DISTRIBUTED PARTICLE FILTERING FOR MULTIOCULAR SOCCER-BALL TRACKING

Toshihiko Misu, Atsushi Matsui, Masahide Naemura, Mahito Fujii, and Nobuyuki Yagi

Science & Technical Research Laboratories
NHK (Japan Broadcasting Corporation)
1-10-11 Kinuta, Setagaya-ku, Tokyo 157-8510, Japan

ABSTRACT

This paper proposes a distributed state estimation architecture for multi-sensor fusion. The system consists of networked subsystems that cooperatively estimate the state of a common target from their own observations. Each subsystem is equipped with a self-contained particle filter that can operate in stand-alone as well as in network mode with a particle exchange function. We applied this flexible architecture to 3D soccer-ball tracking by modeling the imaging processes related to the centroid, size, and motion-blur of a target, and by modeling the dynamics with ballistic motion, bounce, and rolling. To evaluate the precision and robustness of the system, we conducted experiments using multiocular images of a professional soccer match.

Index Terms— Distributed tracking, tracking filters, dynamics, position measurement, state estimation

1. INTRODUCTION

Object tracking technologies are getting more and more attention from researchers as public interest grows in surveillance, robotics, intelligent traffic system, etc. In particular, broadcasters are interested in such technologies for annotations of sportscasts; images showing player/ball paths have become indispensable for analyzing scenes, since they tell a lot about the tactical situation on the field [1]. With respect to object tracking technologies, the common issues in the above fields are robustness, precision, speed, and flexibility.

Robustness to occlusion is a primary concern in sports applications, since the players tend to interfere with each other, or the ball. Many solutions have been proposed for handling occlusion: merge-split labeling, multiview measurement, nongaussianity modeling, etc. [2]. The multiview/multimodal approach [3] also provides a strong constraint on localization, and it can also be a means for acquiring finer precision.

In this paper, we adopt a Bayesian state estimation framework that allows an arbitrary number/formation of cameras. The particle filter observes the ball image from various aspects in order to utilize as many measures on each binarized silhouette: centroid, area, and orientation that reflect line-of-sight, depth, and velocity. To be able to treat complex dynamics, Yan et al. [4] developed a 2D tennis-ball tracker that bifurcates into flying and racket-hit modes. We extend their idea to 3D space to have the tracker manage flying, bouncing, and rolling motions.

For speed and flexibility, Coates [5] proposed a distributed particle filter that relays a learnt likelihood function to next subsystem. Our tracker also distributes the multiocular task to multiple subsystems (tracking units) on individual PCs, but differs from Coates' filter in its strategies for data-fusion; his filter fuses information in a likelihood space, whereas ours works in a state space, neither with any crucial dependencies on other subsystems nor with supervisors such as in [3].

2. SYSTEM ARCHITECTURE

The tracking system (Fig. 1) consists of networked tracking units, each of which captures images from a camera. The tracking unit is equipped with an ordinary particle filter that is capable of estimating the distribution of the target state $\boldsymbol{x}(t)$ based on its monocular observation and on the ball dynamics as depicted in the upper part of Fig. 1. The state $\boldsymbol{x}(t)$, which is to be stored in the "Particle Memory," contains the 3D position $\boldsymbol{s} = [s_X, s_Y, s_Z]^T$, the velocity $\dot{\boldsymbol{s}}$, and the acceleration $\ddot{\boldsymbol{s}}$ of the ball at an instant t:

$$\boldsymbol{x}(t) \quad = \quad [\boldsymbol{s}(t)^T, \dot{\boldsymbol{s}}(t)^T, \ddot{\boldsymbol{s}}(t)^T]^T \ .$$

Nonlinear multimodal modeling of the imaging process including motion blur (see Section 2.3) is implemented in the "Measurement" block. This modeling is an attempt at more efficient scoring of particles than would be possible with common centroid-based trackers. The combinatorial dynamics (see Section 2.4) implemented in "Prediction" and "Time-Lag Compensation" blocks forces a tight-fit constraint that can resolve temporally unstable measurements (i.e. occlusions).

The lower part of Fig. 1 chooses a set of particles from the "Particle Memory," and multicasts it with a timestamp over the network. When a unit receives particles from the other units, "Fusion2" replaces most of the memorized particles with the received particles, leaving a number of its own particles, thereby maintaining a way of bypassing unexpected malfunction in other units (see Section 2.1).

