Towards fast and efficient methods for tracking players in sports

Matej Kristan, Janez Perš, Matej Perše, and Stanislav Kovačič

Faculty of Electrical Engineering, University of Ljubljana, Slovenia
{matej.kristan, janez.pers, matej.perse, stanislav.kovacic}@fe.uni-lj.si
http://vision.fe.uni-lj.si

Abstract. An efficient algorithm for tracking a single player in a sporting match is presented in this paper. The sporting event is considered as a semi-controlled environment for which a set of closed-world assumptions regarding the visual as well as dynamical properties is derived. We show how these assumptions can be used in the context of particle filtering to arrive at a computationally-fast and reliable tracker. The proposed tracker was evaluated on a demanding data set. When compared to several similar trackers that did not utilize all of the closed-world assumptions, the proposed tracker, on average, resulted in a better performance regarding the failure rate as well as position and prediction estimation.

1 Introduction

Tracking players in a sporting match and obtaining their trajectories on the playground offers useful insight into properties that make a certain player/team a winner or a loser, which is why the interest of sport experts in the computer-aided analysis is increasing. When designing a tracker capable of tracking players in multiple-player sports, usually two important features need to be addressed. One is the concept for multiple-target management, of which some interesting examples with application to football and hockey can be found in [1,2,3,4] and [5], respectively. However, if an elaborate multiple-target tracker is based on an inefficient single-target tracker, the overall performance can be rather poor. Thus the second important feature is the method to maintain a successful track of a particular player. The latter is the focus of this paper.

We argue that by considering a sporting match as a semi-controlled environment, certain assumptions can be made, which may lead to an improved tracking performance. Using these assumptions, an algorithm based on a well-known particle filter is designed for tracking a single player in a sporting match. The strength of the proposed tracker is demonstrated on a demanding data set.

The remainder of the paper is organized as follows. Section 3 introduces a sporting event as a semi-controlled environment and in Section 2 the engine of the tracker is presented. Sections 4 and 5 are concerned with the visual and dynamical properties of the tracked player. In Section 6 a scheme for learning player's motion is presented and in the Section 7 the results of the experiments are described and discussed. Finally, Section 8 concludes the paper.

2 Particle filtering

In recent years, particle filters have been shown to provide efficient means of visual tracking in various situations. Since their first appearance in the vision community [6], they soon gained on their popularity by proving to be robust and having the ability to handle the uncertainties usually present in visual data. For the same reasons we use a particle filter as the engine of our tracker. We provide here only the basic concept and notations, and refer the interested reader to [6,7] for more details.

Let $\mathbf{x}_{t-1}$ denote the state of a tracked object at time-step $t-1$, let $\mathbf{y}_{t-1}$ be an observation at $t-1$, and let $\mathbf{y}_{1:t-1}$ denote a set of all observations up to $t-1$. From a Bayesian point of view, all of the interesting information about the target's state $\mathbf{x}_{t-1}$ is embodied by its posterior $p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})$. The aim of tracking is then to recursively estimate this posterior as the new observations $\mathbf{y}_t$ arrive. This process is characterized by two steps: prediction (1) and update (2).

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})d\mathbf{x}_{t-1} \tag{1}$$

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) \tag{2}$$

In our implementation, we use the well-known CONDENSATION algorithm [6], which is a simple particle filter, where the posterior at time-step $t-1$ is represented by a finite set of weighted particles and where the recursions (1) and (2) are carried out via Monte Carlo simulations. This approach requires only a specification of the dynamical model describing the state evolution $p(\mathbf{x}_t|\mathbf{x}_{t-1})$, and a model that evaluates the likelihood of any state given the observation $p(\mathbf{y}_t|\mathbf{x}_t)$.

3 The closed world

Treating a sporting event as a semi-controlled environment is a concept previously introduced by Intille and Bobick [8] under the name *closed world* (CW). The main premise is that for a given region in space and time, a specific context is adequate to explain that region. In our case, the context is a set of the following CW assumptions: *(CW1) The camera overlooking the playground is static. (CW2) The playground is bounded, and its model can be calculated. (CW3) The illumination is nonuniform in space and time. (CW4) The players' textures vary during the game, and are known in the beginning. (CW5) The player cannot change his/her position completely arbitrarily due to the effects of inertia.*

Based on these assumptions, a sports-domain-specific tracker will be presented in the following sections.

