean position and a covariance, which we take as the measurement $\tilde{z}_{i,t}$ and measurement noise $W_{i,t}$ respectively. Please note that in this process a single object is assumed in the image.

3.2.2 Joint likelihood

We need also the joint likelihood $\phi(Z_t|X_t)$. To calculate it we first obtain the colour model of each pixel by decomposing the joint state vector X_t into object components $x_{i,t}$, ordering them according to camera distance, and computing the colour model of all the pixels from the mask associated with each player. In this case occlusions are taken into account, that is, if object B can occlude object A, i.e. $y_B > y_A$, then non-background colour models of the mask(B) override colour models of mask(A) for a given pixel. In this way, we can use an equation similar to 23.

$$log\phi(Z_t|X_t) = \sum_u \log \phi_u \qquad (24)$$

where ϕ_u is given by equation 22.

For practical purposes, the sum in equations 22 or 23 does not have to extend over all the image pixels. The reason is the following. In equation 23 we intend to obtain weights by comparing nearby positions in a grid. The weights are then normalized. In addition, what we need is not exactly $log\, \phi(Z_t|X_t)$, but its gradient, see equation 18, which is computed numerically as a difference. In any case, we have to compare the likelihood for a state X_t and for a nearby state $X_t + \delta X_t$. Therefore the pixels that have the same colour model for X_t and for $X_t + \delta X_t$ does not contribute to the difference.

3.2.3 Results

Table 2 shows the results in terms of tracking failures. After a failure was observed, we manually set the player at the right position and resume the algorithm. Figure 5 shows a failure with IKF when a 'Liverpool' player jumps and is occluded by a 'Fullham' player. This failure is avoided if we run AGSMF algorithm as shown in figure 6. Figures 7

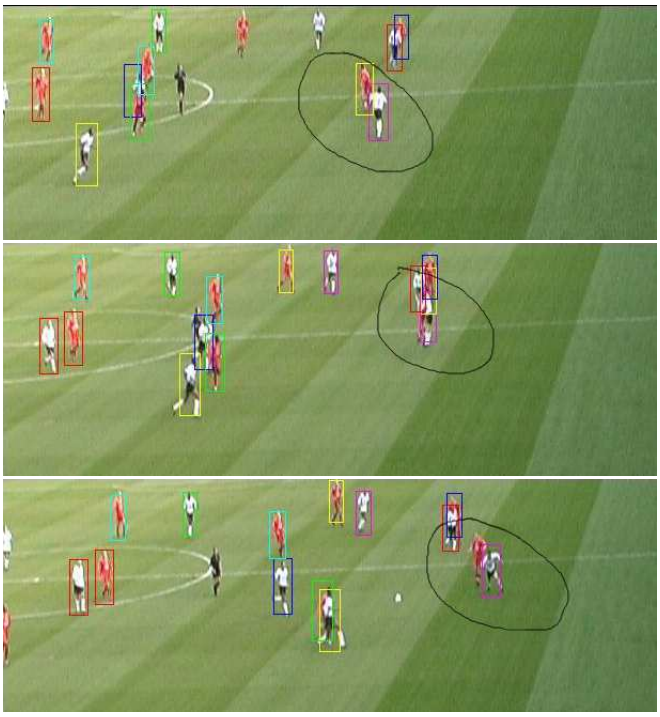

Figure 5. Key frames of the football sequence with IKF algorithm.

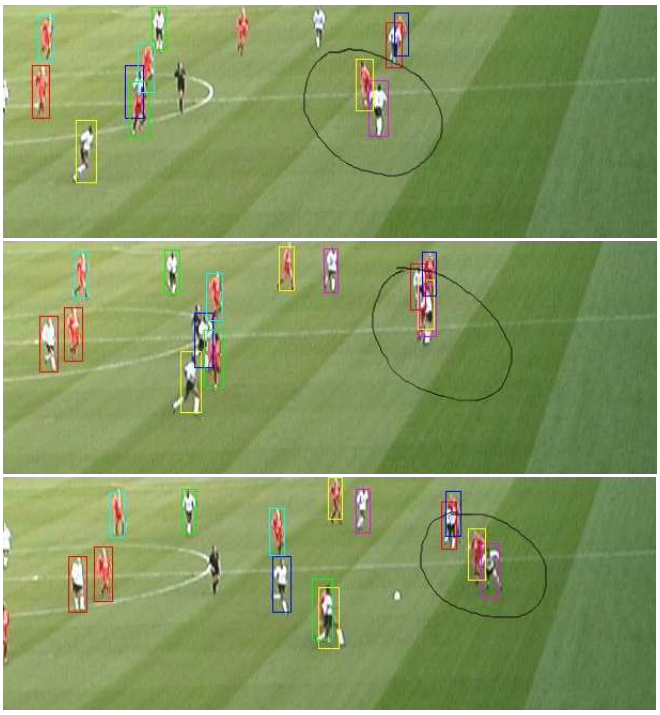

Figure 6. Key frames of the football sequence with AGSMF algorithm.

