A Local-motion-based probabilistic model for visual tracking[1]

M. Kristan[b,a,1], J. Perš[b], S. Kovačič[b], A. Leonardis[a]

[a] *Faculty of Computer and Information Science, University of Ljubljana, Slovenia*
[b] *Faculty of Electrical Engineering, University of Ljubljana, Slovenia*

Abstract

Color-based tracking is prone to failure in situations where visually similar targets are moving in a close proximity or occlude each other. To deal with the ambiguities in the visual information, we propose an additional color-independent visual model based on the target's local motion. This model is calculated from the optical flow induced by the target in consecutive images. By modifying a color-based particle filter to account for the target's local motion, the combined color/local-motion-based tracker is constructed. We compare the combined tracker to a purely color-based tracker on a challenging dataset from hand tracking, surveillance and sports. The experiments show that the proposed local-motion model largely resolves situations when the target is occluded by, or moves in front of, a visually similar object.

Key words: Local-motion, Probabilistic visual models, Visual tracking, Occlusion
PACS: :

1. Introduction

In recent years, particle filters [1] have become a popular approach to tracking from video due to their ability to efficiently handle the uncertainties associated with the visual data and the target's dynamics. Probabilistic trackers such as particle filters usually use contour-based [2] or color-based [3, 4] appearance models to locate and track the target. One drawback of these models is that tracking may fail whenever the target gets in close proximity of another visually similar object. In many applications, such as video surveillance, visual human-computer interface and tracking in sports, the camera is often positioned such

[1] Accepted to Pattern Recognition Journal, 5 January 2009, Available online 10 January 2009, doi:10.1016/j.patcog.2009.01.002

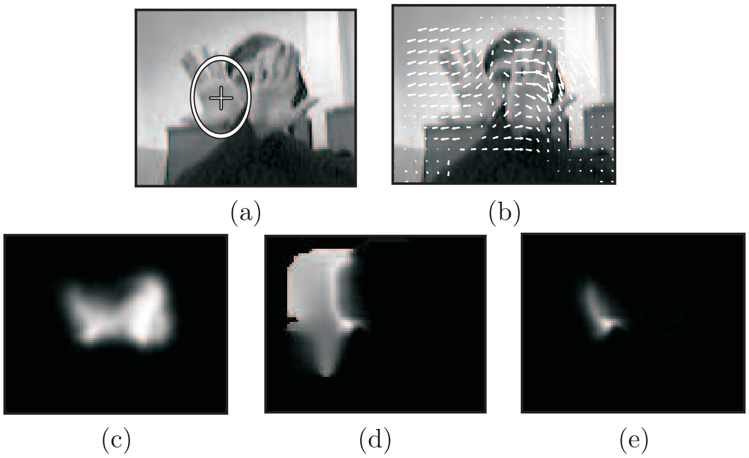
[*] Corresponding author.

URL: `http://www.vicos.uni-lj.si` (M. Kristan) that the scene is viewed from the side only. In these situations, complete occlusions between visually similar objects become quite frequent. Figure 1a shows an example where a person's right hand was tracked through an occlusion by the left hand. Note that the color likelihood function (Figure 1c) is ambiguous with respect to the position of the tracked hand – the mode stretches over both hands, which usually causes tracking to fail.

One solution to disambiguate between similarly colored objects is to use additional color-independent cues, such as edge-orientation histograms [5, 6], point distribution models [7] or texture features [8]. Although these cues alleviate the ambiguities caused by the color similarity, they are intensity-related and are still hampered by the visual similarity between different objects. For that reason several authors have proposed to improve tracking by combination of these models. Isard and Blake [9] use color and contours to track hands on a cluttered background. Li and Chaumette [10] combine shape, color, structure and edge information to improve tracking through varying lighting conditions and cluttered background. Similarly, Stenger et al. [11] and Wang et al. [12] combine color and edge features to make tracking robust to background clutter. Recently, Brasnett et al. [8] proposed a weighted scheme to combine edge, color and texture cues.

Another approach is to utilize an *appearance-independent* cue such as motion. The simplest way to detect motion in images is to calculate the difference between consecutive images. Viola and Jones [13], for example, improved pedestrian detection by learning a cascade of weak classifiers on manually extracted patches of image differences. A probabilistic model of local differences was proposed by Pérez et al. [14]. They partition the image into an array of cells and assume that a cell contains motion if the differences in that cell are approximately uniformly distributed. A Parzen estimator [15] is then applied to produce a motion-based importance function, which is used within a particle filter to guide particles into the regions of the image which contain motion. A drawback of methods which rely on image differencing is that they are essentially local-change detectors and therefore cannot resolve situations when a target is occluded by a moving, visually similar, object. Du et al. [16] have proposed a general multiple-cue integration framework based on Linked Hidden Markov Models and integrated the detected local-changes with other visual features to improve tracking when the tracked object does not exhibit any motion.

An obvious solution is thus to take into account the apparent motion in the images – the *optical flow*. Various bottom-up approaches have been proposed recently, which are based on clustering similar flows to yield moving objects. An attempt to track solely by the optical flow was presented by Du and Piater [17]. In their approach a Kanade-Lucas-Tomasi (KLT) feature tracker [18] was implemented in the context of a mixture particle filter. Targets were identified in each frame by clustering consistent optical flow features. A similar approach was used in [19], where the flow vectors were clustered by region growing and pruning using affine motion consistency as a criterion. Recently, an approach was presented in [20] where the optical flow was used to extract stable trajectories of features. These were then clustered using a minimum-description-length


(a) (b)

(c) (d) (e)

Figure 1: A person's hand (depicted by the ellipse) is tracked after being occluded by the other hand (a). The optical flow induced by both hands is depicted in (b), while the color likelihood and the optical flow likelihood are shown in (c) and (d), respectively. Bright colors correspond to high likelihood, while dark colors correspond to low likelihood. Image (e) shows the combined likelihood of (c) and (d). The position corresponding to the maximum likelihood in (e) is depicted by a white cross in (a).

method to determine the number of independently moving bodies. A drawback of the bottom-up approaches which are based on clustering flow vectors is that, due to the clustering procedure and the nature of the optical flow data, they cannot maintain correct identities of the targets after full occlusion even if the targets are of different colors. Bugeau and Pérez [21] approach this problem by also accounting for the color information in the clustering stage and apply graph cuts to improve segmentation.

1.1. Our approach

We propose a new local motion model, based on the apparent motion in the image, which can be probabilistically integrated with other features to improve tracking in presence of occlusions between visually similar objects and other visual ambiguities. Recall the hand tracking problem from Figure 1. The induced optical flow is shown in Figure 1b and an illustration of the local-motion likelihood, where we consider only the direction of motion, is shown in Figure 1d. Note that, while one of the modes corresponds to the tracked hand, the other hand is *hidden* by this distribution. The product of the local-motion and color likelihoods is shown in Figure 1e, where a single clear mode corresponding to the tracked hand remains, indicating that the visual ambiguity was resolved.