4 The visual cues

In our implementation, players are considered as an elliptical regions as in [9], since an ellipse provides a low-dimensional representation and is still robust enough to capture different appearances of a player. A state of the tracked player $\mathbf{x}_t \in \mathcal{X}$, with $\mathcal{X}$ denoting the state space, is thus parameterized by an ellipse $\mathbf{x}_t = (x_t, y_t, a_t, b_t)$ with a center at (x_t, y_t), and with parameters a_t and b_t denoting the width and the height, respectively.

Let $E = (x, y, a, b)$ be an elliptical region at some state $\mathbf{x}_t = (x, y, a, b)$. Following [9], the color model of the state $\mathbf{x}_t$ is encoded by the RGB color histogram $\mathbf{h}_\mathbf{x} = \{h_i\}_{i=1}^{512}$ with eight bins per color channel, sampled within the ellipse E. The strategy of building the histogram follows that of [9], with the extension of using a mask function $M(\mathbf{u})$, meaning that only those pixels $\mathbf{u}$ within E that are assigned to the foreground by the $M(\mathbf{u})$ are considered.

According to the CW2, the background image can be modelled, thus we define the measure which evaluates whether a player with a predefined reference histogram $\hat{\mathbf{h}}_t$ is present at some state $\mathbf{x}_t$ as

$$D(\mathbf{h}_A, \hat{\mathbf{h}}_t; \mathbf{h}_B) = \beta^{-1} D_n(\mathbf{h}_A, \hat{\mathbf{h}}_t; \mathbf{h}_B), \tag{3}$$

where $\mathbf{h}_A$ and $\mathbf{h}_B$ are histograms sampled at the state $\mathbf{x}_t$ on the current and the precalculated background image, respectively, β is the portion of pixels within the elliptical region of $\mathbf{x}_t$, that are assigned to the foreground by the mask function $M(\mathbf{u})$, and $D_n(\mathbf{h}_A, \hat{\mathbf{h}}_t; \mathbf{h}_B)$ is the normalized distance between $\mathbf{h}_A$ and $\hat{\mathbf{h}}_t$ given the background histogram $\mathbf{h}_B$, defined as

$$D_n(\mathbf{h}_A, \hat{\mathbf{h}}_t; \mathbf{h}_B) = \frac{\varrho(\mathbf{h}_A, \hat{\mathbf{h}}_t)}{\sqrt{\varrho(\mathbf{h}_B, \hat{\mathbf{h}}_t)^2 + \varrho(\mathbf{h}_A, \hat{\mathbf{h}}_t)^2}}, \tag{4}$$

where $\varrho(\mathbf{h}_1, \hat{\mathbf{h}}_2) = 1 - \sum_i \sqrt{\mathbf{h}_{1i}\mathbf{h}_{2i}}$ is the Hellinger distance [10].

In order to carry out the update step (2) of the particle filter, the analytical form of the probability density function of (3) needs to be known. For this reason, a number of players were tracked using a reference tracker [9] and by manual marking. This enabled us to obtain approximately 115 thousand values of (3), typical for the players during the match; these are depicted by the histogram in Fig. 1. To identify the best model for the gathered data, a model selection was carried out using Akaike information criterion (AIC) [11] among four test models of density functions: exponential, gamma, inverse gamma and zero-mean gaussian. The test with the AIC has shown, that the gamma function explained the data significantly better than the other functions, and for this reason the probability density function of (3) was chosen in the form of

$$p(\mathbf{y}_t|\mathbf{x}_t) \propto D(\mathbf{h}_A, \hat{\mathbf{h}}_t; \mathbf{h}_B)^{\gamma_1 - 1} e^{\frac{-D(\mathbf{h}_A, \hat{\mathbf{h}}_t; \mathbf{h}_B)}{\gamma_2}}, \tag{5}$$

with the parameters $\hat{\gamma}_1 = 1.769$ and $\hat{\gamma}_2 = 0.066$, which were the maximum likelihood estimates of the parameters γ_1 and γ_2 calculated from the data.

