Machine Vision and Applications
DOI 10.1007/s00138-008-0174-7

ORIGINAL PAPER

Particle filtering strategies for data fusion dedicated to visual tracking from a mobile robot

Ludovic Brèthes · Frédéric Lerasle · Patrick Danès · Mathias Fontmarty

Received: 7 December 2006 / Accepted: 15 September 2008
© Springer-Verlag 2008

Abstract This paper introduces data fusion strategies within particle filtering in order to track people from a single camera mounted on a mobile robot in a human environment. Various visual cues are described, relying on color, shape or motion, together with several filtering strategies taking into account all or parts of these measurements in their importance and/or measurement functions. A preliminary evaluation enables the selection of the most meaningful visual cues associations in terms of discriminative power, robustness to artifacts and time consumption. The depicted filtering strategies are then evaluated in order to check which people trackers, regarding visual cues and algorithms associations, best fulfill the requirements of the considered scenarios. The performances are compared through some quantitative and qualitative evaluations. Some associations of filtering strategies and visual cues show a significant increase in the tracking robustness and precision. Future works are finally discussed.

Keywords Monocular vision · Person tracking · Particle filtering · Data fusion · Mobile robotics

L. Brèthes · F. Lerasle (✉) · P. Danès · M. Fontmarty
LAAS-CNRS, 7 avenue du Colonel Roche,
31077 Toulouse, France
e-mail: Frederic.Lerasle@laas.fr

L. Brèthes
e-mail: Ludovic.Brethes@laas.fr

P. Danès
e-mail: Patrick.Danes@laas.fr

M. Fontmarty
e-mail: Mathias.Fontmarty@laas.fr

F. Lerasle · P. Danès
Univ. Paul Sabatier, 118 rte de Narbonne,
31062 Toulouse, France

1 Introduction

Tracking people in dynamically changing environments is a critical task over a wide range of applications, e.g. human–computer interface [8,22], teleconferencing [41], surveillance [18,19], motion capture [29], video compression [35], and driver assistance [4]. This paper focuses on mobile robotic applications, where visual tracking of people is one of the ubiquitous elementary functions. Tracking from a mobile platform is a very challenging task, which imposes several requirements. First, the sensors being embedded on the robot, they are usually moving instead of static, and have a restricted perception of the environment. Moreover, the robot is expected to evolve in a wide variety of environmental conditions. Consequently, several hypotheses must be handled simultaneously and a robust integration of multiple visual cues is required in order to achieve some robustness to artifacts. Finally, on-board computational power is limited so that only a small percentage of these resources can be allocated to tracking, the remaining part being required to enable the concurrent execution of other functions as well as decisional routines within the robot's architecture. Thus, care must be taken to design efficient algorithms.

Many 2D people tracking paradigms with a single camera have been proposed in the literature which we shall not attempt to review here. The reader is referred to [15,46] for details. One can mention Kalman filtering [35], the mean-shift technique [12] or its variant [11], tree-based filtering [36] among many others. Beside these approaches, one of the most successful paradigms, focused in this paper, undoubtedly concerns sequential Monte Carlo simulation methods, also known as particle filters [14]. The popularity of these strategies stems from their simplicity, ease of implementation, and modeling flexibility over a wide variety of applications. They seem well-suited to visual tracking as they make

no assumption on the probability distributions entailed in the characterization of the problem and enable an easy combination/fusion of diverse kind of measurements.

Particle filters represent the posterior distribution by a set of samples, or particles, with associated importance weights. This weighted particles set is first drawn from the state vector initial probability distribution, and is then updated over time taking into account the measurements and a prior knowledge on the system dynamics and observation models.

In the Computer Vision community, the formalism has been pioneered in the seminal paper [20] by Isard and Blake, which coins the term CONDENSATION. In this scheme, the particles are drawn from the dynamics and weighted by their likelihood w.r.t. the measurement. CONDENSATION is shown to outperform Kalman filter in the presence of background clutter.

Following the CONDENSATION algorithm, various improvements and extensions have been proposed for visual tracking. Isard et al. in [22] introduce a mixed-state CONDENSATION tracker in order to perform multiple model tracking. The same authors propose in [21] another extension, named ICONDENSATION, which has introduced for the first time importance sampling in visual tracking. It constitutes a mathematically principled way of directing search, combining the dynamics and measurements. So, the tracker can take advantage of the distinct qualities of the information sources and re-initialize automatically when temporary failures occur. Particle filtering with history sampling is proposed as a variant in [37]. Rui and Chen in [34] introduce the Unscented Particle Filter (UPF) into audio and visual tracking. The UPF uses the Unscented Kalman filter to generate proposal distributions that seamlessly integrate the current observation. Partitioned sampling, introduced by MacCormick and Isard in [27], is another way of applying particle filters to tracking problems with high-dimensional configuration spaces. This algorithm is shown to be well-suited to track articulated objects [28]. The hierarchical strategy [33] constitutes a generalization. Last, though outside the scope of this paper, particle filters have also become a popular tool to perform simultaneous tracking of multiple persons [27,42].

As mentioned before, the literature proposes numerous particle filtering algorithms, yet a few studies comparing the efficiency of these filtering strategies have been carried out. When doing so, the associated results are mainly compared against those of the original CONDENSATION approach [26,34,37].

Another observation concerns data fusion. It can be argued that data fusion using particle filtering schemes has been fairly seldom exploited within this visual tracking context. The numerous visual trackers referred to in the literature consider a single cue, i.e. contours [20,28,34] or color [30,32]. The multiple cues association has often been confined to contours and color [8,21,37,47], or color and motion [6,10,33,43].

This data fusion problem has been extensively tackled by Pérez et al. in [33]. The authors propose a hierarchical particle filtering algorithm, which successively takes account of the measurements so as to efficiently draw the particles. To our belief, using multiple cues simultaneously, both in the importance and measurement functions, not only allows to use complementary and redundant information but also enables a more robust failures detection and recovery. More globally, other existing particle filtering strategies should also be evaluated in order to check which ones best fulfill the requirements for the envisaged application.

From these considerations, a first contribution of this paper relates to visual data fusion in robotics scenarios covering a wide variety of environmental conditions. A large spectrum of plausible multi-cues association for such a context is depicted. Evaluations are then performed in order to exhibit the most meaningful visual cues associations in terms of discriminative power, robustness to artifacts and time consumption, be these cues involved in the particle filter importance or measurement functions. A second contribution concerns a thorough comparison of the various particle filtering strategies for data fusion dedicated to the applications envisaged here. Some experiments are presented, in which the designed trackers efficiency is evaluated with respect to temporary target occlusion, presence of significant clutter, as well as large variations in the target appearance and in the illumination of the environment. These trackers have been integrated on a tour-guide robot named Rackham whose role is to help people attending an exhibition.

The paper is organized as follows. Section 2 describes Rackham and outlines the embedded visual trackers. Section 3 sums up the well-known particle filtering formalism, and reviews some variants which enable data fusion for tracking. Then, Sect. 4 specifies some visual measurements which rely on the shape, color or image motion of the observed target. A study comparing the efficiency of various particle filtering strategies is carried out in Sect. 5. Section 6 reports on the implementation of these modalities on Rackham. Last, Sect. 7 summarizes our contribution and puts forward some future extensions.

2 Rackham and the tour-guide scenario progress

Rackham is an iRobot B21r mobile platform whose standard equipment has been extended with one pan-tilt camera EVI-D70 dedicated to H/R interaction, one digital camera for robot localization, one ELO touch-screen, a pair of loudspeakers, an optical fiber gyroscope and wireless Ethernet (Fig. 1).

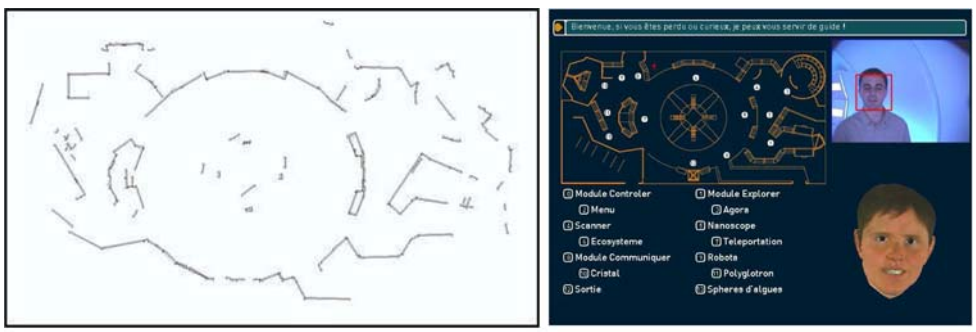
Fig. 1 Rackham

Rackham has been endowed with functions enabling it to act as a tour-guide robot. So, it embeds robust and efficient basic navigation abilities in human-crowded environments. For instance, Fig. 2a shows the laser map of an exhibition, which the robot first builds automatically during an exploration phase with no visitor and then uses for localization. Besides, our efforts have concerned the design of visual functions in order to track, recognize and interact with visitors attending an exhibition. Figure 2b reports the robot's interface display gathering the outputs from such visual functions (top right) together with other interaction facilities: selection of exhibition areas (top left, down left), human-like clone (down right), etc.