The main contribution of this paper is the novel probabilistic local-motion model which improves tracking when a target is occluded by, or moves in proximity of, a visually similar object. The proposed probabilistic model is composed of three major parts. The first two are the novel local-motion feature, which is calculated from the target's optical flow, and the similarity measure which allows comparing the local-motion model to the observed motion in the image. The third part is the adaptation scheme which considers the predicted velocity from the tracker in adaptation of the local-motion model to the target's motion, and can cope with target occlusions. We also derive a probabilistic measurement model of the local motion, which allows application within probabilistic frameworks such as particle filters. As an example, we extend an existing color-based particle filter to account for the local-motion using a simple data-fusion approach, e.g., [22]. We demonstrate that the proposed local-motion model significantly improves tracking on a challenging dataset using examples from hand tracking, surveillance, and sports tracking.

The remainder of the paper is organized as follows. In Section 2 we give a brief overview of the bootstrap particle filter. Section 3 introduces the optical-flow-based local-motion feature, its probabilistic model, and the adaptation scheme. The local-motion-based probabilistic tracker is described in Section 4, and in Section 5 the results of the experiments are reported. We conclude the paper in Section 6.

2. Bootstrap particle filter

We give here only the basic concept of the particle filters and notations, and refer the reader to [23] for more details. Let $\mathbf{x}_{t-1}$ denote the state (e.g., position and size) 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 encompassed by its posterior $p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})$. During tracking, this posterior is recursively estimated as the new observations $\mathbf{y}_t$ arrive, which is realized in two steps: prediction (1) and update (2),

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

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

The recursion (1,2) for the posterior, in its simplest form, thus requires a specification of a 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 $p(\mathbf{y}_t|\mathbf{x}_t)$.

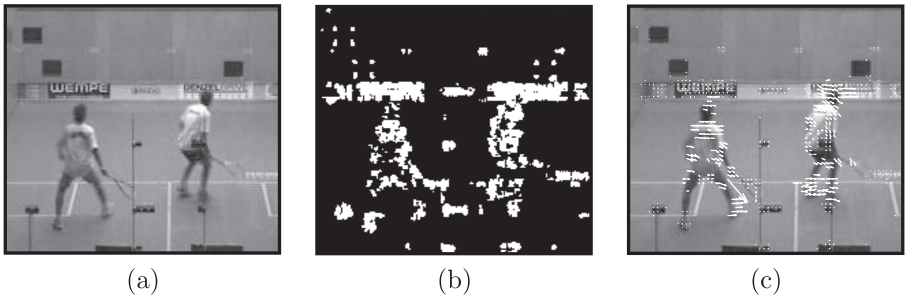
In our implementation we use a simple bootstrap particle filter [24, 2]. The posterior at time-step $t-1$ is estimated by a finite Monte Carlo set of states $\mathbf{x}_{t-1}^{(i)}$ and their respective weights $w_{t-1}^{(i)}$, $p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \approx \{\mathbf{x}_{t-1}^{(i)}, w_{t-1}^{(i)}\}_{i=1}^{N}$, such that all weights in the particle set sum to one. At time-step t the particles are first resampled according to their weights, in order to obtain an unweighted representation of the posterior $p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \approx \{\tilde{\mathbf{x}}_{t-1}^{(i)}, \frac{1}{N}\}_{i=1}^{N}$. Then they are propagated according to the dynamical model $p(\mathbf{x}_t|\tilde{\mathbf{x}}_{t-1}^{(i)})$, to obtain a representation of the prediction $p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) \approx \{\mathbf{x}_{t-1}^{(i)}, \frac{1}{N}\}_{i=1}^{N}$. Finally, a weight is assigned to each particle according to the likelihood function $w_t^{(i)} \propto p(\mathbf{y}_t|\mathbf{x}_t^{(i)})$, all weights are normalized to sum to one, and the posterior at the time-step t is approximated by a new weighted particle set $p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \{\mathbf{x}_t^{(i)}, w_t^{(i)}\}_{i=1}^N$. The current state of the target $\hat{\mathbf{x}}_t$ can then be estimated as the minimum mean-square error (MMSE) estimate over the posterior $p(\mathbf{x}_t|\mathbf{y}_{1:t})$

$$\hat{\mathbf{x}}_t = \sum\nolimits_{i=1}^N \mathbf{x}_t^{(i)} w_t^{(i)}. \tag{3}$$

3. Optical-flow-based local-motion feature

The optical flow at a given point is calculated by estimating the optical flow vector form the current image back to the previous image, and reversing its direction. This approach was chosen since it relates the current image to the previous image, which is in agreement with the scheme, described in the following sections, by which we estimate the reference motion of the target.

In our approach, the optical flow is estimated using the pyramidal implementation [25] of the well known Lucas-Kanade method [18]. One drawback of this method is that it fails to provide a reliable estimation of the flow vectors in regions with poor local texture. We therefore apply Shi-Tomasi feature detection [26] to determine locations with sufficient local texture, and calculate the optical flow only at those locations. The Shi-Tomasi feature at location (x, y) is defined by the smallest eigenvalue of the covariation matrix of gray-scale intensity gradients, which are calculated in the neighborhood of (x, y). The location (x, y) is accepted as a valid Shi-Tomasi feature if the smallest eigenvalue exceeds a predefined threshold ξ_{th}. An example of valid Shi-Tomasi features and the corresponding flow vectors are shown in Figure 2.



(a) (b) (c)

Figure 2: Two players of a squash match are shown in (a). The valid Shi-Tomasi features are depicted by white color in image (b) and the corresponding flow vectors are shown in (c). For clarity, only every third flow vector is shown.

Let $\mathbf{v}_t(x, y) = [r, \phi]$ be the optical flow vector at location (x, y) in the current image with amplitude r and orientation ϕ. The local-motion feature $\mathbf{v}_E = [r_E, \phi_E]$ of a region E is then encoded as the weighted average of the flow vectors

$$\mathbf{v}_E = f_v^{-1} \sum_{(x,y)\in E'} \mathbf{v}_t(x, y) K(x, y), \tag{4}$$

where $E' \in E$ is a set of detected Shi-Tomasi features within region E, $K(x,y)$ is the Epanechnikov kernel [15] used to assign higher weights to those flow vectors that are closer to the center of E, and $f_v = \sum_{(x,y)\in E'} K(x,y)$ is a normalization term. To avoid the pathological situations associated with vectors with amplitude zero, the summation (4) is carried out in Cartesian coordinates.