4.1 The mask function

In addition to histograms, we employ a mask function $M(\mathbf{u})$, which is calculated simply by thresholding the differences between the current and the background image using some threshold κ_t. Since the illumination of the playground

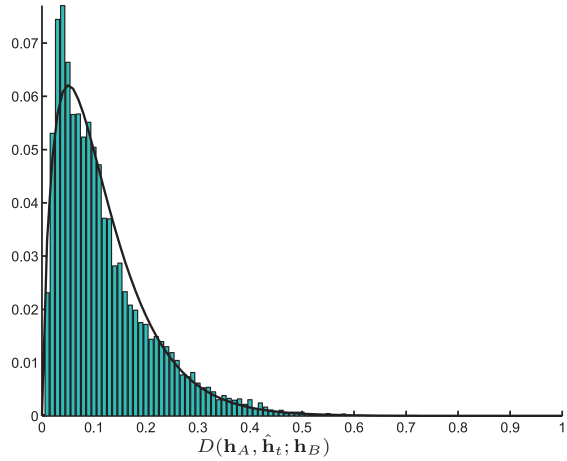


Fig. 1. The figure shows the empirical probability density function of the measure (3) in form of the histogram and overlaid is the maximum-likelihood fitted gamma distribution

is nonuniform in space and time (CW3) and since the visual properties among players vary, the threshold has to be estimated dynamically for a specific player: Let $\tilde{\mathbf{x}}_t$ denote the estimated state of a particular player at time-step t and let $\mathbf{h}_A$ and $\mathbf{h}_B$ be the histograms sampled at that state on the current and the background image, respectively. If the condition $\varrho(\mathbf{h}_A, \hat{\mathbf{h}}_B) < \varrho_{thresh}$ is fulfilled, then the similarity between the player's visual model and the background is significant and the threshold κ_{t+1} for the next time-step is estimated as the threshold, which in the current time-step produces such a mask function that at least some predefined percentage η_0 of the pixels in the current image that lie within the ellipse of the state $\tilde{\mathbf{x}}_t$ are assigned to the background. Otherwise, the mask function is not generated for that player in the next time step.

The parameters η_0 and ϱ_{thresh} were estimated empirically by manually selecting players on heavily cluttered parts of the playground and were set to $\eta_0 = 25\%$ and $\varrho_{thresh} = 0.8$, respectively.

4.2 Adaptation of the visual model

According to the CW4, the player's texture varies during a match, therefore the color model, i.e. the player's current reference histogram $\hat{\mathbf{h}}_t$, has to be able to adapt to these changes. Let $\tilde{\mathbf{x}}_t$ denote the estimated state of a player at the current time-step and let $\mathbf{h}_A$ and $\mathbf{h}_B$ be the histograms sampled at that state on the current and the background image, respectively. The adaptation equation then follows the form of

$$\hat{\mathbf{h}}_t = \alpha_t \mathbf{h}_A + (1 - \alpha_t)\hat{\mathbf{h}}_t^-, \qquad (6)$$

where the superscript in $\hat{\mathbf{h}}_t^-$ denotes the reference histogram prior to adaptation. The intensity of the adaptation is defined with respect to the normalized distance between $\mathbf{h}_A$ and $\hat{\mathbf{h}}_t^-$ as

$$\alpha_t = \Omega_{max} \cdot (1.0 - D_n(\mathbf{h}_A, \hat{\mathbf{h}}_t^-; \mathbf{h}_B)), \tag{7}$$

where Ω_{max} denotes the maximal adaptation. Again, this parameter was estimated by means of a controlled experiment, and was set to $\Omega_{max} = 0.05$.

5 Dynamic modelling

Most of the time during the game, the player's aim is to act in an unpredictable fashion in order to confuse the opponent. This implies that the dynamics might be modelled by a random-walk model [12] as $p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \mathbf{x}_{t-1}, \Lambda_t)$, where $\mathcal{N}(\cdot; \cdot, \cdot)$ denotes the normal distribution with mean $\mathbf{x}_{t-1}$ and a diagonal covariance matrix $\Lambda_t = diag(\sigma_{xy}^2, \sigma_{xy}^2, \sigma_{ab}^2, \sigma_{ab}^2)$.