When Rackham is left alone with no mission, it tries to find out people whom he could interact with, a behavior hereafter called "search for interaction". As soon as a visitor comes into its neighborhood, it introduces itself, tries to identify his/her face and explains how to use its services thanks to the touch-screen. More precisely, if the interlocutor is unknown, the robot opens a face learning session, then asks him/her to define the mission. On the contrary, when interacting with a formerly identified user, the robot can suggest missions/services which are complementary to the ones executed in the past. Once the robot and its tutor have agreed on an area to visit, Rackham plans and displays its trajectory, prior to inviting its user to follow. While navigating, the robot keeps on giving information about the progress of the ongoing path and verifies the user presence. Whenever the guided visitor leaves during the execution of this "guidance mission", the robot detects this and stops. If, after a few seconds, this user is not re-identified, the robot restarts a "search for interaction" session. Otherwise, when a known user is re-identified, the robot proposes him/her to continue the ongoing "guidance mission".

The design of visual modalities has been undertaken within this demonstration scenario. Three basic tracking modalities, focused in this paper, have been outlined which the robot must basically deal with:

1. *The search for interaction*, where the robot, static and left alone, visually tracks visitors thanks to the camera mounted on its helmet, in order to heckle them when they enter the exhibition (Fig. 3a). This modality involves the whole human body tracking at long H/R distances (>3 m);
2. *The proximal interaction*, where a user can interact through the ELO touch-screen, to select the area he/she wants to visit (Fig. 3b); during this interaction, the robot remains static and must keep, thanks to the camera materializing its eye, the visual contact with its tutor's face at short H/R distances (<1 m);
3. *The guidance mission,* where the robot drives the visitor to the selected area; during its mission, the robot must also maintain the interaction with the guided visitor (Fig. 3c). This modality involves the upper human body tracking at medium H/R distances.



(a) (b)

Fig. 2 Interface display (a), SICK laser map of the exhibition (b)

Fig. 3 The robot visual modalities: **a** search for interaction, **b** proximal interaction, **c** guidance mission

These trackers involves the camera EVI-D70 whose characteristics are: image resolution 320×240 pixels, retina dimension $1/2''$, and focal length 4.1mm.

3 Particle filtering algorithms for data fusion

3.1 A generic algorithm

Particle filters are sequential Monte Carlo simulation methods to the state vector estimation of any Markovian dynamic system subject to possibly non-Gaussian random inputs [3,13,14]. Their aim is to recursively approximate the posterior probability density function (pdf) $p(x_k|z_{1:k})$ of the state vector x_k at time k conditioned on the set of measurements $z_{1:k} = z_1, \ldots, z_k$. A linear point-mass combination

$$p(x_k|z_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta\left(x_k - x_k^{(i)}\right), \quad \sum_{i=1}^{N} w_k^{(i)} = 1, \qquad (1)$$

is determined—with $\delta(\cdot)$ the Dirac distribution—which expresses the selection of a value—or "particle"—$x_k^{(i)}$ with probability—or "weight"—$w_k^{(i)}$, $i = 1, \ldots, N$. An approximation of the conditional expectation of any function of x_k, such as the minimum mean square error (MMSE) estimate $\mathrm{E}_{p(x_k|z_{1:k})}[x_k]$, then follows.

Let the system be fully described by the prior $p(x_0)$, the dynamics pdf $p(x_k|x_{k-1})$ and the observation pdf $p(z_k|x_k)$. The generic particle filtering algorithm—or "Sampling Importance Resampling" (SIR)—is shown in Table 1. Its initialization consists in an independent identically distributed (i.i.d.) sequence drawn from $p(x_0)$. At each further time k, the particles keep evolving stochastically, being sampled from an *importance function* $q(x_k|x_{k-1}^{(i)}, z_k)$ which aims at adaptively exploring "relevant" areas of the state space. They are then suitably weighted so as to guarantee the consistency of the approximation (1). To this end, step 5 affects each particle $x_k^{(i)}$ a weight $w_k^{(i)}$ involving its *likelihood* $p(z_k|x_k^{(i)})$ w.r.t. the measurement z_k as well as the values at $x_k^{(i)}$ of the importance function and dynamics pdf.

In order to limit the degeneracy phenomenon, which says that whatever the sequential Monte Carlo simulation method, after few instants all but one particle weights tend to zero, step 8 inserts a resampling stage, e.g. the so-called "systematic resampling" defined in [25]. There, the particles associated with high weights are duplicated while the others collapse, so that the resulting sequence $x_k^{(s^{(1)})}, \ldots, x_k^{(s^{(N)})}$ is i.i.d. according to $\sum_{i=1}^{N} w_k^{(i)} \delta(x_k - x_k^{(i)})$. Note that this resampling stage should rather be fired only when the filter efficiency—related to the number of "useful" particles—goes beneath a predefined threshold [14].

3.2 Importance sampling from either dynamics or measurements: basic strategies

The CONDENSATION—for "Conditional Density Propagation" [20]—can be viewed as the instance of the SIR algorithm in which the particles are drawn according to the system dynamics, viz. when $q(x_k|x_{k-1}^{(i)}, z_k) = p(x_k|x_{k-1}^{(i)})$. This endows CONDENSATION with a prediction-update structure, in that $\sum_{i=1}^{N} w_{k-1}^{(i)} \delta(x_k - x_k^{(i)})$ approximates the prior $p(x_k|z_{1:k-1})$. The weighting stage becomes $w_k^{(i)} \propto w_{k-1}^{(i)} p(z_k|x_k^{(i)})$. In the visual tracking context, the original algorithm [20] defines the particles likelihoods from contour primitives, yet other visual cues have also been exploited [33].

Resampling by itself cannot efficiently limit the degeneracy phenomenon. In addition, it may lead to a loss of diversity in the state space exploration. The importance function must thus be defined with special care.

In visual tracking, the modes of the likelihoods $p(z_k|x_k)$, though multiple, are generally pronounced. As CONDENSATION draws the particles $x_k^{(i)}$ from the system dynamics but "blindly" w.r.t. the measurement z_k, many of these may well be assigned a low likelihood $p(z_k|x_k^{(i)})$ and thus a low weight in step 5, significantly worsening the overall filter performance. An alternative, henceforth labeled "Measurement-based SIR" (MSIR), merely consists in sampling the particles at time k—or just some of their entries—according to an importance function $\pi(x_k|z_k)$ defined from the current image. The first MSIR strategy was ICONDENSATION [21], which guides the state space exploration by a color blobs detector. Other visual detection functionalities can be used as well, e.g. face detector (Sect. 4), or any other intermittent primitive which, despite its sporadicity, is very discriminant when present [33]: motion, sound, etc.

In an MSIR scheme, a particle $x_k^{(i)}$ whose entries are drawn from the current image may be inconsistent with its predecessor $x_{k-1}^{(i)}$ from the point of view of the state dynamics. As expected, the smaller is the value $p(x_k^{(i)}|x_{k-1}^{(i)})$, the lesser

Table 1 Generic particle filtering algorithm (SIR)

$[\{x_k^{(i)}, w_k^{(i)}\}]_{i=1}^N = \mathrm{SIR}([\{x_{k-1}^{(i)}, w_{k-1}^{(i)},\}]_{i=1}^N, z_k)$