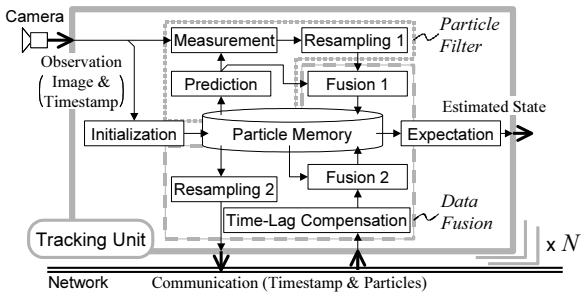


Fig. 1. Tracking unit.

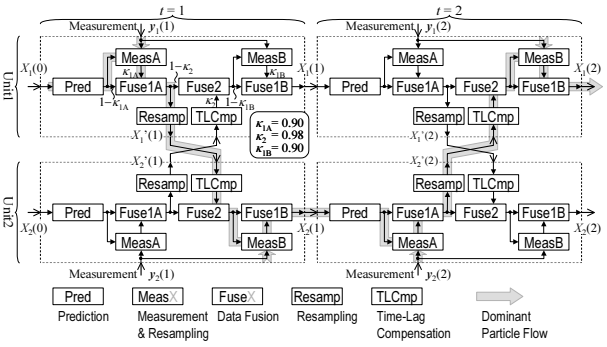


Fig. 2. Flow of particles.

Hereinafter, the probability density function $p(\boldsymbol{x})$ for a state $\boldsymbol{x}$ is approximated in the particle form with K state vectors $\{\boldsymbol{x}^{\langle k\rangle}\}_{k=1}^{K}$ and their importance weights $\{w^{\langle k\rangle}\}_{k=1}^{K}$. $X^{\langle k\rangle} = \{\boldsymbol{x}^{\langle k\rangle}, w^{\langle k\rangle}\}$ denotes the k-th particle. A measurement by the n-th camera at an instant t is denoted by $\boldsymbol{y}_n(t)$.

2.1. Data Fusion

Each of the "Fusion" blocks in Fig. 1 mixes the unit's own particles with those received from the others units (or those updated by its "Measurement" block) so as to gain the resulting ingredient proportion of (own) : (alian) $= (1{-}\kappa) : \kappa$. The smaller coefficient $\kappa \in [0, 1]$ makes the tracking unit more independent (i.e. performs monocularly) on the others. Fig. 2 illustrates an example of the particle flow between two tracking units. For simplicity, we assume that the tracking units are synchronized with each other with constantly zero time lag, and that the "Measurement" block fires twice a cycle, before and after it receives particles.

First, MeasA updates the importance weights based on a measurement $\boldsymbol{y}_1(1)$. Then, Fuse1A merges the updated and not updated particles in the proportion of $90 : 10$. Thus, the 90% of the particles experience the measurement $\boldsymbol{y}_1(1)$.

TLCmp of Unit2 compensates the time lag if required (no time lag in this case). At Fuse2, 98% of Unit2's own particles are replaced with ones received from Unit1. After MeasB and Fuse1B, $(0.90 \times 0.98 \times 0.90)K$ particles successively experience both measurements $\boldsymbol{y}_1(1)$ and $\boldsymbol{y}_2(1)$. In other words, 79% of particles have importance weights that have been subjected to a factorized likelihood of $p(\boldsymbol{y}_2(1)|\boldsymbol{x}(1)) \cdot p(\boldsymbol{y}_1(1)|\boldsymbol{x}(1))$. The rest "half-baked" particles (with a small minority of "completely raw" ones) will leave a chance to overcome malfunctions of the measurement process(es).

In case of occlusion, frame-out, or camera trouble, the unit automatically switches to a hands-off policy ($\kappa_{1A} = 0$ or $\kappa_{1B} = 0$), where it passively accepts the others' estimates. Since the tracking unit, in real environment, asynchronously invokes measurement(s) and particle reception in a best-effort manner, the system can deal with time-varying latency by using TLCmp and it deals with unexpected system failures by bypassing Fuse2 (i.e. $\kappa_2 = 0$) after a time-out period elapses.