Table 2. Results on the football sequence

Algorithm	Tracker coalescence	Tracker Lost	Total
IKF	4	1	5
AGSMF	1	0	1

Table 3. Pixel distance between ground truth position and tracked position for an occluded player

Tracker	Mean	Stdev	Max
IKF	5.9	4.1	14.8
AGSMF	3.6	2.0	8.0

and 8 show also a failure that is avoided by AGSMF, in this case for an occlusion of two 'Liverpool' players.

The failure of AGSMF reported in table 2 is caused by a player who is completely occluded for some frames and has a nearby team mate, who 'attracts' the tracker of the occluded player. In this case, dynamical information is not enough to split the two trackers. This could be avoided by including some interaction between the players with an MRF.

In this sequence, the object independent trackers do not give many errors. However, we have also checked the accuracy of both methods, IKF and AGSMF. We have compared tracked position with ground truth position for a player that is occluded for about 70 frames. Even though player labels are correct in these frames, AGSMF gives better accuracy as shown in table 3.

4. Conclusions

In this paper we have proposed a multi-target tracking algorithm with a principled occlusion reasoning in a mean field approach. Assuming that the object posteriors can be approximated by Gaussians, we have obtained an algorithm that corrects iteratively an initial solution based on independent Kalman filters. The algorithm has been tested on a synthetic sequence and on a standard benchmark sequence of a football match. The results show that it reduces the number of tracking failures even though the approach is formally simple.

In future, we plan to add multimodality to the model and to include interaction with pairwise potentials (MRFs).

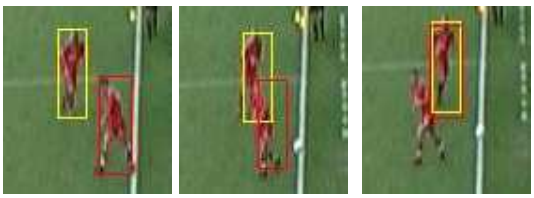

Figure 7. Key frames of the football sequence with IKF algorithm.

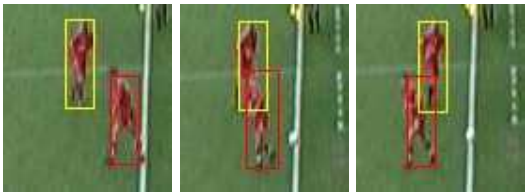

Figure 8. Key frames of the football sequence with AGSMF algorithm.

Acknowledgments

This work is partially supported by the Spanish grant TIN2006-11044 (MEC) and FEDER. J. Martínez is supported by a FPI grant BES-2004-3741 from the Spanish Ministry of Education. R. Igual is supported by a grant of the 'Fundación Antonio Gargallo', 2007/B003.

References

[1] VS-PETS2003 football data base, 2003. available at http://www.cvg.cs.rdg.ac.uk/datasets/index.html, last visit on March 2008.

[2] Y. Bar-Shalom and T. Forman. *Tracking and Data Association*. Academic Press, 1988.

[3] E. Isaacson and H. B. Keller. *Analysis of Numerical Methods*, chapter Iterative solutions of non-linear equations. John Wiley and Sons, 1966.

[4] T. Jaakkola. *Advanced Mean Field Methods: Theory and Practice*, chapter Tutorial on variational approximation methods. MIT Press, 2001.

[5] E. Jones, T. Oliphant, P. Peterson, et al. Scipy:open source scientific tools for python, 2001. http://www.scipy.org/, last visit on July 2007.

[6] Z. Khan, T. Balch, and F. Dellaert. MCMC-Based Particle Filtering For Tracking a Variable Number of Interacting Targets. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 27(11):1805–1819, 2005.

[7] O. Lanz. Approximate Bayesian Multibody Tracking. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 28(9):1436–1449, 2006.

[8] J. MacCormick and A. Blake. A Probabilistic Exclusion Principle for Tracking Multiple Objects. *International Journal of Computer Vision*, 39(1):57–71, 2000.

[9] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler. Tracking Groups of People. *Computer Vision and Image Understanding*, 80(1):42–56, 2000.

[10] C. Medrano, J. Elías, J. Martínez, and C. Orrite. Mean field approach for tracking similar objects. *Computer Vision and Image Understanding*, 2008. Submitted for publication.

[11] K. Petersen and M. Pedersen. The matrix cookbook, 2006. available at http://matrixcookbook.com/, last visit on May 2008.

[12] C. Rasmussen and G. D. Hager. Probabilistic Data Association Methods For Tracking Complex Visual Objects. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 23(6):560–576, 2001.

[13] L. Sigal and S. M. Black. Measure Locally, Reason Globally: Occlusion-Sensistive Articulated Pose Estimation. In *Proceeding of the IEEE Conference On Computer Vision and Pattern Recognition*, volume 2, pages 2041–2048, 2006.

[14] E. B. Sudderth, M. I. Mandel, W. Freeman, and A. S. Willsky. Distributed Occlusion Reasoning For Tracking With Nonparametric Belief Propagation. In *Neural Information Processing Systems, NIPS 2004*, 2004.

[15] Y. Wu, T. Yu, and G. Hua. Tracking Appearances With Occlusions. In *Proceeding of the IEEE Conference On Computer Vision and Pattern Recognition*, volume 1, pages 789–795, 2003.

[16] T. Yu and Y. Wu. Collaborative Tracking of Multiple Targets. In *Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition*, number 1, pages 834–841, 2004.