1: **IF** $k = 0$, **THEN** Draw $x_0^{(1)}, \ldots, x_0^{(i)}, \ldots, x_0^{(N)}$ i.i.d. according to $p(x_0)$, and set $w_0^{(i)} = \frac{1}{N}$ **END IF**
2: **IF** $k \geq 1$ **THEN** $\{$—$[\{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}]_{i=1}^N$ being a particle description of $p(x_{k-1}|z_{1:k-1})$—$\}$
3: **FOR** $i = 1, \ldots, N$, **DO**
4: "Propagate" the particle $x_{k-1}^{(i)}$ by independently sampling $x_k^{(i)} \sim q(x_k|x_{k-1}^{(i)}, z_k)$
5: Update the weight $w_k^{(i)}$ associated with $x_k^{(i)}$ according to $w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(z_k|x_k^{(i)})p(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{k-1}^{(i)}, z_k)}$, prior to a normalization step s.t. $\sum_i w_k^{(i)} = 1$
6: **END FOR**
7: Compute the conditional mean of any function of x_k, *e.g.* the MMSE estimate $\mathrm{E}_{p(x_k|z_{1:k})}[x_k]$, from the approximation $\sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)})$ of the posterior $p(x_k|z_{1:k})$
8: At any time or depending on an "efficiency" criterion, resample the description $[\{x_k^{(i)}, w_k^{(i)}\}]_{i=1}^N$ of $p(x_k|z_{1:k})$ into the equivalent evenly weighted particles set $[\{x_k^{(s^{(i)})}, \frac{1}{N}\}]_{i=1}^N$, by sampling in $\{1, \ldots, N\}$ the indexes $s^{(1)}, \ldots, s^{(N)}$ according to $P(s^{(i)} = j) = w_k^{(j)}$; set $x_k^{(i)}$ and $w_k^{(i)}$ with $x_k^{(s^{(i)})}$ and $\frac{1}{N}$
9: **END IF**

is the weight $w_k^{(i)}$. One solution to this problem, as proposed in the genuine ICONDENSATION algorithm, consists in sampling some of the particles from the dynamics and some w.r.t. the prior, so that the importance function reads as, with $\alpha, \beta \in [0; 1]$

$$q\left(x_k|x_{k-1}^{(i)}, z_k\right) = \alpha\pi(x_k|z_k) + \beta p\left(x_k|x_{k-1}^{(i)}\right) + (1 - \alpha - \beta)p_0(x_k). \quad (2)$$

This combination enables the tracker to benefit from the distinct qualities of the information sources and to re-initialize automatically when temporary failures occur.

3.3 Towards the "optimal" case: the auxiliary particle filter

It can be shown [14] that the "optimal" recursive scheme, i.e. which best limits the degeneracy phenomenon, must define $q^*(x_k|x_{k-1}^{(i)}, z_k) \triangleq p(x_k|x_{k-1}^{(i)}, z_k)$ and thus $w_k^{*(i)} \propto w_{k-1}^{*(i)} p(z_k|x_{k-1}^{(i)})$ in the SIR algorithm (Table 1). Each weight $w_k^{*(i)}$ can then be computed before drawing $x_k^{(i)}$. So, the overall efficiency can be enhanced by resampling the weighted particle set $[\{x_{k-1}^{(i)}, w_k^{*(i)}\}]_{i=1}^N$, which in fact represents the smoother pdf $p(x_{k-1}|z_{1:k})$, just before its "propagation" through the optimal importance function $q^*(x_k|x_{k-1}^{(i)}, z_k)$.

Despite such an algorithm can be seldom implemented exactly, it can be mimicked by the "Auxiliary Particle Filter" (AUXILIARY_PF) along the lines of [31], see Table 2. Let the importance function $\pi(x_k|x_{k-1}^{(i)}, z_k) = p(x_k|x_{k-1}^{(i)})$ be defined in place of $q^*(x_k|x_{k-1}^{(i)}, z_k)$, and $\hat{p}(z_k|x_{k-1}^{(i)})$ be an approximation of the *predictive likelihood* $p(z_k|x_{k-1}^{(i)})$ (steps 3–5); for instance, one can set $\hat{p}(z_k|x_{k-1}^{(i)}) = p(z_k|\mu_k^{(i)})$, where $\mu_k^{(i)}$ characterizes the distribution of x_k conditioned on $x_{k-1}^{(i)}$, e.g. $\mu_k^{(i)} = \mathrm{E}_{p(x_k|x_{k-1}^{(i)})}[x_k]$ or $\mu_k^{(i)} \sim p(x_k|x_{k-1}^{(i)})$. First, an *auxiliary weight* $\lambda_k^{(i)} \propto w_{k-1}^{(i)} \hat{p}(z_k|x_{k-1}^{(i)})$ is associated with each particle $x_{k-1}^{(i)}$. The approximation $[\{x_{k-1}^{(i)}, \lambda_k^{(i)}\}]_{i=1}^N$ of $p(x_{k-1}|z_{1:k})$ is then resampled into $[\{x_{k-1}^{(s^{(i)})}, \frac{1}{N}\}]_{i=1}^N$ (step 6), prior to its propagation until time k through $\pi(x_k|x_{k-1}^{(s^{(i)})}, z_k)$ (step 8). Finally, the weights of the resulting particles $x_k^{(i)}$ must be corrected (step 9) in order to take account of the "distance" between $\lambda_k^{(i)}$ and $w_k^{*(i)}$, as well as of the dissimilarity between the selected and optimal importance functions $\pi(x_k^{(i)}|x_{k-1}^{(s^{(i)})}, z_k)$ and $p(x_k^{(i)}|x_{k-1}^{(s^{(i)})}, z_k)$.

The particles cloud can thus be steered towards relevant areas of the state space. In the visual tracking context, the approximate predictive likelihood can rely on distinct visual cues from these involved in the computation of the "final-stage" likelihoods $p(z_k|x_k^{(i)})$. The main limitation of the AUXILIARY_PF algorithm is its bad performance when the dynamics is uninformative compared to the state final-stage likelihood, e.g. when the dynamics is very noisy or when the observation density has sharp modes. Therefore, $\mu_k^{(i)}$ being a bad characterization of $p(x_k|x_{k-1}^{(i)})$, the pdf $p(x_{k-1}|z_{1:k})$ of the smoother is poorly approximated by $[\{x_{k-1}^{(i)}, \lambda_k^{(i)}\}]_{i=1}^N$. The resampling stage in step 6 of Table 2 may well eliminate some particles which, once propagated through the dynamics, would be very likely w.r.t. z_k. At the same time, other particles may well be duplicated which, after the prediction step, come to lie in the final-stage likelihood tails.

In the framework of auxiliary particle filters, the Unscented Transform [24] can constitute a way to define a better approximation $\hat{p}(z_k|x_{k-1})$ of the predictive likelihood $p(z_k|x_{k-1})$, which is the basis of the auxiliary resampling stage, see steps 3–6 of Table 2. As is the case in the Unscented Particle Filter [39], this transform can also be entailed in the association to each particle of the Gaussian near-optimal importance function from which it is sampled. Andrieu et al. propose such a strategy in [2]. Nevertheless, despite

Table 2 Auxiliary particle filter (AUXILIARY_PF)

$[\{x_k^{(i)}, w_k^{(i)}\}]_{i=1}^N = \text{AUXILIARY\_PF}([\{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}]_{i=1}^N, z_k)$

1: **IF** $k = 0$, **THEN** Draw $x_0^{(1)}, \ldots, x_0^{(i)}, \ldots, x_0^{(N)}$ i.i.d. according to $p(x_0)$, and set $w_0^{(i)} = \frac{1}{N}$ **END IF**
2: **IF** $k \geq 1$ **THEN** {—$[\{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}]_{i=1}^N$ being a particle description of $p(x_{k-1}|z_{1:k-1})$—}
3: **FOR** $i = 1, \ldots, N$, **DO**
4: From the approximation $\hat{p}(z_k|x_{k-1}^{(i)}) = p(z_k|\mu_k^{(i)})$ –*e.g.* with $\mu_k^{(i)} \sim p(x_k|x_{k-1}^{(i)})$ or $\mu_k^{(i)} = \mathrm{E}_{p(x_k|x_{k-1}^{(i)})}[x_k]$–, compute the auxiliary weights $\lambda_k^{(i)} \propto w_{k-1}^{(i)} \hat{p}(z_k|x_{k-1}^{(i)})$, prior to a normalization step s.t. $\sum_i \lambda_k^{(i)} = 1$
5: **END FOR**
6: Resample $[\{x_{k-1}^{(i)}, \lambda_k^{(i)}\}]_{i=1}^N$ –or, equivalently, sample in $\{1, \ldots, N\}$ the indexes $s^{(1)}, \ldots, s^{(N)}$ of the particles at time $k-1$ according to $P(s^{(i)} = j) = \lambda_k^{(j)}$– in order to get $[\{x_{k-1}^{(s^{(i)})}, \frac{1}{N}\}]_{i=1}^N$; both $\sum_{i=1}^N \lambda_k^{(i)} \delta(x_{k-1} - x_{k-1}^{(i)})$ and $\frac{1}{N}\sum_{i=1}^N \delta(x_{k-1} - x_{k-1}^{(s^{(i)})})$ mimic $p(x_{k-1}|z_{1:k})$
7: **FOR** $i = 1, \ldots, N$, **DO**
8: "Propagate" the particles by independently drawing $x_k^{(i)} \sim p(x_k|x_{k-1}^{(s^{(i)})})$
9: Update the weights, prior to their normalization, by setting $w_k^{(i)} \propto \frac{p(z_k|x_k^{(i)}) p(x_k^{(i)}|x_{k-1}^{(s^{(i)})})}{\hat{p}(z_k|x_{k-1}^{(s^{(i)})}) \pi(x_k^{(i)}|x_{k-1}^{(s^{(i)})}, z_k)} = \frac{p(z_k|x_k^{(i)})}{\hat{p}(z_k|x_{k-1}^{(s^{(i)})})} = \frac{p(z_k|x_k^{(i)})}{p(z_k|\mu_k^{(s^{(i)})})}$
10: Compute $\mathrm{E}_{p(x_k|z_{1:k})}[x_k]$ from the approximation $\sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)})$ of the posterior $p(x_k|z_{1:k})$
11: **END FOR**
12: **END IF**

its attractiveness and its ability to mimic the optimal case, this is more difficult to implement and shows a higher computational cost.