Note, that the variances in Λ_t determine the amount by which the player's state is expected to change between consecutive time-steps and depend on the size of the player in the current image. We account for this dependence by writing

$$\sigma_{xy} = H_{t-1} \cdot \alpha_{xy} \ , \sigma_{ab} = H_{t-1} \cdot \alpha_{ab}, \tag{8}$$

where $H_{t-1} = \sqrt{a_{t-1}^2 + b_{t-1}^2}$ is a measure of size of the elipse from the previous state $\mathbf{x}_{t-1}$. The equations in (8) require some sensible estimates for the α_{xy} and α_{ab}; we derive these next.

Based on the findings in [13,14] regarding the dynamics of handball and basketball players during a match, we estimated the highest velocity of a player as $v_{max} = 8.0$m/s, or, at a frame rate of 25frames/s, as $v_{max} = 0.32$m/frames. During the tracking, the player is usually determined by an ellipse approximately the size of his/her shoulders which is estimated to be $H_t \approx 0.4$m. Assuming the Gaussian form on the velocity distribution, the highest velocity can be approximated with three standard deviations as $v_{max} = 3\sigma_{xy}$/frame, and the parameter for σ_{xy} in (8) is then $\alpha_{xy} = \frac{0.8}{3} \doteq \frac{1}{4}$. Thus the expected change of the ellipse center within two consecutive time steps is approximately one fourth of the ellipse's size.

The changes in the shape of the player's ellipse occur mainly due to tilting of the player, and thus we can assume that within two consecutive time-steps the size along each axis can change at most by 15 percent. Following similar line of thought as above, the parameter for σ_{ab} can be estimated as $\alpha_{ab} = 0.05$.

While it is true that the intention of a player is to move in a way to appear unpredictable to the opponent, the motion itself, however, is not entirely unpredictable since it is constrained by the player's *task* and the *physical limitations*, which enforce a sort of inertia on the motion (CW5). We therefore define the state-evolution model to be

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \mathbf{x}_{t-1} + \mathbf{d}_t, \Lambda_t), \tag{9}$$

where $\mathbf{d}_t$ is a time-step-constant drift modelling the influence of the inertia. The method for estimating the drift is described in the next section.

6 Local smoothing

In order to satisfactory estimate the drift $\mathbf{d}_t$ at time-step t, a reliable estimation of the past few states is needed. Since we are using a particle filter to recursively estimate the posterior of the target in time, the variance of the estimated state will usually depend on the number of particles used and the strategy by which the particles are propagated. For example, in order to cope with the sudden changes in motion, the common strategy is to increase the variance of the noise in the dynamical model. This, however, results in many particles having low values and contribute very little to the final estimation of the current state. The logical solution is then to increase the number of particles and/or use a clever strategy to concentrate particles in regions with high probability. Such strategies might be application of an auxiliary-variable particle filter [7], or perhaps methods of local likelihood sampling [15] to name just two. Even though each of the above methods are likely to result in an efficient tracking, they introduce an additional computational complexity which slows down the tracking. We propose here an alternative approach, where at each time-step the current state estimated from the particle filter is smoothed according to a locally-in-time learned dynamical model, which assumes that the player is not likely to change his/her velocity abruptly.

Let $\mathbf{o}_{t-T:t-1} = \{\mathbf{o}^{(k)}\}_{k=t-T}^{t-1}$ denote a sequence of the T past smoothed states of the tracked target, let $\pi_{t-T:t-1} = \{\pi^{(k)}\}_{k=t-T}^{t-1}$ denote the set of their weights and let $\mathbf{v}^{(k)} = \mathbf{o}^{(k)} - \mathbf{o}^{(k-1)}$ denote the shift between two successive smoothed states. We define a local shift distribution based on the past smoothed states as

$$p(\mathbf{v}|\mathbf{o}_{t-T:t-1}) = \sum_{k=t-T}^{t-1} \delta(\mathbf{v}^{(k)} - \mathbf{v}) G^{(k)}(t), \tag{10}$$

where $\delta(\cdot)$ is the dirac-delta function and where the weights $G^{(k)}(t)$ are defined as

$$G^{(k)}(t) = c_0 \pi^{(k)} \pi^{(k-1)} e^{-\frac{1}{2}\frac{(k-t+1)^2}{\sigma_o^2}}. \tag{11}$$

The first term c_0 in the above equation is the normalizing constant ensuring that $\sum_{k=t-T}^{t-1} G^{(k)}(t) = 1$, the second and the third terms reflect the likelihood of the states $\mathbf{o}^{(k)}$ and $\mathbf{o}^{(k-1)}$, respectively, and the last term is a Gaussian that assigns higher a priori weights to the more recent shifts.