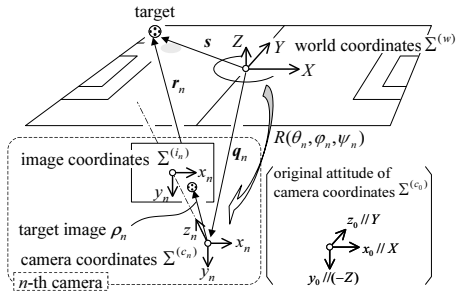


Fig. 3. Coordinate systems.

2.2. Coordinate Systems

Fig. 3 illustrates the coordinate systems used in the following discussion. Our method estimates the ball trajectory in a world coordinate system $\Sigma^{(w)}$, which is fixed upon the real world. The three axes x_0, y_0 and z_0 in the X-, $(-Z)$- and Y-directions of $\Sigma^{(w)}$ define the common original attitude $\Sigma^{(c_0)}$ of the cameras, and successive relative rotations of pan θ_n, tilt φ_n and roll ψ_n determine the n-th camera's attitude, which is expressed with a rotation matrix $R(\theta_n, \varphi_n, \psi_n)$.

The n-th camera is modeled as a pinhole of focal length f_n that maps a state vector $\boldsymbol{x}$ (which contains 3D position $\boldsymbol{s}^{(w)}$) onto its image coordinates $\boldsymbol{\rho}_n^{(i_n)}$ in $\Sigma^{(i_n)}$ as follows:

$$\boldsymbol{\rho}_n^{(i_n)} = f_n \boldsymbol{r}_n^{(c_n)} \Big/ ([0,0,1]\boldsymbol{r}_n^{(c_n)})$$
$$\boldsymbol{r}^{(c_n)} = R(\theta_n, \varphi_n, \psi_n)([I_{3\times3}, O_{3\times3}, O_{3\times3}]\boldsymbol{x} - \boldsymbol{q}_n^{(w)}) .$$

Using the camera parameters $\boldsymbol{\theta}_n = [\theta_n, \varphi_n, \psi_n, f_n, [\boldsymbol{q}_n^{(w)}]^T]^T$, we employ the following simple expression for the projection:

$$\boldsymbol{\rho}_n^{(i_n)} = \boldsymbol{h}(\boldsymbol{x} \mid \boldsymbol{\theta}_n) .$$

2.3. Measurement Model

The "Measurement" block calculates the importance weight $w^{\langle k\rangle}$ based on the likelihood $p(\boldsymbol{y}(t) \mid \boldsymbol{x}(t))$ for a measurement vector $\boldsymbol{y}(t)$ of the input image sequence:

$$w^{\langle k\rangle}(t) = p(\boldsymbol{y}(t) \mid \boldsymbol{x}^{\langle k\rangle}(t)) .$$

Table 1. Physical parameters and typical values

parameters		typical values
ball radius	r_B	0.11 m
gravitational acceleration	g	9.8 m/s^2
dynamic coefficient of friction	μ	0.31
maximum height of friction	H	0.30 m
coefficient of restitution	e	0.35
focal length	f	15 mm
exposure time	T	1/59.94 s
pixel dimensions	$u_x \times u_y$	10 μm $\times$ 20 μm
spatio-temporal resolution		(1920$\times$540) pel $\times$ 14.985 Hz

We chose the centroids $\boldsymbol{\eta}$, the areas $\boldsymbol{\alpha}$, and the orientations $\boldsymbol{\beta}$ of the silhouettes of ball candidates as components of the measurement $\boldsymbol{y}$, and defined the likelihood $p(\boldsymbol{y} \mid \boldsymbol{x})$ in a factorized form of individual measurements $\boldsymbol{\eta}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$:

$$p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{\theta}) = p(\boldsymbol{\eta} \mid \boldsymbol{x}, \boldsymbol{\theta}) \cdot p(\boldsymbol{\alpha} \mid \boldsymbol{x}, \boldsymbol{\eta}, \boldsymbol{\theta}) \cdot p(\boldsymbol{\beta} \mid \boldsymbol{x}, \boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\theta}).$$

2.3.1. Extraction of Ball Candidates as Preprocess

Non-green objects inside the turf area and moving objects outside the turf are extracted as preliminary silhouettes by using a hybrid technique of chroma keying and inter-frame subtraction. M ball candidates ($M \geq 0$) are selected by thresholding the size, aspect ratio, and mean color of the silhouettes. In the following, $\mathcal{S}^{[m]}$ denotes the silhouette of the m-th candidate.