3.4 Other strategies suited to visual tracking

3.4.1 History sampling

Several interesting particle filtering alternatives to visual tracking are proposed in [37]. One of them considers dynamic models of order greater than or equal to 2, in which the state vector has the form $x_k = (u_k', v_k', h_k')'$, with $[.]'$ the transpose operator. The subvector $(u_k', v_k')'$ or "innovation part"—of x_k obeys a stochastic state equation on x_{k-1}, while h_k—called "history part"—is a deterministic function $f(x_{k-1})$. It is assumed that the innovations $(u_k^{(i)\prime}, v_k^{(i)\prime})'$ are sampled from an importance function such as $q_I(u_k, v_k|x_{k-1}^{(i)}, z_k) = \pi(u_k|z_k) p(v_k|u_k^{(i)}, x_{k-1}^{(i)})$, i.e. the subparticles $u_k^{(i)}$ are positioned from the measurement only while the $v_k^{(i)}$'s are drawn by fusing the state dynamics with the knowledge of $u_k^{(i)}$—and that the pdf of the measurement conditioned on the state satisfies $p(z_k|x_k) = p(z_k|u_k, v_k)$. This context is particularly well-suited to visual tracking, for state-space representations of linear AR models entail the above decomposition of the state vector, and because the output equation does not involve its "history part".

The authors define procedures enabling the avoidance of any contradiction between $(u_k^{(i)\prime}, v_k^{(i)\prime})'$ and its past $x_{k-1}^{(i)}$. Their "Rao-Blackwellized Subspace SIR with History Sampling" (RBSSHSSIR) is summarized in Table 3. Its step 5 consists, for each subparticle $u_k^{(i)}$ drawn from $\pi(u_k|z_k)$, in the resampling of a predecessor particle—and thus of the "history part" of $x_k^{(i)}$—which is at the same time likely w.r.t. $u_k^{(i)}$ from the dynamics point of view and assigned with a significant weight. The RBSSHSSIR algorithm noticeably differs from ICONDENSATION precisely because of this stage, yet necessary lest the weighted particles $[\{x_k^{(i)}, w_k^{(i)}\}]_{i=1}^N$ is not a consistent description of the posterior $p(x_k|z_{1:k})$.

An original proof of the RBSSHSSIR algorithm is sketched in [9], using arguments similar to these underlying the AUXILIARY_PF. It is shown that the algorithm applies even when the state process is of the first order, by just suppressing the entry $f(x_{k-1})$ from x_k.

3.4.2 Partitioned and hierarchical sampling

Partitioned and Hierarchical sampling can significantly enhance the efficiency of a particle filter in cases when the system dynamics comes as the successive application of elementary dynamics, provided that intermediate likelihoods can be defined on the state vector after applying each partial evolution model. The classical single-stage sampling of the full state space is then replaced by a layered sampling approach: thanks to a succession of sampling operations followed by resamplings based on the intermediate likelihoods, the search can be guided so that each sampling stage refines the output from the previous stage.

To outline the technical aspects of each strategy, let $\xi_0 = x_{k-1}, \xi_1, \ldots, \xi_{M-1}, \xi_M = x_k$ be $M+1$ "auxiliary vectors" such that the dynamics $p(x_k|x_{k-1})$ reads as the convolution

$$p(x_k|x_{k-1}) = \int \tilde{d}_M(\xi_M|\xi_{M-1}) \ldots \tilde{d}_1(\xi_1|\xi_0)\, \mathrm{d}\xi_1 \ldots \mathrm{d}\xi_{M-1}, \qquad (3)$$

Table 3 Rao-Blackwellized subspace particle filter with history sampling (RBSSHSSIR)

$[\{x_k^{(i)}, w_k^{(i)}\}]_{i=1}^N = \text{RBSSHSSIR}([\{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}]_{i=1}^N, z_k)$

IF $k = 0$, **THEN** Draw $x_0^{(1)}, \ldots, x_0^{(i)}, \ldots, x_0^{(N)}$ i.i.d. according to $p(x_0)$, and set $w_0^{(i)} = \frac{1}{N}$ **END IF**
IF $k \geq 1$ **THEN** $\{$—$[\{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}]_{i=1}^N$ being a particle description of $p(x_{k-1}|z_{1:k-1})$—$\}$
  FOR $i = 1, \ldots, N$, **DO**
    Draw $u_k^{(i)} \sim \pi(u_k|z_k)$
    Sample in $\{1, \ldots, N\}$ the index $I_k^{(i)}$ of the predecessor particle of $u_k^{(i)}$ according to the weights $(w_{k-1}^{(1)} p(u_k^{(i)}|x_{k-1}^{(1)}), \ldots, w_{k-1}^{(.)} p(u_k^{(i)}|x_{k-1}^{(.)}), \ldots, w_{k-1}^{(N)} p(u_k^{(i)}|x_{k-1}^{(N)}))$, *i.e.* according to $P(I_k^{(i)} = j) = \frac{w_{k-1}^{(j)} p(u_k^{(i)}|x_{k-1}^{(j)})}{\sum_{l=1}^N w_{k-1}^{(l)} p(u_k^{(i)}|x_{k-1}^{(l)})}$
    Draw $v_k^{(i)} \sim p(v_k|u_k^{(i)}, x_{k-1}^{(I_k^{(i)})})$
    Set $x_k^{(i)} = \left(u_k^{(i)\prime}, v_k^{(i)\prime}, f(x_{k-1}^{(I_k^{(i)})}) \right)'$
    Update the weights, prior to their normalization, by setting $w_k^{(i)} \propto \frac{p(z_k|u_k^{(i)}) \sum_{l=1}^N w_{k-1}^{(l)} p(u_k^{(i)}|x_{k-1}^{(l)})}{\pi(u_k^{(i)}|z_k)}$
    Compute the conditional mean of any function of x_k, *e.g.* the MMSE estimate $\text{E}_{p(x_k|z_{1:k})}[x_k]$, from the approximation $\sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)})$ of the posterior $p(x_k|z_{1:k})$
  END FOR
END IF

i.e. the successive application of $\tilde{d}_1(\xi_1|\xi_0), \ldots, \tilde{d}_m(\xi_m|\xi_{m-1}), \ldots, \tilde{d}_M(\xi_M|\xi_{M-1})$. The measurement pdf $p(z_k|x_k)$ is supposed to factorize as $p(z_k|x_k) = \prod_{m=1}^M p_m(z_k|x_k)$.

The second partitioned particle filtering algorithm proposed in [28] assumes that the state vector dynamics are component-wise independent, i.e. if all the vectors ξ_m, $m = 1, \ldots, M$, are analogously partitioned into M subvectors $\xi_m^1, \ldots, \xi_m^M$, then $\tilde{d}_m(\xi_m|\xi_{m-1}) = p(\xi_m^m|\xi_{m-1}^m) \prod_{r \neq m} \delta(\xi_m^r - \xi_{m-1}^r)$ holds for all $m = 1, \ldots, M$ so that $p(x_k|x_{k-1}) = \prod_{m=1}^M p(x_k^m|x_{k-1}^m)$. In addition, the intermediate likelihoods $p_m(z_k|x_k)$ are supposed to concern a subset of the state vector all the more important as $m \longrightarrow M$, i.e. to have the form $p_m(z_k|x_k) = l_m(z_k|x_k^1, \ldots, x_k^m)$. Under these hypotheses, the partitioned particle filter follows the algorithm outlined in Table 4, with $\tilde{q}_m(\xi_m|\xi_{m-1}, z_k) = \tilde{d}_m(\xi_m|\xi_{m-1})$, $p_m(z_k|x_k) = l_m(z_k|x_k^1, \ldots, x_k^m)$.