The current drift $\mathbf{d}_t$ is then estimated as the expected value over the local shift distribution

$$\mathbf{d}_t = \langle \mathbf{v} \rangle_{p(\mathbf{v}|\mathbf{o}_{t-T:t-1})}, \tag{12}$$

where $\langle \cdot \rangle_{p(\mathbf{v}|\mathbf{o}_{t-T:t-1})}$ denotes the expectation operator over $p(\mathbf{v}|\mathbf{o}_{t-T:t-1})$.

The number of the smoothed states used in (10) is for practical applications set to $T = 3\sigma_o$, since the a priori weights of all older states are negligible. Assuming that a player can not radically change his/her velocity within one half of a second, a value for the parameter σ_o is chosen to comply with this time frame. Since all our test sequences are recorded at a frame rate of 25fps, we have chosen this parameter to be $\sigma_o = 4.3$, which means, that in our application only $T = 13$ past smoothed states are considered.

The smoothed state is calculated as follows. At time-step t, when the approximation to $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ becomes available from the particle filter, the mean square estimate (MSE) $\hat{\mathbf{x}}_t$ of the state is calculated from this distribution and fused with the prediction on the smoothed states $\tilde{\mathbf{o}}_t = \mathbf{o}^{(t-1)} + \mathbf{d}_t$ according to their likelihoods $w_{\hat{\mathbf{x}}_t} = p(\mathbf{y}_t|\hat{\mathbf{x}}_t)$ and $w_{\tilde{\mathbf{o}}_t} = p(\mathbf{y}_t|\tilde{\mathbf{o}}_t)$, respectively, as

$$\mathbf{o}^{(t)} = \frac{\tilde{\mathbf{o}}_t \cdot w_{\tilde{\mathbf{o}}_t} + \hat{\mathbf{x}}_t \cdot w_{\hat{\mathbf{x}}_t}}{w_{\tilde{\mathbf{o}}_t} + w_{\hat{\mathbf{x}}_t}}. \tag{13}$$

The corresponding weight of the new smoothed state $\mathbf{o}^{(t)}$ is then evaluated by the likelihood function $\pi^{(t)} = p(\mathbf{y}_t|\mathbf{o}^{(t)})$.

The proposed closed-world single-player tracker with local smoothing is summerized in the Algorithm 1. An example of tracking a player in a squash match is shown in Fig. 2, where the evolution of the local shift distribution with respect to the player's movement is illustrated. The first image (Fig. 2a) shows the player standing still, with the samples of the local shift distribution spreading around the his center indicating no preferable direction. As the player begins to move towards the center of the playground (Fig. 2b), the samples gather around the direction of the travel. In Fig. 2c, when the player suddenly stops, the samples spread around his center and as he begins to move towards the upper right corner (Fig. 2b), the samples again gather in that direction.

Algorithm 1 *The proposed closed-world tracking algorithm*

```
– Calculate the background image, e.g. pixel-wise by means of a median filter along
  the temporal axis.
– Initialize the tracker by selecting the player. (e.g. manually)
– For t = 1, 2, 3 ... do:
    1. Construct the mask function $M_t(\mathbf{u})$ according to the Section 4.1.
    2. Calculate the drift $\mathbf{d}_t$ by (12).
    3. Run the conventional CONDENSATION iteration (Section 2) using the dy-
       namical model from (9).
    4. Calculate the new smoothed state $\mathbf{o}^{(t)}$ following (13).
    5. Sample the histogram at state $\mathbf{o}^{(t)}$ and adapt the reference histogram to
       that histogram as described in the Section 4.2.
    6. If needed, estimate the threshold for the mask function in the next time-step
       (Section 4.1).
```