2.3.2. Likelihood Factor for Centroid

The static moment of the silhouette $\mathcal{S}^{[m]}$ defines the centroid $\boldsymbol{\eta}^{[m]}$. Assuming Gaussian noise $\mathcal{N}(\mathbf{0}, \Sigma_{\text{img}})$ is present in the measurement, we define the likelihood $p(\boldsymbol{\eta} \mid \boldsymbol{x}, \boldsymbol{\theta})$ as follows:

$$p(\boldsymbol{\eta} \mid \boldsymbol{x}, \boldsymbol{\theta}) \simeq C \exp\left(-\frac{1}{2} \min_{m=1,\ldots,M} \{d(\boldsymbol{\eta}^{[m]}, \boldsymbol{x})\}^2\right) \quad (1)$$

$$\{d(\boldsymbol{\eta}, \boldsymbol{x})\}^2 = [\boldsymbol{\eta} - \boldsymbol{h}(\boldsymbol{x} \mid \boldsymbol{\theta})] \Sigma_{\text{img}}^{-1} [\boldsymbol{\eta} - \boldsymbol{h}(\boldsymbol{x} \mid \boldsymbol{\theta})]^T,$$

where C is a constant. To reduce computational costs, Eq. (1) considers only the candidate that is closest to the projected particle; $\hat{m} = \arg\min_{m=1,\ldots,M} \{d(\boldsymbol{\eta}^{[m]}, \boldsymbol{x})\}$.

2.3.3. Likelihood Factor for Area

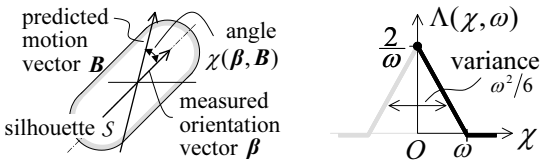
The likelihood factor for area is defined as

$$p(\boldsymbol{\alpha} \mid \boldsymbol{x}, \boldsymbol{\eta}, \boldsymbol{\theta}) \propto \exp\left\{\frac{-1}{2\sigma_\alpha^2} \left(\frac{A(\boldsymbol{x}) - \alpha^{[\hat{m}]}}{L} - \mu_\alpha\right)^2\right\}, \quad (2)$$

where A is the predicted area of a silhouette that consists of the area of the instantaneous image (the first term in the following) and that of motion blur (the second term):

$$A(\boldsymbol{x}) \simeq \frac{\pi r_B^2 f^2}{([0,0,1]\boldsymbol{r}^{(c)})^2} + 2 r_B f^2 T \| \dot{\boldsymbol{s}}^{(w)} - (\hat{\boldsymbol{z}}^{(w)} \cdot \dot{\boldsymbol{s}}^{(w)}) \hat{\boldsymbol{z}}^{(w)} \|$$

$$\hat{\boldsymbol{z}}^{(w)} = [R(\theta, \varphi, \psi)]^T [0,0,1]^T.$$



(a) Distance $\chi(\boldsymbol{\beta}, \boldsymbol{B})$ (b) Likelihood $\Lambda(\chi, \omega)$

Fig. 4. Definitions of distance χ and likelihood Λ.

We assumed that Gaussian noise $\mathcal{N}(\mu_\alpha, \sigma_\alpha^2)$ was added to the observed area normalized by the contour length L in Eq. (2) since the error on the observed area α due to defocus is approximately proportional to the contour length L.

2.3.4. Likelihood Factor for Orientation

We define the likelihood $p(\boldsymbol{\beta} \mid \boldsymbol{x}, \boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\theta})$ according to the orientation $\boldsymbol{\beta}$ of the silhouette's principal axis. As illustrated in Fig. 4(a), the distance $\chi(\boldsymbol{\beta}, \boldsymbol{B})$ between the vectors $\boldsymbol{\beta}$ and $\boldsymbol{B}$ is defined as the angle that the directions of the two vectors make. As the more elongated silhouette $\mathcal{S}^{[m]}$ will provide more robust information on the particle velocity, we defined a triangular likelihood function $\Lambda(\chi, \omega)$ whose width ω stretches/shrinks inversely proportional (with coefficient $B_0 = 1$) to the length $\|\boldsymbol{B}\|$ as shown in Fig. 4(b):

$$p(\boldsymbol{\beta} \mid \boldsymbol{x}, \boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\theta}) = \Lambda(\chi(\boldsymbol{\beta}^{[\hat{m}]}, \boldsymbol{B}(\boldsymbol{x} \mid \boldsymbol{\theta})), B_0 / \|\boldsymbol{B}\|).$$