Partitioned sampling has been successfully applied to the visual tracking of an open kinematic chain in [28], by organizing the state vector so that its first entries depict the top elements of the chain—to be accurately positioned sooner for a higher efficiency—while its last components are related to the extremities. A branched algorithm has also proved to be able to track multiple persons in [27].

The hierarchical particle filter developed in [33] can be viewed as a generalization of the partitioned scheme outlined above. No restriction is imposed on the functions $\tilde{d}_1(\cdot|\cdot), \tilde{d}_M(\cdot|\cdot)$. The measurement z_k is supposed made up with M sensory information $z_k^1, \ldots, z_k^M$ conditionally independent given x_k, so that the intermediate likelihoods come as $p_m(z_k|x_k) = p_m(z_k^m|x_k)$. Importantly, the particles relative to the auxiliary vectors $\xi_1, \ldots, \xi_M$ are not sampled from $\tilde{d}_1(\cdot|\cdot), \ldots, \tilde{d}_M(\cdot|\cdot)$ but instead from distributions $\tilde{q}_1(\cdot|\cdot), \ldots, \tilde{q}_M(\cdot|\cdot)$ related to the importance function $q(x_k|x_{k-1}, z_k)$ by $q(x_k|x_{k-1}, z_k) = \int \tilde{q}_M(x_k|\xi_{M-1}, z_k^M) \ldots \tilde{q}_1(\xi_1|x_{k-1}, z_k^1)\, d\xi_1 \ldots d\xi_{M-1}$.

Incorporating each likelihood $p_m(z_k|\cdot)$ after applying the intermediate dynamics leads to the algorithm depicted in Table 4, with $\tilde{q}_m(\xi_m|\xi_{m-1}, z_k) = \tilde{q}_m(\xi_m|\xi_{m-1}, z_k^m)$ and $p_m(z_k|x_k) = p_m(z_k^m|x_k)$.

4 Importance and measurement functions

Importance sampling offers a mathematically principled way of directing search according to visual cues which are discriminant though possibly intermittent, e.g. motion. Such cues are logical candidates for detection modules and efficient proposal distributions. Besides, each sample weight is updated taking into account its likelihood w.r.t. the current image. This likelihood is computed by means of measurement functions, according to visual cues (e.g. color, shape) which must be persistent but may however be proner to ambiguity in cluttered scenes. In both importance sampling and weight update steps, combining or fusing multiple cues enables the tracker to better benefit from distinct information sources, and can decrease its sensitivity to temporary failures in some of the measurement processes. Measurement and importance functions are depicted in the next subsections.

4.1 Measurement functions

4.1.1 Shape cue

The use of shape cues requires that silhouette templates of human limbs have been learnt beforehand (Fig. 4). Each

Table 4 Partitioned (PARTITIONED_PF) and hierarchical (HIERARCHICAL_PF) particle filtering: $\tilde{q}_m(\xi_m|\xi_{m-1}, z_k)$ and $p_m(z_k|x_k)$ are defined either as: (PARTITIONED_PF) $\tilde{q}_m(\xi_m|\xi_{m-1}, z_k) = \tilde{d}_m(\xi_m|\xi_{m-1})$, $p_m(z_k|x_k) = l_m(z_k|x_k^1, \ldots, x_k^m)$, or: (HIERARCHI- CAL_PF) $\tilde{q}_m(\xi_m|\xi_{m-1}, z_k) = \tilde{q}_m(\xi_m|\xi_{m-1}, z_k^m)$, $p_m(z_k|x_k) = p_m(z_k^m|x_k)$

$[\{x_k^{(i)}, w_k^{(i)}\}]_{i=1}^N$= PARTITIONED_OR_HIERARCHICAL_PF $\left([\{x_{k-1}^{(i)}, w_{k-1}^{(i)}, \}]_{i=1}^N, z_k\right)$

1: **IF** $k = 0$, **THEN** Draw $x_0^{(1)}, \ldots, x_0^{(i)}, \ldots, x_0^{(N)}$ i.i.d. according to $p(x_0)$, and set $w_0^{(i)} = \frac{1}{N}$ **END IF**
2: **IF** $k \geq 1$ **THEN** —$[\{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}]_{i=1}^N$ being a particle description of $p(x_{k-1}|z_{1:k-1})$—
3: Set $\{\xi_0^{(i)}, \tau_0^{(i)}\} = \{x_{k-1}^{(i)}, w_{k-1}^{(i)}\}$
4: **FOR** $m = 1, \ldots, M$, **DO**
5: **FOR** $i = 1, \ldots, N$, **DO** Independently sample $\xi_m^{(i)} \sim \tilde{q}_m(\xi_m|\xi_{m-1}^{(i)}, z_k)$ and associate $\xi_m^{(i)}$ the weight $\tau_m^{(i)} \propto \tau_{m-1}^{(i)} \frac{p_m(z_k|\xi_m^{(i)}) \tilde{d}_m(\xi_m^{(i)}|\xi_{m-1}^{(i)})}{\tilde{q}_m(\xi_m^{(i)}|\xi_{m-1}^{(i)}, z_k)}$ **END FOR**
6: Resample $[\{\xi_m^{(i)}, \tau_m^{(i)}\}]_{i=1}^N$ into the evenly weighted particles set $[\{\xi_m^{(s^{(i)})}, \frac{1}{N}\}]_{i=1}^N$; rename $[\{\xi_m^{(s^{(i)})}, \frac{1}{N}\}]_{i=1}^N$ into $[\{\xi_m^{(i)}, \tau_m^{(i)}\}]_{i=1}^N$
7: **END FOR**
8: Set $\{x_k^{(i)}, w_k^{(i)}\} = \{\xi_M^{(i)}, \tau_M^{(i)}\}$, which is a consistent description of $p(x_k|z_{1:k})$
⋮
(…): **END IF**

particle x is classically given an edge-based likelihood $p(z^S|x)$ that depends on the sum of the squared distances between N_p points uniformly distributed along the template corresponding to x and their nearest image edges [20], i.e.

$$p\left(z^S|x\right) \propto \exp\left(-\frac{D^2}{2\sigma_s^2}\right), \quad D = \sum_{j=1}^{N_p} |x(j) - z(j)|, \tag{4}$$

where the similarity measure D involves each jth template point $x(j)$ and associated closest edge $z(j)$ in the image, the standard deviation σ_s being determined a priori.

A variant [17] consists in converting the edge image into a Distance Transform image. Interestingly, the DT is a smoother function of the model parameters. In addition, the DT image reduces the involved computations as it needs to be generated only once whatever the number of particles involved in the filter. The similarity distance D in (4) is replaced by

$$D = \sum_{j=1}^{N_p} I_{\mathrm{DT}}(j), \tag{5}$$

where $I_{\mathrm{DT}}(j)$ terms the DT image value at the j-th template point. Figure 5 plots this shape-based likelihood for an example where the target is a 2D elliptical template corresponding coarsely to the subject on the right of the input image.

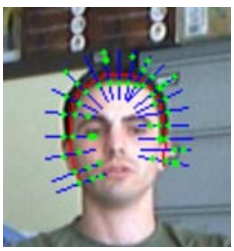


Fig. 4 Shape cue

In case of cluttered background, using only shape cues for the model-to-image fitting is not sufficiently discriminant, as multiple peaks are present.

4.1.2 Color cue

Reference color models can be associated with the targeted ROIs. These models are defined either a priori, or on-line using some automatic detection modules. Let h_{ref}^c and h_x^c two N_{bi}-bin normalized histograms in channel $c \in \{R, G, B\}$, respectively corresponding to the model and to a region B_x parametrized by the state x. The color likelihood $p(z^C|x)$ must favor candidate color histograms h_x^c close to the reference histogram h_{ref}^c. The likelihood has a form similar to (4), provided that D terms the Bhattacharyya distance [30] between the two histograms h_x^c and h_{ref}^c, i.e. for a channel c,

$$D(h_x^c, h_{\mathrm{ref}}^c) = \left(1 - \sum_{j=1}^{N_{bi}} \sqrt{h_{j,x}^c \cdot h_{j,\mathrm{ref}}^c}\right)^{1/2}. \tag{6}$$

A single histogram does not capture any information on the spatial arrangement of colors and so can lead to noticeable drift. This drift can be avoided by splitting the tracked region into sub-regions with individual reference color models. Let