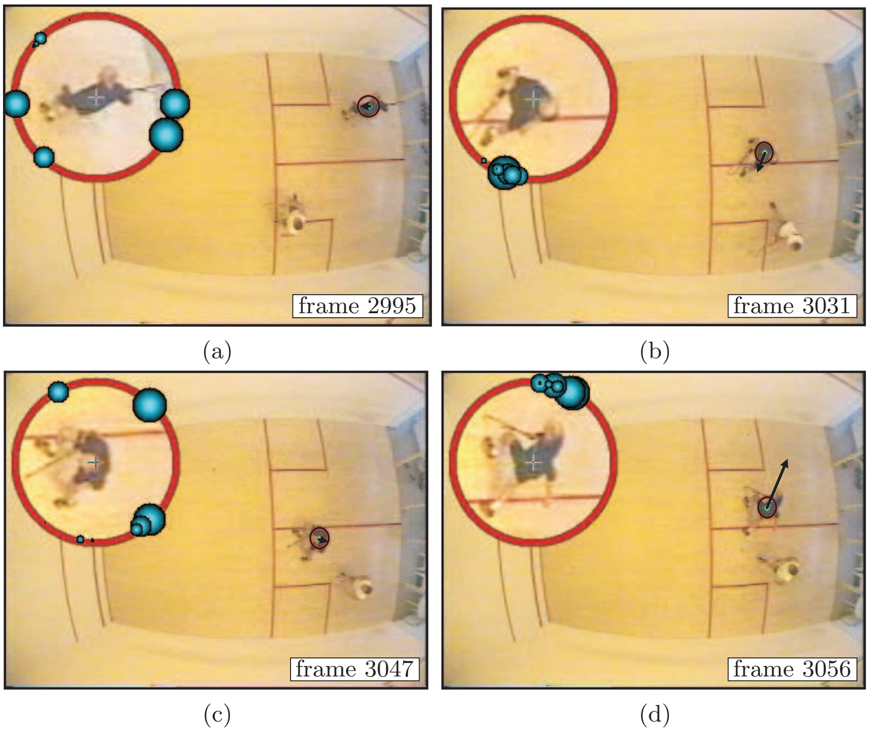


Fig. 2. The figures (a-d) show the tracked player during a squash match. The current smoothed state is depicted by an ellipse superimposed on the player and the arrow indicates the current drift ($\mathbf{d}_t$). The local shift distribution is presented by the dots on the circle of the player's enlarged image, where the size of each dot corresponds to the appropriate weight $G_t^{(k)}$. For better visualization only the angular shift distribution is shown here, i.e. the $p(\mathbf{v}|\mathbf{o}_{t-T:t-1})$ integrated over the radius

7 Experimental study and results

Several experiments were conducted to illustrate the performance of the proposed closed-world tracker from the Algorithm 1; for brevity, we will denote the tracker by $\mathbf{CW}_{ls}$.

The first experiment was designed to quantify the effect of the local smoothing (Section 6) in the $\mathbf{CW}_{ls}$. This experiment considered seven players of different colors sprinting on a path drawn on the playground (Fig. 3a) while performing sharp turns. The average size of each player was approximately 10×10 pixels. To establish a performance criterion, each player was manually tracked five times and the average of the five trajectories obtained for each player was taken as the ground truth. In this way, approximately 273 ground truth positions $p_t = (x_t, y_t)$ per player were obtained. The performance was evaluated in terms of the average RMS error

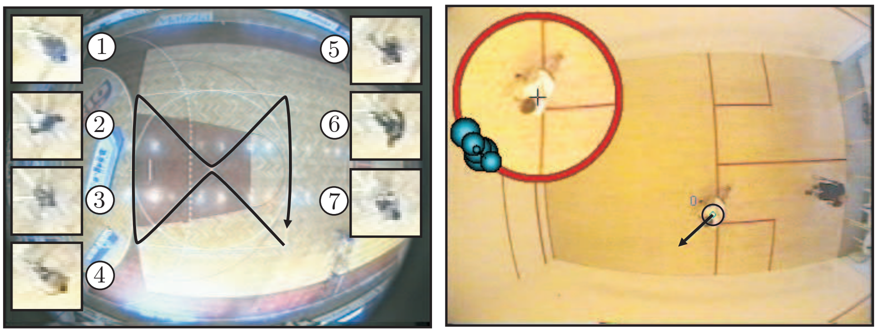
$$E = \frac{1}{7} \sum_{i=1}^{7} \frac{1}{R} \sum_{r=1}^{R} (\frac{1}{T} \sum_{t=1}^{T} \|{}^{(i)}p_t - {}^{(i)}\tilde{p}_t^{(r)}\|^2)^{\frac{1}{2}}, \tag{14}$$

where ${}^{(i)}p_t$ is the ground truth position at time-step t for the i-th player, ${}^{(i)}\tilde{p}_t^{(r)}$ is the corresponding estimated position for r-th replication of the experiment and $\| \cdot \|$ is the l_2 norm.