3.1. Local-motion likelihood

Let $\mathbf{v}_{\text{ref}} = [r_{\text{ref}}, \phi_{\text{ref}}]$ be a reference vector, which models the target's local-motion, and let $\mathbf{v}_E$ be a local-motion vector calculated within region E. We define the angular and amplitude similarity measure G_ϕ and G_r, respectively, between $\mathbf{v}_{\text{ref}}$ and $\mathbf{v}_E$ as

$$G_\phi(\mathbf{v}_E, \mathbf{v}_{\text{ref}}) = \begin{cases} \frac{\angle(\mathbf{v}_E, \mathbf{v}_{\text{ref}})}{\pi} & ; \quad r_E > \delta_{\text{th}} \wedge r_{\text{ref}} > \delta_{\text{th}} \\ 1 & ; \quad \textit{otherwise} \end{cases}, \tag{5}$$

$$G_r(\mathbf{v}_E, \mathbf{v}_{\text{ref}}) = \begin{cases} \frac{|r_{\text{ref}} - r_E|}{r_{\text{ref}} + r_E} & ; \quad r_E > \delta_{\text{th}} \vee r_{\text{ref}} > \delta_{\text{th}} \\ 0 & ; \quad \textit{otherwise} \end{cases}, \tag{6}$$

such that $G_\phi(\cdot,\cdot) \in [0,1]$, $G_r(\cdot,\cdot) \in [0,1]$, $\angle(\cdot,\cdot)$ is the angle between two vectors, $|\cdot|$ is the L_1 norm, and δ_{th} is a threshold on the amplitude below which the vectors are considered as noise[2]. If region E contains no valid Shi-Tomasi features, the vector $\mathbf{v}_E$ is undefined and the similarity measures are $G_\phi = 1.0$ and $G_r = 1.0$.

We have observed, in a preliminary study, that, whenever the target is correctly located, the probability density functions (pdf) of (5) and (6) can be well approximated by exponential distributions. However, in practice we approximate the current reference motion using motions observed in previous time-steps. This may impair the quality of tracking whenever the target suddenly significantly changes its motion. To cope with such events, we introduce a uniform component to the probability density function. The joint probability density function of (5) and (6) with parameters $\theta = [\lambda_\phi, \lambda_r, w_{\text{noise}}]$ is then defined as

$$p(G_\phi, G_r|\theta) \propto (1 - w_{\text{noise}}) e^{-(\frac{G_\phi}{\lambda_\phi} + \frac{G_r}{\lambda_r})} + w_{\text{noise}}, \tag{7}$$

where λ_ϕ and λ_r are the parameters of the exponential distributions and $0 < w_{\text{noise}} < 1$ is the weight of the uniform component.

3.2. Adaptation of the local-motion feature

After each tracking iteration, the current state $\hat{\mathbf{x}}_t$ of the target and its current velocity $\hat{\mathbf{v}}_t$ are calculated, e.g., via the MMSE estimate (3) from the particle filter. The new region E containing the target is determined and the local-motion vector $\mathbf{v}_{Et} = [\phi_{Et}, r_{Et}]$ (4) is estimated. If the region E contains at

[2] Note that the similarity measures $G_\phi(\cdot,\cdot)$ and $G_r(\cdot,\cdot)$ are actually the *distance measures* between the vectors $\mathbf{v}_{\text{ref}}$ and $\mathbf{v}_E$: in case $\mathbf{v}_{\text{ref}}$ and $\mathbf{v}_E$ are equal, they yield a value 0 and when $\mathbf{v}_{\text{ref}}$ cannot be considered close to $\mathbf{v}_E$, they yield a value greater than 0.

least one valid Shi-Tomasi feature, then $\mathbf{v}_{Et}$ is used to adapt the reference local-motion model $\mathbf{v}_{\text{ref}} = [\phi_{\text{ref}}, r_{\text{ref}}]$. This is achieved by applying an autoregressive scheme

$$\begin{aligned} \phi_{\text{ref}}^{+} &= \beta_{\phi t}\phi_{\text{ref}}^{-} + (1-\beta_{\phi t})\phi_{Et}, \\ r_{\text{ref}}^{+} &= \beta_{rt} r_{\text{ref}}^{-} + (1-\beta_{rt}) r_{Et}, \end{aligned} \tag{8}$$

where the subscripts $(\cdot)^{-}$ and $(\cdot)^{+}$, respectively, denote the reference model prior and after the adaptation. The variables $\beta_{\phi t}$ and β_{rt} are the current adaptation intensities

$$\begin{aligned} \beta_{\phi t} &\propto p(G_{\phi}(\hat{\mathbf{v}}_t, \mathbf{v}_{Et}), 0|\theta), \\ \beta_{rt} &\propto p(0, G_{r}(\hat{\mathbf{v}}_t, \mathbf{v}_{Et})|\theta), \end{aligned} \tag{9}$$

such that $\beta_{\phi t} \in [0,1]$, $\beta_{rt} \in [0,1]$, and $p(\cdot,\cdot|\theta)$ is defined in (7). If the region E does not contain any valid Shi-Tomasi features, then $\mathbf{v}_{Et}$ is undefined and the reference is not adapted.

From (9) it follows that the reference local-motion model is adapted to local changes in the target's motion only when the *velocity*, with which the tracker predicts the target is moving, is approximately in agreement with the observed local-motion at the current estimated state. Otherwise the adaptation is low, since the target is probably being occluded by another object.

4. Local-motion-based probabilistic tracking

We derive the combined color/local-motion-based tracker by extending a color-based particle filter to account for the local-motion. As a reference tracker we have used the closed-world tracker from [27, 28], where the target is modelled by an ellipse and color histograms are used to encode the color cues. In our case, a standard discrete-time-counterpart of the nearly-constant-velocity (NCV) dynamic model [29] with independent noise on horizontal and vertical directions was used on the target's position, and a random-walk (RW) model [29] was used on the target's size. Thus the reference tracker, we denote it by $\mathbf{T}_{\text{ref}}$, was conceptually a color-based bootstrap particle filter with the target state defined as $\mathbf{x}_t = [x_t, v_{xt}, y_t, v_{yt}, a_t, b_t]^T$, where $[x_t, y_t]$, $[v_{xt}, v_{yt}]$, $[a_t, b_t]$ are the target's position, velocity and size, respectively. The state-transition model $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ was thus a composition of the NCV and RW models, defined by the relation

$$\begin{gathered} \mathbf{x}_t = F\mathbf{x}_{t-1} + G\mathbf{w}_t, \qquad (10) \\ F = \text{diag}[\tilde{F}, \tilde{F}, 1, 1], \quad G = \text{diag}[\tilde{G}, \tilde{G}, 1, 1], \\ \tilde{F} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \qquad \tilde{G} = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}, \end{gathered}$$

where $\mathbf{w}_t$ is a discrete-time white noise sequence defined by a zero-mean normal distribution, $\mathbf{w}_t \sim \mathcal{N}(0, \Lambda)$, with covariance matrix $\Lambda = \text{diag}[\sigma_{xy}^2, \sigma_{xy}^2, \sigma_H^2, \sigma_H^2]$. The parameter σ_H corresponds to the noise in the random-walk models on the target's size. As in [28], we set this parameter in all our subsequent experiments such that the size of the target does not change between two time-steps by more than 15%. The parameter σ_{xy} corresponds to the noise in the nearly-constant-velocity models on the target's position. This parameter varies between experiments since it largely depends on the target's size (in pixels) and the volatility of its motion.