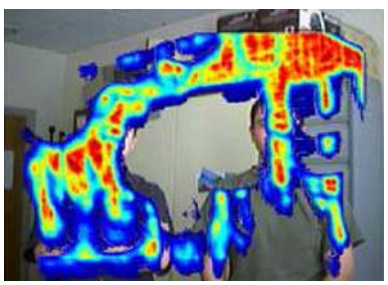


Fig. 5 Shape DT-based likelihood

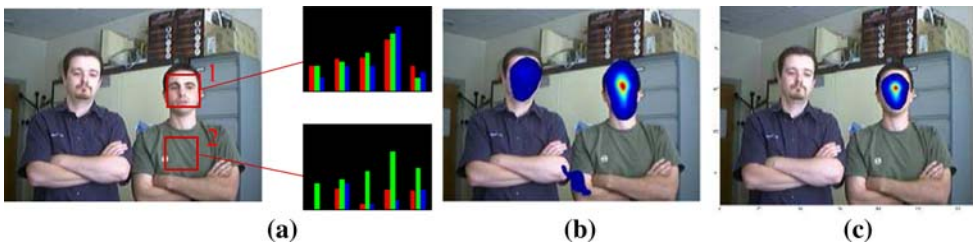


Fig. 6 **a** Color-based regions of interest (ROIs) and corresponding RGB histograms. **b**, **c** Likelihoods regarding single-part and multiple-part color models, respectively

the union $B_x = \bigcup_{p=1}^{N_R} B_{x,p}$ be associated with the set of N_R reference histograms $\{h^c_{\text{ref},p} : c \in \{\text{R, G, B}\}, \text{p} = 1, \ldots, \text{N}_\text{R}\}$. By assuming conditional independence of the color measurements, the likelihood $p(z^C|x)$ becomes

$$p(z^C|x) \propto \exp\left(-\sum_c \sum_{p=1}^{N_R} \frac{D^2(h^c_{x,p}, h^c_{\text{ref},p})}{2\sigma_c^2}\right). \tag{7}$$

Figure 6b and c plots single and multi-patch likelihoods for the above example. The ROIs corresponding to the face and clothes of the person on the right, are compared to their reference model shown in Fig. 6a.

4.1.3 Motion cue

For a static camera, a basic method consists in computing the luminance absolute difference image from successive frames. To capture motion activity, we propose to embed the frame difference information into a likelihood model similar to the one developed for the color measurements.

Pérez et al. in [33] define a reference histogram model for motion cues. For motionless regions, the measurements fall in the lower histograms bins while moving regions fall a priori in all the histograms bins. From these considerations, the reference motion histogram h^M_{ref} is given by $h^M_{j,\text{ref}} = \frac{1}{N'_{bi}}$, $j = 1, \ldots, N'_{bi}$. The motion likelihood is set to

$$p(z^M|x) \propto \exp\left(-\frac{D^2(h^M_x, h^M_{\text{ref}})}{2\sigma_m^2}\right), \tag{8}$$

and is illustrated on Fig. 7a, b, and c.

4.1.4 Multi-cues fusion

Fusing multiple cues enables the tracker to better benefit from M distinct measurements $(z^1, \ldots, z^M)$. Assuming that these are mutually independent conditioned on the state, the unified measurement function thus factorizes as

$$p(z^1, \ldots, z^M|x) = \prod_{m=1}^{M} p(z^m|x). \tag{9}$$

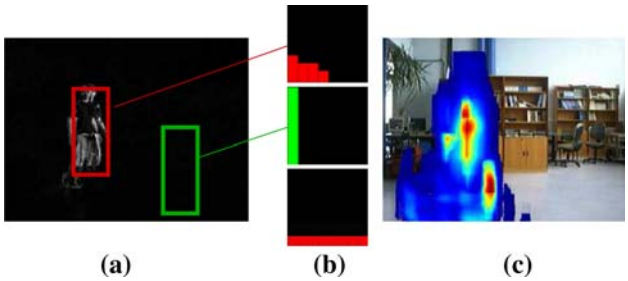


Fig. 7 **a** Absolute luminance frame difference. **b** Motion histograms of two ROIs (*top*, *middle*) and of a reference ROI (*bottom*). **c** Consequent associated likelihood

Yet, to avoid the evaluation of the likelihood for each cue, we hereafter propose some variants so as to combine multiple cues into a single likelihood model.

4.1.5 Shape and motion cues combination

Considering a static camera, it is highly possible that the targeted subject be moving, at least intermittently. To cope with background clutter, we thus favor the moving edges (if any) by combining motion and shape cues into the definition of the likelihood $p(z^S, z^M|x)$ of each particle x. Given $\overrightarrow{f}(z(j))$ the optical flow vector at pixel $z(j)$, the similarity distance D in (4) is then replaced by

$$D = \sum_{j=1}^{N_p} |x(j) - z(j)| + \rho \cdot \gamma(z(j)), \tag{10}$$

where $\gamma(z(j)) = 0$ (resp. 1) if $\overrightarrow{f}(z(j)) \neq 0$ (resp. if $\overrightarrow{f}(z(j)) = 0$) and $\rho > 0$ terms a penalty. Figure 8 plots this more discriminant likelihood function for the example seen above. The target is still the subject on the right, but is assumed to be moving.

4.1.6 Shape and color cues combination

We propose in [7] a likelihood model $p(z^S, z^C|x)$ which combines both shape and color cues through a skin-colored regions segmentation. The use of color features makes the tracker more robust to situations where there is poor grey-level contrast between the human limbs and the background.

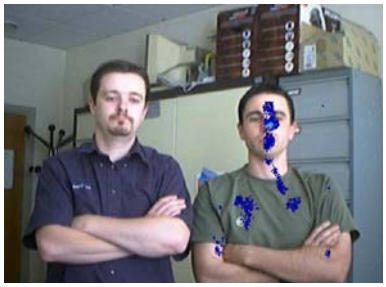


Fig. 8 Likelihood combining shape and motion cues

Numerous techniques for skin blobs segmentation are based on a skin pixel classification (see a review in [44]) as human skin colors have specific color distributions. Training images from the Compaq database [23] enable to construct a reference color histogram model [35] in a selected color space. The originality of the segmentation method [7] lies in the sequential application of two watershed algorithms, the first one being based on chromatic information and the last one relying on the intensity of the selected skin-color pixels. This second phase is useful to segment regions with similar colors but different luminance values (like hand and sleeve in Fig. 9).

A new DT image I'_{DT} is defined from the set of Canny edges points I_{DT} and from the contours of the segmented skin blobs. The latters enable to define a mask applied onto the original DT image I_{DT}. Canny edges points which are outside the mask are thus given a penalty in the DT image I'_{DT} as illustrated in Fig. 9d.

The similarity distance D in (4) is then replaced by

$$D = \sum_{j=1}^{N_p} I'_{\mathrm{DT}}(j) + \rho \cdot \gamma(j), \tag{11}$$

where $\gamma(j) = 0$ (resp. 1) if $z^{\mathrm{mask}}(j) = 1$ (resp. if $z^{\mathrm{mask}}(j) = 0$) and $\rho > 0$ terms a penalty. This strategy makes the model $p(z^S, z^C|x)$ relevant even if skin colored regions are not fully extracted or are not detected at all. Typically, overexposure (close to a bay window) or underexposure (in a corridor) make more uncertain the separation of the skin regions from background. In these situations, all the edges have the same strength in the DT image. More details on the segmentation process can be found in [7].

4.2 Importance functions

4.2.1 Shape cue

We use the face detector introduced by Viola et al. [45] which covers a range of ±45° out-of-plane rotation. It is based on a boosted cascade of Haar-like features to measure relative darkness between eyes and nose/cheek or nose bridge. Let B be the number of detected faces and $\mathbf{p}_i = (u_i, v_i)$, $i = 1, \ldots, B$, the centroid coordinate of each such region. An importance function $\pi(.)$ at location $\mathbf{x} = (u, v)$ follows, as the Gaussian mixture proposal

$$\pi(\mathbf{x}|z^S) = \sum_{i=1}^{B} \frac{1}{B} \mathcal{N}\left(\mathbf{x};\ \mathbf{p}_i, \mathrm{diag}(\sigma_{u_i}^2, \sigma_{v_i}^2)\right), \tag{12}$$

where $\mathcal{N}(\cdot;\ \mu, \Sigma)$ denotes the Gaussian distribution with mean μ and covariance Σ.

4.2.2 Color cue

Skin-colored blobs detection is performed by subsampling the input image prior to grouping the classified skin-like pixels. Then, the importance function $\pi(\mathbf{x}|z^C)$ is defined from the resulting blobs by a Gaussian mixture similar to (12).

4.2.3 Motion cue

The Bhattacharyya distance $D(h_x^M, h_{\mathrm{ref}}^M)$ to the reference motion histogram h_{ref}^M is evaluated on a subset of locations obtained by subsampling the input image and keeping the scale factor fixed. These locations are taken as the nodes of a regular grid. Nodes that satisfy $D^2(h_x^M, h_{\mathrm{ref}}^M) > \tau$ are selected. The importance function $\pi(\mathbf{x}|z^M)$ is a Gaussian mixture (12) centered on the detected locations of high motion activity. Figure 10 reports an importance function derived from the motion cues developed in Fig. 7a and b.