2.4. State Transition Model

We modeled the state transition from $t = t_0$ to $t = t_1$ by using a composite function of parabolic flight $\boldsymbol{\phi}_p$, bounce $\boldsymbol{\phi}_b$, and decelerating rolling $\boldsymbol{\phi}_f$ due to friction (see also Table 1):

$$\boldsymbol{x}(t_1) = (\boldsymbol{\phi}_f \circ \boldsymbol{\phi}_b \circ \boldsymbol{\phi}_p)(\boldsymbol{x}(t_0), t_1 - t_0) + \boldsymbol{\nu}(t_1 - t_0)$$

$$\boldsymbol{\phi}_p(\boldsymbol{x}, \Delta t) = \begin{bmatrix} I_{3\times3} & (\Delta t) I_{3\times3} & (\Delta t)^2 I_{3\times3}/2 \\ O_{3\times3} & I_{3\times3} & (\Delta t) I_{3\times3} \\ O_{3\times3} & O_{3\times3} & I_{3\times3} \end{bmatrix} \boldsymbol{x}$$

$$\boldsymbol{\phi}_b(\boldsymbol{x}, \Delta t) = \begin{cases} \boldsymbol{x} & (s_Z \geq r_B) \\ [s_X, s_Y, 2B - s_Z, \dot{s}_X, \dot{s}_Y, -e\dot{s}_Z, \ddot{s}_X, \ddot{s}_Y, \ddot{s}_Z]^T & (\text{otherwise}) \end{cases}$$

$$\boldsymbol{\phi}_f(\boldsymbol{x}, \Delta t) = \begin{cases} \boldsymbol{x} & (s_Z > H) \\ \boldsymbol{x} - \left[0, \ldots, 0, \frac{-[s_X, s_Y]}{\sqrt{s_X^2 + s_Y^2}} \mu g, 0\right]^T & (\text{otherwise}) \end{cases},$$

where $\boldsymbol{\nu}(\Delta t) \sim \mathcal{N}(\mathbf{0}, \Sigma_{\boldsymbol{\nu}} \Delta t)$ is additive process noise whose variance is proportional to the time interval $\Delta t = t_1 - t_0$.

3. EXPERIMENTS

We conducted experiments using multiview professional soccer images that were acquired by two fixed high-definition cameras on the roof of a stadium with a baseline of 31 [m].

Table 2. Errors in estimated 3D trajectories

instant	t [s] $\longrightarrow$	1.20	2.20	4.14	5.47
absolute	$\|\mathrm{U}_1-\mathrm{GT}\|$ [m]	0.46	0.27	1.21	0.31
errors	$\|\mathrm{U}_2-\mathrm{GT}\|$ [m]	0.62	0.25	1.18	0.34
discrepancy	$\|\mathrm{U}_2-\mathrm{U}_1\|$ [m]	0.93	0.09	0.15	0.17

(U_1: Unit1's estimate, U_2: Unit2's estimate, GT: ground truth)

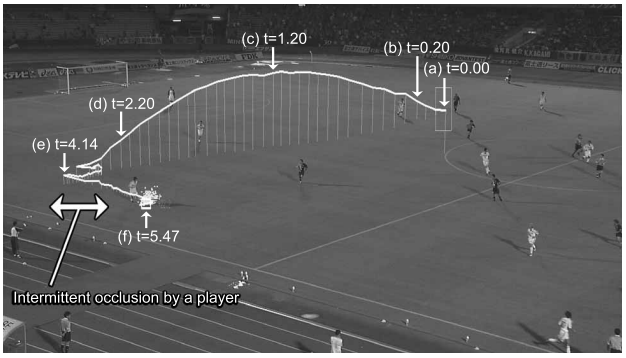


Fig. 5. Trajectory estimated by Unit1.