We define the observation vector $\mathbf{y}_t$ as a concatenation of the color ($\mathbf{y}_{t\mathrm{col}}$) and motion ($\mathbf{y}_{t\mathrm{mot}}$) observations, e.g., $\mathbf{y}_t = [\mathbf{y}_{t\mathrm{col}}, \mathbf{y}_{t\mathrm{mot}}]$. Under the assumption that the target's color properties are independent of its motion, the likelihood function for the particle filter can be written as

$$p(\mathbf{y}_t|\mathbf{x}_t) = p(\mathbf{y}_{t\mathrm{col}}, \mathbf{y}_{t\mathrm{mot}}|\mathbf{x}_t)p(\mathbf{y}_{t\mathrm{mot}}|\mathbf{x}_t) = p(\mathbf{y}_{t\mathrm{col}}|\mathbf{x}_t)p(\mathbf{y}_{t\mathrm{mot}}|\mathbf{x}_t), \tag{11}$$

where $p(\mathbf{y}_{t\mathrm{col}}|\mathbf{x}_t)$ is the color likelihood at state $\mathbf{x}_t$, and $p(\mathbf{y}_{t\mathrm{mot}}|\mathbf{x}_t)$ presents the local-motion likelihood at that state. Note that, in the case of the color-based tracker $\mathbf{T}_{\mathrm{ref}}$, the likelihood function is equal to $p(\mathbf{y}_{t\mathrm{col}}|\mathbf{x}_t)$. The combined color/local-motion-based tracker, we denote it by $\mathbf{T}_{\mathrm{com}}$, is then obtained by replacing the likelihood function in $\mathbf{T}_{\mathrm{ref}}$ by (11) and setting

$$p(\mathbf{y}_{t\mathrm{mot}}|\mathbf{x}_t) = p(G_\phi(\mathbf{v}_{\mathbf{x}_t}, \mathbf{v}_{\mathrm{ref}}), G_r(\mathbf{v}_{\mathbf{x}_t}, \mathbf{v}_{\mathrm{ref}})|\theta). \tag{12}$$

In the equation above, $p(\cdot, \cdot|\theta)$ is defined in (7), $\mathbf{v}_{\mathbf{x}_t}$ is the local-motion (4) sampled at state $\mathbf{x}_t$, and $\mathbf{v}_{\mathrm{ref}}$ is the reference local-motion. While for a hypothesized state $\mathbf{x}_t^{(i)}$ the color histograms are sampled within the elliptical region associated with that state, in practice we found it sufficient to sample the local-motion feature (4) within a rectangular region superimposed over the ellipse. The proposed combined color/local-motion-based probabilistic tracker $\mathbf{T}_{\mathrm{com}}$ is summarized in Algorithm 1.

5. Experimental study

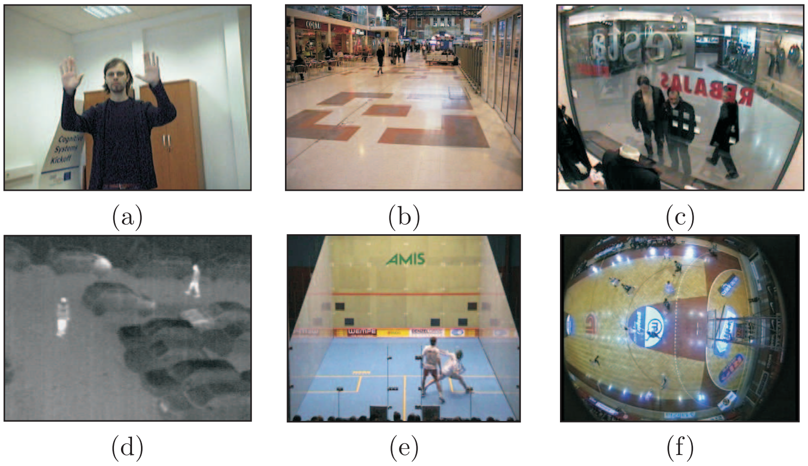
The trackers $\mathbf{T}_{\mathrm{com}}$ and $\mathbf{T}_{\mathrm{ref}}$ (section 4) were compared on experiments from hand tracking, surveillance, and sports tracking (Figure 3) to demonstrate how we can use the proposed local-motion model to improve intensity-related trackers. In each experiment, a single target was manually selected in the first frame and tracked throughout the recording. All recordings were taken at the frame rate of 25 frames/s, except for the recording used for hand tracking, which was taken at 30 frames/s. The Shi-Tomasi feature detection from section 3 was performed using 3 × 3 pixels neighborhoods and only features whose smallest eigenvalue exceeded $\xi_{\mathrm{th}} = 10^{-3}$ were accepted. The size of the integration window in the Lucas-Kanade optical flow calculation was set to 9 × 9 pixels. The amplitude threshold used in (5) and (6) was set to $\delta_{\mathrm{th}} = 10^{-2}$ pixels. In all the experiments, except for the experiment with the hand tracking, a single-level pyramid was used to calculate the optical flow. The scale of motion in the experiment with the hand tracking was larger than in the other experiments and the optical flow could not be estimated well enough by a single-level pyramid.