4.2.4 Multi-cues mixture

The importance function $\pi(\cdot)$ can be extended to consider the outputs from any of the M detectors, i.e.

$$\pi(\mathbf{x}|z^1, \ldots, z^M) = \frac{1}{M} \sum_{j=1}^{M} \pi(\mathbf{x}|z^j). \tag{13}$$

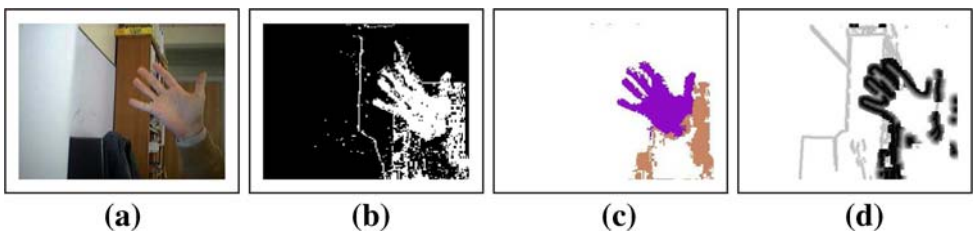


Fig. 9 **a** Input image. **b** Map of the skin color. **c** Skin blobs segmentation. **d** DT image after masking

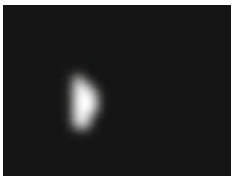


Fig. 10 Motion-based importance function

Figure 11b shows an importance function based on two detectors.

5 People tracking modalities

For our three visual modalities, the aim is to fit the *template* relative to the tracked visitor all along the video stream, through the estimation of its image coordinates (u, v), its scale factor s, as well as, if the template is shape-based, its orientation θ. All these parameters are accounted for in the state vector x_k related to the k-th frame. With regard to the dynamics model $p(x_k|x_{k-1})$, the image motions of observed people are difficult to characterize over time. This weak knowledge is thus formalized by defining the state vector as $\mathbf{x_k} = (u_k, v_k, s_k, \theta_k)^{'}$ and assuming that its entries evolve according to mutually independent random walk models, viz. $p(\mathbf{x_k}|\mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{x_k}\,;\,\mathbf{x}_{k-1}, \Sigma)$, where $\mathcal{N}(\cdot\,;\,\mu, \Sigma)$ terms the Gaussian distribution with mean μ and covariance $\Sigma = \mathrm{diag}(\sigma_u^2, \sigma_v^2, \sigma_s^2, \sigma_\theta^2)$.

5.1 Visual cues evaluation

A preliminary evaluation enables the selection of the most meaningful visual cues associations in terms of discriminative power, precision, time consumption and robustness to artifacts (e.g. clutter or illumination changes), be these cues involved in the importance or measurement functions. Results are computed from a database of over than 400 images acquired from the robot in a wide range of typical conditions. For each database image, a "ground truth" is worked out beforehand regarding the presence/absence and possibly the location of a targeted moving head. The discriminative power of a measurement (resp. importance) function is then computed by comparing the likelihood peaks (resp. the detections) locations with the "true" target location. A peak (resp. detection) in a region of interest around the target is counted as a true positive while outside peaks (resp. detections) are considered as false positives. At last, a false negative occurs when no peaks (resp. no detections) are found inside the region of interest.

Fig. 11 **a** Skin blobs (*blue*) and face (*red*) detectors. **b** Importance function mixing their outputs

5.1.1 Measurement functions

Figure 12 illustrates the average discriminative power of the measurement functions depicted in Sect. 4.1, and Fig. 13 assesses their precisions. Function $S1$ (resp. $S2$) terms the shape-based likelihood $p(z^S|\mathbf{x}_k)$ built upon the similarity distance (4) (resp. (5)). Function $C1$ is relative to the color-based likelihood $p(z_k^C|\mathbf{x}_k)$ relying on the similarity distance (6). The measurement functions $S1C1$ and $S2C1$ fuse shape and color cues by multiplying their likelihoods according to (9). Function $S2C2$ combines both shape and color cues through the skin-colored regions segmentation detailed in [7]. $S1M$ combines shape and motion cues according to (10). Finally, the functions $S1MC1$ and $MC1$ enable the fusion in (9) of all or parts of the three aforementioned cues, respectively shape, motion and color. As Fig. 13 shows, shape cues provide the best accuracy, thus it is important to privilege shape-based measurement functions.

In terms of discriminative power (Fig. 12), using only one cue in the definition of a likelihood is a bad choice. For example, the shape-based likelihoods $S1$, $S2$ are very sensitive to clutter and thus generate a high false positives rate in spite of their good true positives rates. To explain the good results of the color-based likelihood $C1$, it is important to notice that for each image of the database, a color model is computed from the true target location so that no

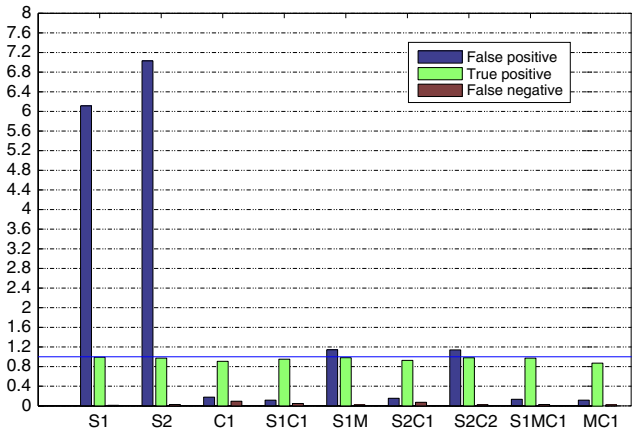


Fig. 12 Average number of detections relative to false positives, true positives and false negatives, for various likelihood functions. The *horizontal red line* depicts the mean target presence rate

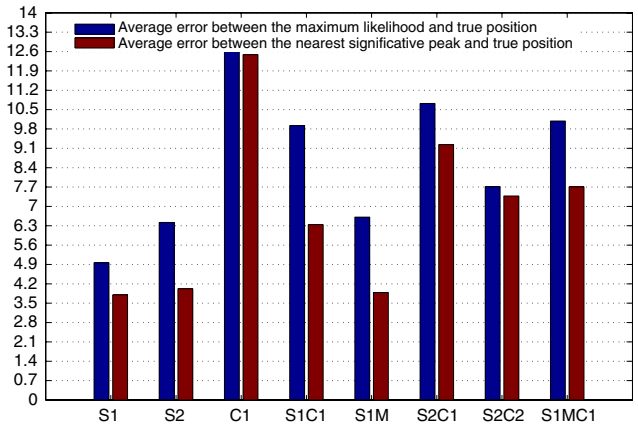


Fig. 13 Average distance between the true target position and (*1*) the maximum likelihood peak, (*2*) the nearest significative peak

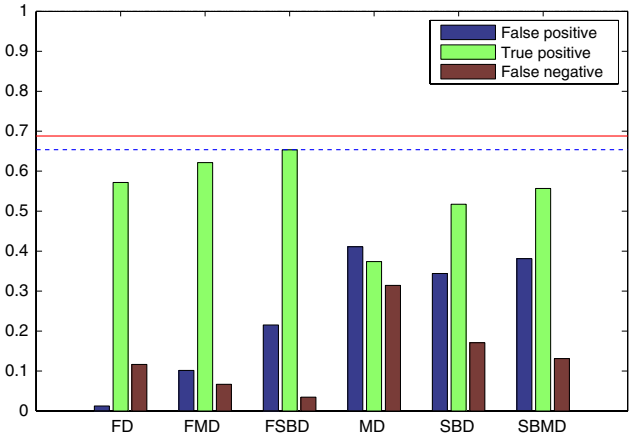


Fig. 15 Average detection rate relative to false positives, true positives and false negatives for various importance functions. The *red* and *blue* lines depict the real true positives rate and the frontal face recognition rate in the database, respectively

color model drift is taken into account in the evaluation. As expected, the more cues are involved in the definition of a likelihood, the higher is its discriminative power. For instance, the motion attribute can penalize motionless contours due to the static background. Fusing color-based likelihoods with shape-based likelihoods eliminates the influence of background contours and makes conveniently colored regions become more likely. Though the fusion strategy $S1MC1$ slightly increases the discriminative power, it is not selected because of its important time consumption. Its running time and these of the other measurement functions are illustrated in Fig. 14.