The $\mathbf{CW}_{ls}$ tracker was compared to a tracker that did not employ smoothing and where the adaptation and the background subtraction steps 5 and 6 in the Algorithm 1 were conducted directly on the MSE of the player's state from the particle filter. This tracker employed a random-walk model on the ellipse size with the standard deviation (std) of 5% of the current size, and a nearly constant velocity (NCV) dynamical model [12] on the position. The variances of the NCV velocity noise were learned on the ground-truth data and were set to $\sigma_{\dot{x}} = \sigma_{\dot{y}} = 0.94$pixel/frame. We denote this tracker by $\mathbf{CW}_{ncv}$.

The ellipse width and height in both trackers was constrained to lie within the interval of [8, 12] pixels. Each player was tracked thirty times (R=30) with the $\mathbf{CW}_{ls}$ and $\mathbf{CW}_{ncv}$. For each tracker, a RMS error (14) on the current position and prediction was calculated with respect to the ground-truth data. In order to evaluate the *repeatability* of the trackers, the average stds of the position estimates were also calculated. The form of the equation for calculating the average std is similar to (14), where the ground truth ${}^{(i)}p_t$ is replaced by the tracker's average estimate of the current position, obtained over all thirty repetitions.

The results are presented in Fig. 4. The $\mathbf{CW}_{ls}$ tracker in general outperformed $\mathbf{CW}_{ncv}$ attaining smaller errors on position and prediction as well as resulting in a smoother tracking. Using only 25 particles, the $\mathbf{CW}_{ls}$ performed equally well as the $\mathbf{CW}_{ncv}$ at 75 particles with respect to the RMS error on position (Fig. 4a) and outperformed $\mathbf{CW}_{ncv}$ even when the number of particles in both trackers was increased. Regarding the RMS error on prediction, the $\mathbf{CW}_{ls}$ generally achieved smaller errors than $\mathbf{CW}_{ncv}$. As the number of particles increased, the errors became virtually the same (Fig. 4b), however, $\mathbf{CW}_{ls}$



(a) (b)

Fig. 3. The left image (a) shows seven players and the path used in the first two experiments. The right image (b) shows an example of a tracked squash player from the third experiment

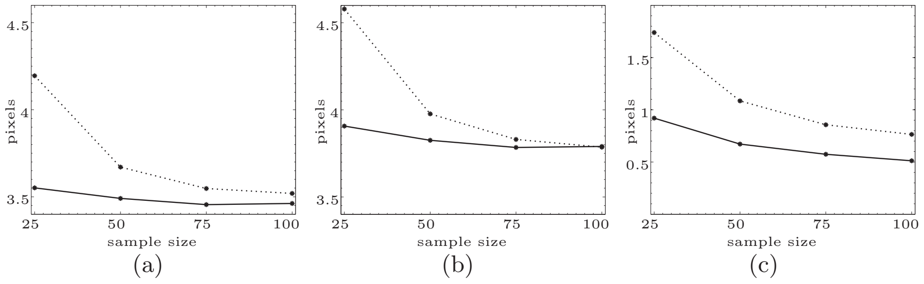
still maintained a smoother track (Fig. 4c). An important thing to note here, is that the $\mathbf{CW}_{ls}$ outperformed the $\mathbf{CW}_{ncv}$ even though the dynamics in $\mathbf{CW}_{ncv}$ were learned on the test data. This implies powerful capabilities of using local smoothing of Section 6 in the closed-world tracking.