Algorithm 1 The combined color/local-motion-based probabilistic tracker.

```
Initialize:
  Initialize the tracker by selecting the target (e.g., manually).
Tracking: For time-step $k = 1, 2, 3, \dots$
  Execute an iteration of the particle filter using the likelihood function
  $p(\mathbf{y}_t|\mathbf{x}_t)$ defined in (11) and the current reference local-motion $\mathbf{v}_{\text{ref}}$:
    • Start from the $N$ particles which approximate the posterior from the
      previous time-step: $p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \approx \{\mathbf{x}_{t-1}^{(i)}, \pi_{t-1}^{(i)}\}_{i=1:N}$
    • Sample with replacement $N$ particles,
      $\tilde{\mathbf{x}}_{t-1}^{(i)} \sim p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})$,
      and propagate them using the dynamic model (10),
      $\mathbf{x}_t^{(i)} \sim p(\mathbf{x}_t|\hat{\mathbf{x}}_{t-1}^{(i)})$.
    • Extract the local-motion feature (4) $\mathbf{v}_{\mathbf{x}_t^{(i)}}$ at each predicted state $\mathbf{x}_t^{(i)}$.
    • Recalculate the weights of all $N$ particles using (11):
      $\tilde{\pi}_t^{(i)} = p(\mathbf{y}_{t\text{col}}|\mathbf{x}_t) p(G_\phi(\mathbf{v}_{\mathbf{x}_t}, \mathbf{v}_{\text{ref}}), G_r(\mathbf{v}_{\mathbf{x}_t}, \mathbf{v}_{\text{ref}})|\theta)$
    • Normalize the weights $\pi_t^{(i)} = \frac{\tilde{\pi}_t^{(i)}}{\sum_{i=1}^N \tilde{\pi}_t^{(i)}}$.
  Estimate the current MMSE state (3) $\hat{\mathbf{x}}_t$ and the current velocity $\hat{\mathbf{v}}_t$.
  Estimate the new reference $\mathbf{v}_{\text{ref}}$ according to section 3.2.
  Update the color-based features (e.g., [28]).
```

Therefore, to compensate for the larger scale of motion, a two-level pyramid was used instead. The parameters of the local-motion likelihood function (11) were set experimentally to $\lambda_\phi = 0.1$, $\lambda_r = 0.3$ and $w_{\text{noise}} = 0.01$. Note that, since λ_r was chosen to be larger than λ_ϕ, the amplitude of the local motion had a smaller impact on the value of the likelihood function in comparison to the angle. The reasoning behind this is that during accelerated movement, typical for hands and people, the amplitude of the optical flow changes more significantly than its direction. In our implementation we use a standard discrete-time counterpart of the nearly-constant-velocity dynamic model which requires specification of the variance, σ_{xy}^2, of the noise acting on the target's velocity. We have set this parameter to $\sigma_{xy} = 1$ pixel in x and y direction for all experiments except for the case of hand tracking, where the noise of $\sigma_{xy} = 3$ pixels was used. The number of particles in the particle filter was set to $N = 50$, and all other parameters were set as in [28]. The parameters were kept constant throughout the experiments. For the videos demonstrating the results presented in this paper, please see http://vicos.fri.uni-lj.si/data/matejk/pr08/index.htm.

In the experiment with hand tracking, a recording of a person waving his hands was used (Figure 3a). Both hands were approximately 20×20 pixels large, and were tracked five times independently of each other. The hands occluded



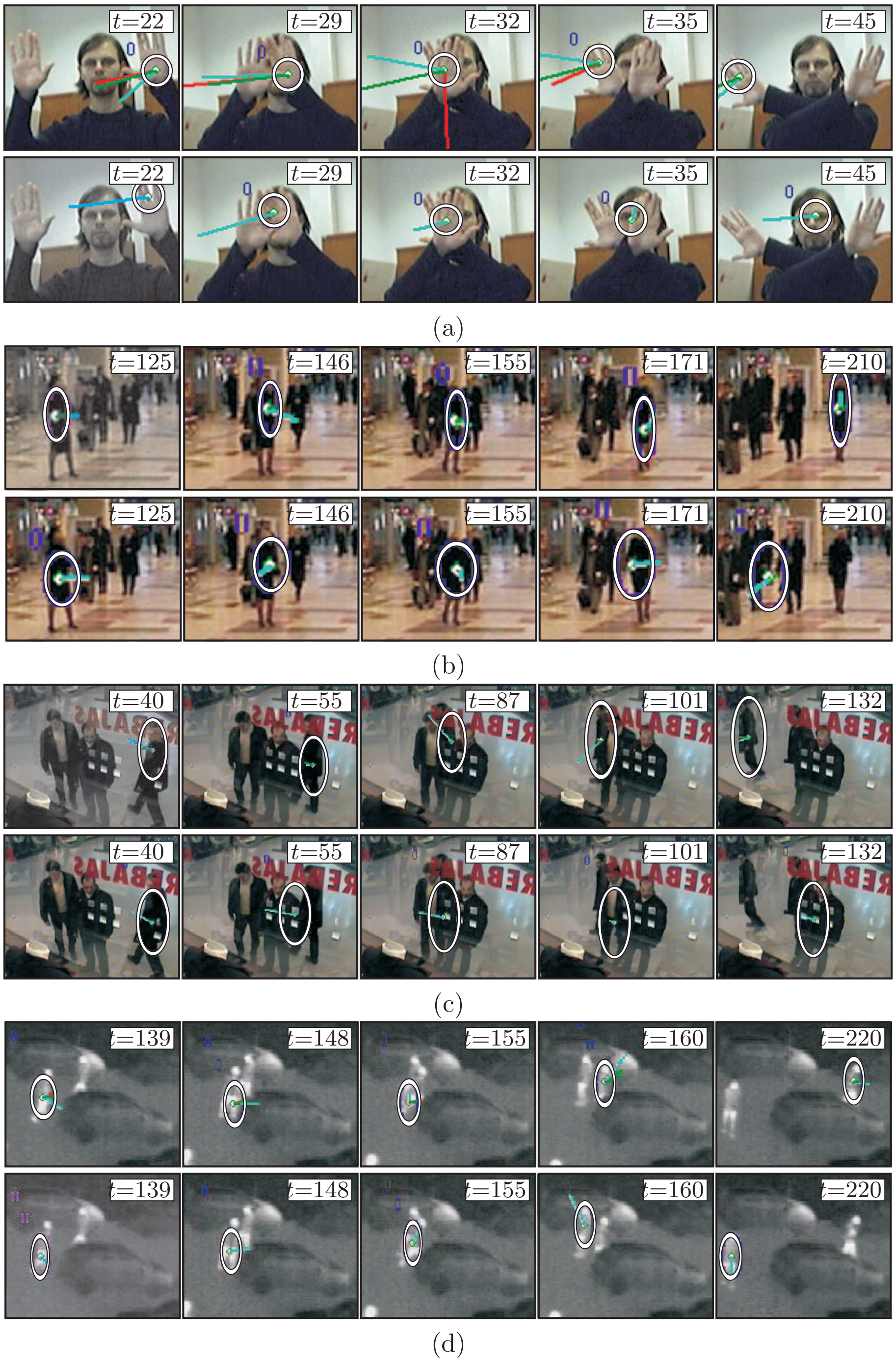
(a) (b) (c)

(d) (e) (f)

Figure 3: Images from the recordings used in the experiments with hand tracking (a), surveillance (b,c,d) and sports tracking (e,f).