Each tracking unit owned $K = 1000$ particles, all of which were transmitted to the party. Table 1 lists the settings.

Assigned respectively to the left and the right cameras, the two tracking units (Unit1 and Unit2) obtained almost the same loci as evaluated in Table 2. Fig. 5 visualizes the estimated path of the ball that underwent various dynamics: ballistic flight (a)–(d), intermittent occlusion around (e), and rolling (f). During the occlusion phase, the system got binocular, monocular, or no observation depending on the player/ball positions. The seamless switching of κ_{1A} and κ_{1B} in Fig. 2 made full use of available viewpoints to minimize increases in positional errors.

Figs. 6 and 7 show the behaviors of the tracking units when they were temporally isolated from each other by a network failure. At $t = 1.00$ [s], the particle distribution was unimodal because of the prior binocular observation. Although the particles scattered in line-of-sight directions during the failure period, they shrank to the intersection point of the two line-of-sight rays after reconnection, and regained their precision. These results also indicate the tracking units' dynamic particle-handover and on-line plug-in/-out capabilities.

We also tested the trackers in another camera formation and got similar results. The prototype with dual Xeon 3.2 GHz processors per tracking unit required about 0.5 [s/frame] of computation time, including time for preprocessing.

4. CONCLUSIONS

We proposed a novel distributed particle filter that exchanges particles among tracking units. Implementing the various measurements and complex state-transition processes of a ball, we constructed a soccer-ball tracker. The experiments showed the architecture could seamlessly integrate multiview information to improve positional accuracy and robustness against occlusion and system instability. We are planning to build a sports annotation tool that can detect special events such as corner-kicks by using the ball and players' paths.

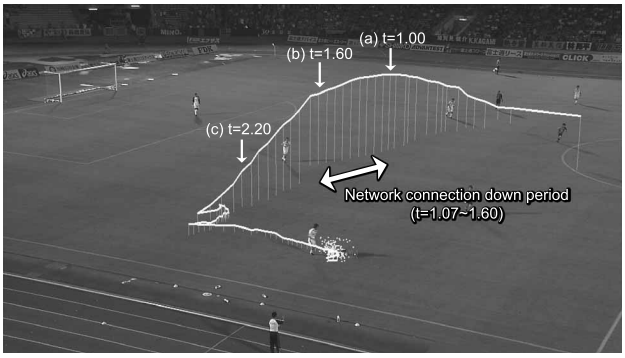


Fig. 6. Trajectory by Unit2 during a network failure.

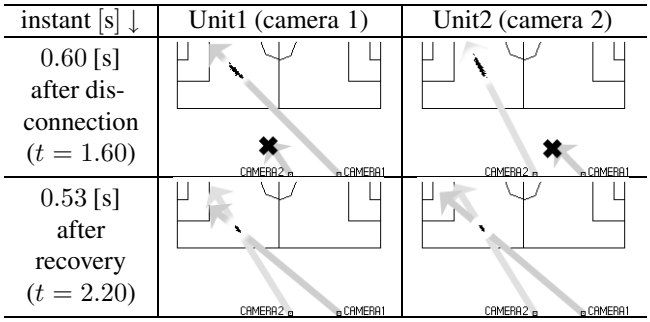


Fig. 7. Particle distributions after disconnection and recovery.

5. REFERENCES

[1] K. Okuma, J. J. Little, and D. Lowe, "Automatic acquisition of motion trajectories: Tracking hockey players," *Proc. SPIE*, vol. 5304, pp. 202–213, 2003.

[2] P. F. Gabriel, J. G. Verly, J.H. Piater, and A. Genon, "The state of the art in multiple object tracking under occlusion in video sequences," *Advanced Concepts for Intelligent Vision Systems*, pp. 166–173, 2003.

[3] J. B. Hayet, M. Mathes, J. Czyz, J. Piater, J. Verly, and B. Macq, "A modular multi-camera framework for team sports tracking," *Proc. AVSS'2005*, pp. 493–498, 2005.

[4] F. Yan, W. Christmas, and J. Kittler, "A tennis ball tracking algorithm for automatic annotation of tennis match," *BMVC05*, 2005.

[5] M. Coates, "Distributed particle filters for sensor networks," *Information Processing in Sensor Networks*, vol. IPSN2003, Springer, pp. 99–107, 2004.