In the second experiment, all seven players were tracked with a reference tracker $\mathbf{T}_{ref}$ from [9] thirty times and the average failure rate was calculated. The conceptual difference between this tracker and the $\mathbf{CW}_{ls}$ was that the reference tracker did not make use of the background information and employed a NCV dynamical model. The number of particles used in this experiment was 25, the parameters of the dynamical model in $\mathbf{T}_{ref}$ were set as above, the variance in the likelihood function was set to $\sigma^2 = 0.04$, and the parameters for the model adaptation were set as in [9]. The reference tracker produced on average 2.2 failures per a player, while both $\mathbf{CW}_{ls}$ and $\mathbf{CW}_{ncv}$ never lost any player.

To demonstrate the generality of the $\mathbf{CW}_{ls}$, an experiment was conducted on a recording of a squash match (Fig. 3b) from the CVBASE data set [16]. Two players were separately tracked with $\mathbf{CW}_{ls}$ using the same parameters as in the first experiment, with the ellipse width and height constrained to lie within the interval of [16, 24] pixels. The tracker, again, maintained a successful track using only 25 samples. The videos demonstrating the results of all experiments are available online at [17].

8 Conclusion

A computationally efficient algorithm for tracking a single player in a sporting match was presented in this paper. The effectiveness of the tracker was achieved by considering the sporting event as a semi-controlled environment for which certain closed-world assumptions can be derived. The proposed tracker was eval-



(a) (b) (c)

Fig. 4. RMS estimation errors on position (a) and prediction (b) for the $\mathbf{CW}_{ncv}$ (*dotted*) and $\mathbf{CW}_{ls}$ (*full*) as a function of the number of particles, and the corresponding average standard deviations of the position estimates (c)

uated on a demanding data set and exhibited good capabilities of learning the player's dynamics, while attaining a successful track.

Since the proposed tracker is based on a simple particle filter, i.e. the Condensation algorithm, it is expected to improve in performance if a more efficient particle filter is considered. Note also that the proposed algorithm is general enough to allow extensions to the case of multiple targets.

References

1. Needham, C.J., Boyle, R.D.: Tracking multiple sports players through occlusion, congestion and scale. In: BMVC01, Manchester, UK (2001) 93–102
2. Vermaak, J., Doucet, A., Pérez, P.: Maintaining multi-modality through mixture tracking. In: ICCV03. Volume 1., Nice, France (2003) 110–116
3. Choi, K., Seo, Y.: Probabilistic tracking of the soccer ball. In: Statistical Methods in Video Processing. (2004) 50–60
4. Czyz, J., Ristic, B., Macq, B.: A color-based particle filter for joint detection and tracking of multiple objects. In: ICASSP05. (2005)
5. Okuma, K., Taleghani, A., De Freitas, N., Little, J.J., Lowe, D.G.: A boosted particle filter: Multitarget detection and tracking. Volume 1. (2004) 28–39
6. Isard, M., Blake, A.: Condensation – conditional density propagation for visual tracking. International Journal of Computer Vision **29** (1998) 5–28
7. Arulampalam, M., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. Transactions on Signal Processing **50** (2002) 174–188
8. Intille, S.S., Bobick, A.F.: Visual tracking using closed-worlds. In: ICCV95. (1995) 672–678
9. Nummiaro, K., Koller-Meier, E., Van Gool, L.: Color features for tracking non-rigid objects. Chinese Journal of Automation **29** (2003) 345–355
10. Saito, N., Coifman, R.: Improved local discriminant bases using empirical probability density estimation. In: Computing Section of American Statistical Association. (1997) 312–321

11. Akaike, H.: A new look at the statistical model identification. IEEE Transactions on Automatic Control **19** (1974) 716–723
12. Rong Li, X., Jilkov P., V.: Survey of maneuvering target tracking: Dynamic models. Transactions on Aerospace and Electronic Systems **39** (2003) 1333–1363
13. Erdmann, W.S.: Gathering of kinematic data of sport event by televising the whole pitch and track. (1992) 159–162
14. Bangsbo, J.: The physiology of soccer: With special reference to intense intermittent exercise. Acta Physiologica Scandinavica **619** (1994) 1–155
15. Torma, P., Szepesvari, C.: On using likelihood-adjusted proposals in particle filtering: Local importance sampling. In: ISPA05. (2005)
16. CVBASE06: Squash1.avi. http://vision.fe.uni-lj.si/cvbase06 (19/01/06)
17. http://vision.fe.uni-lj.si/research/cvbase06kristan.html (19/01/06)