each other 17 times with majority of occlusions occurring in front of the person's face. The reference tracker $\mathbf{T}_{\mathrm{ref}}$ failed on average 24 times, by either following the wrong hand or the face after the hands were crossed. The proposed combined tracker $\mathbf{T}_{\mathrm{com}}$ resolved a majority of occlusions, and failed only four times by losing the hand and locking onto the person's face. A detailed inspection of the results showed that, in those situations where $\mathbf{T}_{\mathrm{com}}$ failed, the target's color model was strongly favoring the face, while the local-motion feature at the edge of the tracked hand sufficiently supported the target's reference motion. The reference motion model deteriorated, which caused the tracker to drift from the hand to the face. Figure 4a shows an example where $\mathbf{T}_{\mathrm{ref}}$ lost the hand after it was occluded by another hand, while $\mathbf{T}_{\mathrm{com}}$ resolved the occlusion. Note that there were situations in which $\mathbf{T}_{\mathrm{ref}}$ lost the tracked hand even though it was not occluded, but was merely moving close to the other hand or the face; $\mathbf{T}_{\mathrm{com}}$ was still able to resolve all of these situations. The resolution of the occlusions can be completely attributed to the nature of the local-motion model and its adaptation scheme: In cases when the target gets occluded by a differently moving object, the observed local motion contradicts the predicted motion from the tracker's dynamic model, which stops the adaptation of the reference local-motion model. Assuming that the target does not significantly change its motion during the occlusion, the tracker can recover the target when it reappears. However, the tracker is prone to fail in situations when the motion-consistency assumption is violated. We discuss such situation in a later example of tracking a player of squash.

Three recordings were used to demonstrate the performance of the proposed model on a surveillance application. In the first recording, taken from PETS 2006 database [30] (Figure 3b), a person walking in front of a group of visually similar persons was tracked. The size of the person in the video



(a)

(b)

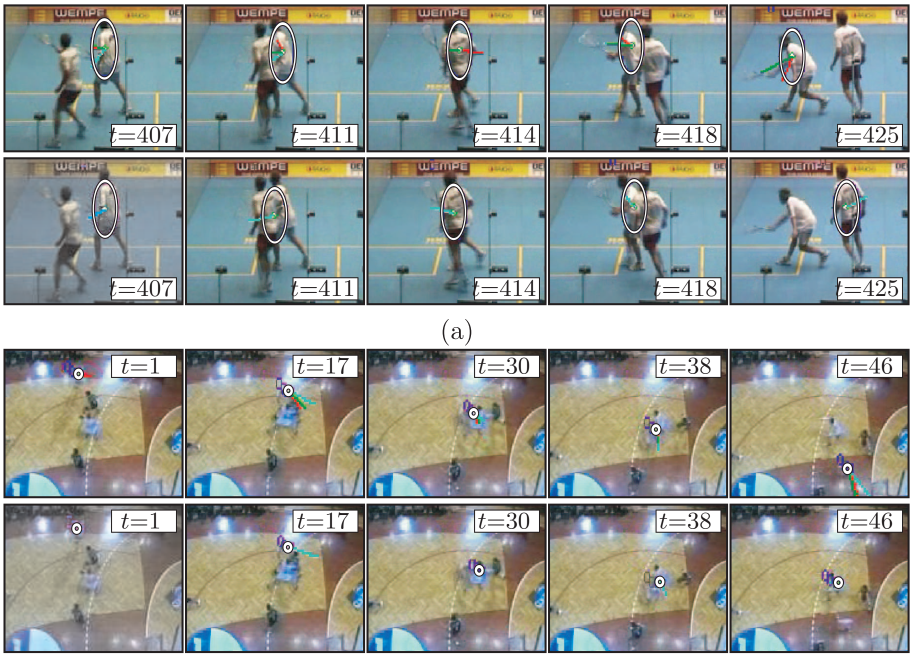
(c)

(d)

Figure 4: Frames from the experiments with hand tracking (a) and surveillance (b,c,d). The upper rows show results for tracking with the proposed combined tracker $\mathbf{T}_{\text{com}}$, and the lower rows show results for the reference tracker $\mathbf{T}_{\text{ref}}$.

was approximately 13×30 pixels. The results of the experiment are shown in Figure 4b. Due to the visual similarity, the purely color-based tracker $\mathbf{T}_{\mathrm{ref}}$ could not discern the tracked person from the group, even though the person was walking *in front* of the group, and tracking failed. On the other hand, the proposed combined tracker $\mathbf{T}_{\mathrm{com}}$ was again able to make use of the motion, and successfully tracked the person throughout the recording. The second recording was from PETS 2002 database [30] (Figure 3c) and contained a 30×80 pixels large person slowly walking behind two other visually similar persons. This recording was used to demonstrate the ability of the proposed tracker $\mathbf{T}_{\mathrm{com}}$ to track objects even under lasting partial occlusion. Figure 4c shows the results, where $\mathbf{T}_{\mathrm{ref}}$ loses the tracked person in occlusion. Again, the proposed tracker $\mathbf{T}_{\mathrm{com}}$ successfully tracks the person throughout the entire sequence. These results are comparable to the results of the surveillance system [31] that was also able to track the same person throughout the occlusion. However, that system applied sophisticated appearance models, tracked all objects in the scene, and used heuristics to explicitly handle occlusions between the targets. In contrast, our method used a very simple color model and did not apply any occlusion handling. However, it still successfully tracked the person through the occlusion solely due to the proposed local-motion model. The third recording was a thermographic (FLIR) video of a parking lot and contained a 10×20 pixels large person which was occluded by another person (Figure 3d). Since the persons could not be distinguished solely by their appearance, the reference tracker $\mathbf{T}_{\mathrm{ref}}$ failed to track the correct person after the occlusion (Figure 4d). $\mathbf{T}_{\mathrm{com}}$ was able to utilize the motion information and successfully tracked the person even after the occlusion.

Two experiments were used to demonstrate the performance of the proposed visual model with tracking in sports. In the first experiment we have tracked a player of squash (Figure 3e) five times to evaluate how the proposed local-motion model behaves in situations where the target does not have a simple motion model and rapidly changes its direction. The player was approximately 25×45 pixels large and was occluded 14 times by another visually similar player. The reference tracker $\mathbf{T}_{\mathrm{ref}}$ failed on average twelve times, while $\mathbf{T}_{\mathrm{com}}$ failed on average three times thus significantly improving the tracking performance. Figure 5a shows five frames from the recording where, after the occluded player appears ($t = 418$), the visual information becomes ambiguous, since both players wear white t-shirts, and $\mathbf{T}_{\mathrm{ref}}$ fails ($t = 425$). On the other hand, $\mathbf{T}_{\mathrm{com}}$ successfully utilizes the local-motion information to resolve this ambiguity, and tracks the correct player even after the occlusion. Note that during tracking there were many situations in which the player radically changed the direction of motion and instantly contradicted the tracker's reference motion model. In those cases the uniform component in (7) assigned a small motion probability to all particles. When the player was not occluded, the color model was able to localize him, thus recovering the position and the velocity from the tracker. Since the reference motion is adapted only when the estimated velocity agrees with the observed motion, the local-motion model was able to quickly adapt and continue with tracking. In a few situations the player slowed down or changed



(a)

(b)

Figure 5: Frames from the experiments with sports tracking. The upper rows show results for tracking with the proposed combined tracker $\mathbf{T}_{\text{com}}$, and the lower rows show results for the reference tracker $\mathbf{T}_{\text{ref}}$.

direction of motion while he was occluded by the other player. In those situations, the motion and the color information were both ambiguous, the tracker degraded to a color-based tracker, and lost the player. To demonstrate how the proposed visual model can help tracking through near collisions, we have tracked a white, 9×9 pixels large basketball player (Figure 3f) which nearly collided with another white player. As the tracked player approached the other white player, $\mathbf{T}_{\text{ref}}$ could not resolve the visual ambiguity and failed (Figure 5b). On the other hand, the $\mathbf{T}_{\text{col}}$ successfully tracked the player throughout the contact. For a better overview of the performance of the trackers we summarize their failure rates for the separate experiments in the third and fourth column of the Table 1.

One drawback of the proposed tracker $\mathbf{T}_{\text{com}}$ is that it entails additional computation to calculate the optical flow features. We have therefore also recorded the processing times required for a tracking iteration in the above experiments[3]. The average execution times in, milliseconds per frame, are given in the fifth and the sixth column of the Table 1 for the $\mathbf{T}_{\text{col}}$ and $\mathbf{T}_{\text{com}}$, respectively. We see

[3] All the tests were performed on a laptop computer with 1.8GHz AMD Athlon CPU and 2GB RAM.

Table 1: Quantitative results of experiments with the purely color-based tracker $\mathbf{T}_{\text{col}}$ and the combined tracker $\mathbf{T}_{\text{com}}$. The recording names in the first column correspond to the recordings in Figure 3(a,b,c,d,e,f) in that order. The second column shows approximate object sizes in the experiments. *Failure rate* denotes the average number of times that the trackers have failed during the experiments. *Execution time*, shows the time spent for a tracking iteration, and *Execution time ratio* shows the time ratio between the fifth and the sixth column.

	object size	Failure rate		Execution time [ms]		Execution
recording	[pixels]	$\mathbf{T}_{\text{col}}$	$\mathbf{T}_{\text{com}}$	$\mathbf{T}_{\text{col}}$	$\mathbf{T}_{\text{com}}$	time ratio
Hands	20 × 20	24	4	10	17	0.6
PETS 2006	13 × 30	1	0	11	14	0.8
PETS 2002	30 × 80	1	0	22	31	0.7
FLIR	10 × 20	1	0	10	13	0.8
Squash	25 × 45	12	3	15	26	0.6
Basketball	9 × 9	1	0	8	9	0.9

that all experiments exhibited a real-time performance, exceeding 30 frames per second. The execution times varied, which is expected and can be attributed to the varying size of objects in the experiments and the noise levels in the dynamic models, which effectively determine the "search areas" in the particle filter. To quantify how much of the processing time is spent for the calculation of the flow features in comparison to the entire tracking iteration in $\mathbf{T}_{\text{com}}$, we have used the following rationale. We can assume that most of the processing time of a single tracking iteration of $\mathbf{T}_{\text{col}}$ and $\mathbf{T}_{\text{com}}$ is spent on extraction, comparison and calculation of the color and motion features. The execution time of $\mathbf{T}_{\text{col}}$ thus roughly measures the time spent for processing the color features, whereas the execution time of $\mathbf{T}_{\text{com}}$ then measures the processing of the color as well as the local-motion features. By dividing the execution times of $\mathbf{T}_{\text{col}}$ by $\mathbf{T}_{\text{com}}$, we can approximately evaluate what portion of time is spent for operations other than calculation of flow features. This is shown in the last column of the Table 1. From that column we can say that, on average, the tracking iteration of the $\mathbf{T}_{\text{com}}$ spent approximately 70% of the processing time for calculation with color features, while the remaining 30% was spent on the processing of the motion features.

6. Conclusion

A novel method to track visually similar objects through (near) occlusion is presented. Discrimination between the visually similar objects is achieved by deriving a novel probabilistic local-motion model, which is calculated from the optical flow induced by the objects. We show how this model can be probabilistically combined with a color cue within the framework of particle filters into a combined color/local-motion-based tracker. Examples from hand tracking, surveillance, and sports tracking have shown that the local-motion model significantly improves tracking when the target is occluded by, or moves in front of, a visually similar object. The experiments have also shown that the proposed model deals well with situations when the target is rapidly changing its motion. While the proposed local-motion feature improves tracking when the object's motion is pronounced, the improvement diminishes when the object slows down, since the discrimination power of motion also decreases. We have observed such a behavior in an experiment of tracking a squash player. When the player slowed down to a stop, and was located close to a visually-similar player, the players could not be distinguished by motion well enough and the tracking failed. This motivates the combination of the proposed local-motion feature with other features than the color to improve tracking in these situations. A drawback of the local-motion feature is, of course, that it entails additional computational load on the tracking iteration. In our experiments, however, the processing of the local-motion still required only a small portion of the tracking iteration (on average 30%), and allowed real-time tracking with at least thirty frames per second.

Since the proposed local-motion model can help resolve ambiguities associated with multiple visually similar targets, it can be used in existing probabilistic multi-cue integration frameworks like [32, 8, 16], or as extension to multiple-target tracking schemes, such as [31], to increase their robustness when tracking visually-similar targets. Note also that the local-motion-based feature is general enough to be used not only within the framework of particle filters, but also with non-stochastic methods: For example, the discrimination-based trackers such as the recently proposed AdaBoost tracker [33] or the level-set-based blob trackers like [34, 35]. In particular, an appealing property of the AdaBoost tracker [33] and the level-set-based tracker [35] is that they report a realtime operation and allow for a straight-forward inclusion of the local-motion feature proposed in this paper. These considerations will be the focus of future work.

Acknowledgement

We would like to thank the editor and the reviewers for their valuable comments during the review process, which improved the content of this paper. Research program P2-0214 (RS), M3-0233 project PDR sponsored by the ministry of defense of republic of Slovenia, EU FP6-004250-IP project CoSy, and EU FP7-ICT215181-IP project CogX.

References

[1] A. Doucet, N. de Freitas, N. Gordon (Eds.), Sequential Monte Carlo Methods in Practice, New York: Springer-Verlag, 2001. 1

[2] M. Isard, A. Blake, Condensation – conditional density propagation for visual tracking, Int. J. Comput. Vision 29 (1) (1998) 5–28. 1, 4

[3] P. Pérez, C. Hue, J. Vermaak, M. Gangnet, Color-based probabilistic tracking, in: Proc. European Conf. Computer Vision, Vol. 1, 2002, pp. 661–675. 1

[4] S. T. Birchfield, S. Rangarajan, Spatiograms versus histograms for region-based tracking, in: Proc. Conf. Comp. Vis. Pattern Recognition, Vol. 2, 2005, pp. 1158–1163. 1

[5] W. L. Lu, J. J. Little, Tracking and recognizing actions at a distance, in: Proc. Workshop on Compter Vision Based Analisys in Sport Environments In conjunction with ECCV06, 2006, pp. 49–60. 2

[6] C. Yang, R. Duraiswami, L. Davis, Fast multiple object tracking via a hierarchical particle filter, in: Proc. Int. Conf. Computer Vision, 2005, pp. 212 – 219. 2

[7] T. Mathes, J. H. Piater, Robust non-rigid object tracking using point distribution models, in: Ann. Symp. German Association Patt. Recogn., 2006, pp. 515–524. 2

[8] P. Brasnett, L. Mihaylova, D. Bull, N. Canagarajah, Sequential Monte Carlo tracking by fusing multiple cues in video sequences, Image and Vision Computing 25 (8) (2007) 1217–1227. 2, 15

[9] M. Isard, A. Blake, Icondensation: Unifying low-level and high-level tracking in a stochastic framework, Lecture Notes in Computer Science 1406 (1998) 893–908. 2

[10] P. Li, F. Chaumette, Image cues fusion for object tracking based on particle filter, in: Intl. Conf. Articulated Motion And Deformable Objects, 2004, pp. 99–107. 2

[11] B. Stenger, A. Thayananthan, P. H. S. Torr, R. Cipolla, Model-based hand tracking using a hierarchical bayesian filter, IEEE Trans. Pattern Anal. Mach. Intell. 28 (9) (2006) 1372– 1384. 2

[12] H. Wang, D. Suter, K. Schindler, C. Shen, Adaptive object tracking based on an effective appearance filter, IEEE Trans. Pattern Anal. Mach. Intell. 29 (9) (2007) 1661–1667. 2

[13] P. Viola, M. J. Jones, D. Snow, Detecting pedestrians using patterns of motion and appearance, in: Proc. Int. Conf. Computer Vision, Vol. 2, 2003, pp. 734–741. 2

[14] P. Pérez, J. Vermaak, A. Blake, Data fusion for visual tracking with particles, Proc. of the IEEE 92 (3) (2004) 495–513. 2

[15] D. W. Scott, Multivariate Density Estimation, New York: Wiley, 1992. 2, 6

[16] W. Du, J. Piater, A probabilistic approach to integrating multiple cues in visual tracking, in: 10th European Conference on Computer Vision, 2008. 2, 15

[17] W. Du, J. H. Piater, Tracking by cluster analysis of feature points using a mixture particle filter, in: Advanced Video and Signal Based Surveillance, 2005, pp. 165– 170. 2

[18] B. D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: Imaging Understanding Workshop, 1981, pp. 121–130. 2, 5

[19] S. Pundlik, S. Birchfield, Motion segmentation at any speed, in: Proc. British Machine Vision Conference, Vol. I, 2006, pp. 427–436. 2

[20] G. J. Brostow, R. Cipolla, Unsupervised bayesian detection of independent motion in crowds, in: Proc. Conf. Comp. Vis. Pattern Recognition, 2006, pp. I: 594–601. 2

[21] A. Bugeau, P. Pérez, Detection and segmentation of moving objects in highly dynamic scenes, in: Proc. Conf. Comp. Vis. Pattern Recognition, 2007, pp. 1 – 8. 3

[22] K. Smith, D. Gatica-Perez, J. M. Odobez, Using particles to track varying numbers of interacting people, in: Proc. Conf. Comp. Vis. Pattern Recognition, Vol. 1, 2005, pp. 962–969. 4

[23] M. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-gaussian Bayesian tracking, IEEE Trans. Signal Proc. 50 (2) (2002) 174–188. 4

[24] N. J. Gordon, D. J. Salmond, A. F. M. Smith, Novel approach to nonlinear/non-gaussian Bayesian state estimation, in: IEE Proc. Radar and Signal Processing, Vol. 40, 1993, pp. 107–113. 4

[25] J. Y. Bouguet, Pyramidal implementation of the lucas kanade feature tracker: Description of the algorithm, Tech. rep., Intel Corporation, Microprocessor Research Labs, http://www.intel.com/research/mrl/research/opencv/ http://www.intel.com/research/mrl/research/opencv/ (last visit: 2007). 5

[26] J. Shi, C. Tomasi, Good features to track, in: Proc. Conf. Comp. Vis. Pattern Recognition, 1994, pp. 593 – 600. 5

[27] M. Kristan, J. Perš, M. Perše, S. Kovačič, Towards fast and efficient methods for tracking players in sports, in: ECCV Workshop on Computer Vision Based Analysis in Sport Environments, 2006, pp. 14–25. 7

[28] M. Kristan, J. Perš, M. Perše, S. Kovačič, Closed-world tracking of multiple interacting targets for indoor-sports applications, Comput. Vision Image Understanding In press. `doi:DOI:10.1016/j.cviu.2008.01.009`.
URL `http://www.sciencedirect.com/science/article/B6WCX-4S4JYW4-1/2/5608ea6fb41768ef4b5b`
7, 8, 9

[29] X. Rong Li, V. Jilkov P., Survey of maneuvering target tracking: Dynamic models, IEEE Trans. Aerospace and Electronic Systems 39 (4) (2003) 1333–1363. 7

[30] PETS: Performance Evaluation of Tracking and Surveillance, Online database, http://www.cvg.rdg.ac.uk/slides/pets.html, last visited: 4.4.2007 (2006). 10, 12

[31] A. Senior, Tracking people with probabilistic appearance models, in: Perf. Eval. Track. and Surveillance in conjunction with ECCV02, 2002, pp. 48–55. 12, 15

[32] I. Leichter, M. Lindenbaum, E. Rivlin, A probabilistic framework for combining tracking algorithms, in: Proc. Conf. Comp. Vis. Pattern Recognition, Vol. II, 2004, pp. 445–451. 15

[33] H. Grabner, M. Grabner, H. Bischof, Real-time tracking via on-line boosting, in: Proc. British Machine Vision Conference, 2006, pp. 47–56. 15

[34] A. X. L. Yilmaz, M. Shah, Contour-based object tracking with occlusion handling in video acquired using mobile cameras, IEEE Trans. Pattern Anal. Mach. Intell. 26 (11) (2004) 1531–1536. 15

[35] Y. Shi, W. Karl, Real-time tracking using level sets, in: Proc. Conf. Comp. Vis. Pattern Recognition, Vol. 2, 2005, pp. 34– 41. 15