Similar arguments lead to the rejection of $S2C2$. In fact, the associations of either shape and color cues ($S1C1$, $S2C1$), shape and motion ($S1M$) or color and motion ($MC1$) show the best tradeoff between computational cost and discriminative power. These which enjoy the least time consumption, namely $S2C1$, $S1M$ and $MC1$, have been kept for future evaluations.

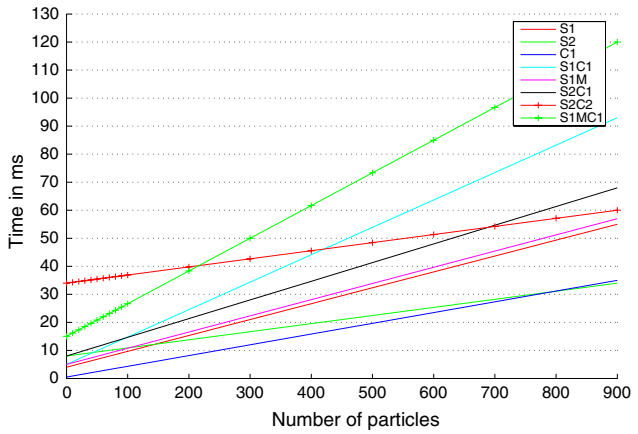


Fig. 14 Average running time per image for various likelihood functions

5.1.2 *Importance functions*

Recall that the importance functions in Sect. 4.2 are relative to face detection (yielding $\pi(\mathbf{x}_k|z_k^S)$ and denoted FD), motion detection (yielding $\pi(\mathbf{x}_k|z_k^M)$ and denoted MD) or skin blob detection (yielding $\pi(\mathbf{x}_k|z_k^C)$ and denoted SBD). The importance functions FMD, $FSBD$, $SBMD$ mix all or parts of the three aforementioned detectors thanks to (13). Figure 15 illustrates the average discriminative power for importance functions associated with a single detector or merging the outputs from several detectors.

Though the FD importance function enjoys a high detection rate and a low false positives rate, it is unfortunately restricted to frontal faces located in the H/R distances interval [0.5 m; 3 m] to permit Haar-like feature extraction. For such short and medium H/R distances[1] ($< 3m$), the multi-cues importance function $FSBD$, which associates FD and SBD into (13), clearly enlarges the spectrum of detected faces as its true positives rate is higher. Moreover, as reported in Fig. 16, the time consumption induced by SBD is negligible compared to the one of FD. The performances are significantly worse for MD, yet this detector is well-suited for long-range H/R distances[2] (>3 m) where shape and skin-color are not sufficiently reliable to use FD-based or SBD-based detectors.

In the selection of an importance function, strategies enjoying a higher true positives rate are preferred. This

[1] i.e. modalities #2 and #3 in Sect. 2.

[2] i.e. modality #1 in Sect. 2.

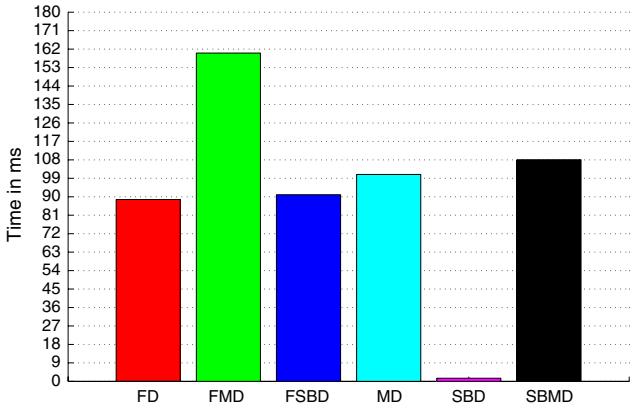


Fig. 16 Average computation time of one image for each importance function. Note that this time is independent of the number of particles involved in the tracking

ensures a sufficient number of particles to be sampled on the target, despite some may be positioned on false detections. Consequently, the importance functions $FSBD$ and MD are considered as candidate elements of the aforementioned visual tracking modalities, to be characterized and evaluated below.

5.2 Particle filtering strategies evaluations

The filtering strategies depicted in Sect. 3 must be examined in order to check which ones best fulfill the requirements of the considered H/R interaction modalities. For the sake of comparison, importance functions rely on dynamics or measurements alone (and are respectively noted DIF for "Dynamics-based Importance Function" and MIF for "Measurement-based Importance Function"), or combine both (and are termed DMIF for "Dynamics and Measurement-based Importance Function"). Further, each modality is evaluated on a database of sequences acquired from the robot in a wide range of typical conditions: cluttered environments, appearance changes or sporadic disappearance of the targeted subject, jumps in his/her dynamics, etc. For each sequence, the mean estimation error with respect to "ground truth", together with the mean failure rate (% of target loss), are computed from several filter runs and particles numbers. The error is computed as the distance (in pixels) between the estimated position and the true position of the object to be tracked. It is important to keep in mind that a failure is registered when this error exceeds a threshold (related to the region of interest), and is followed by a re-initialization of the tracker. Due to space reasons, only a subset of the associated figure plots is shown here. This analysis motivates our choices depicted hereafter for the three visual tracking modalities. The presented results have been obtained on a 3 GHz Pentium IV personal computer.

5.2.1 *Face tracker (proximal interaction)*

This modality involves the state vector $\mathbf{x}_k = (u_k, v_k, s_k, \theta_k)'$. As the robot remains static, both shape and motion cues are combined into the $S1M$ measurement function. The tracker is evaluated in nominal conditions, viz. under no disturbance, as well as against cluttered environments and illumination changes. A typical run is shown in Fig. 17. Figures 18 and 19 plot the tracking failure rate as well as tracking errors averaged over scenarios involving cluttered backgrounds. Dynamics-based Importance Functions lead to a better precision (about 10 pixels) together with a low failure rate, so that detection modules are not necessary in this "easiest" context.

The AUXILIARY_PF strategy shows an higher time consumption than the CONDENSATION algorithm though with no improvement of the approximation of the posterior. The increased computational cost is of course due to the auxiliary sampling step. The fair performance comes from the poorly informative dynamics model, see the end of Sect. 3.3. This is why we opt for a CONDENSATION algorithm, which can run at ≈ 40 Hz for $N = 200$ particles (Fig. 20).

The parameters values of our face tracker are listed in Table 5. The standard deviations of the Gaussian noises entailed in the random walk dynamics are set using classical arguments relating them to the admissible evolution of the template between two consecutive images. The standard deviations of the importance functions come from an offline prior study of the underlying detectors, as was done in [21]. The parameter σ_s involved in the shape-based likelihood (and, in Tables 6, 7, the parameters σ_c, σ_m involved in the color-based and motion-based likelihoods), are defined as follows: first, for each element of an image database, a "ground truth" is determined by manually adjusting the template position, orientation and scale; then the distances involved in the various likelihoods are computed for several perturbations of the template parameters around their "true" values; assuming that these distances are samples of a Gaussian centered on 0 enables the estimation of $\sigma_s, \sigma_c, \sigma_m$. The remaining coefficients, including the number of particles, are selected by trial-and-error, so as to ensure overall good performance of the tracker while limiting its computational cost.

5.2.2 *Upper human body tracker (guidance mission)*

This modality involves the state vector $\mathbf{x}_k = (u_k, v_k, s_k)'$—the orientation θ_k being set to a known constant—as well as two color models $h^c_{\text{ref},1}$,$h^c_{\text{ref},2}$, respectively corresponding to the head and the torso of the guided person, in the measurement function (7). To overcome appearance changes of these ROIs in the video stream, their associated color models are

Fig. 17 Tracking scenario with CONDENSATION over a cluttered background. The *red* template depicts the MMSE estimate

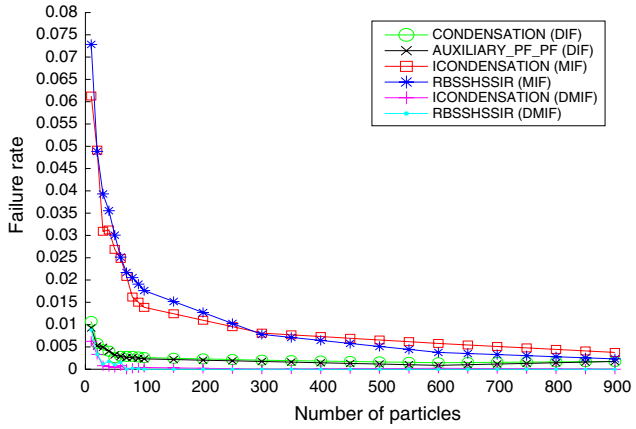


Fig. 18 Average failure rate versus number of particles on sequences involving cluttered background

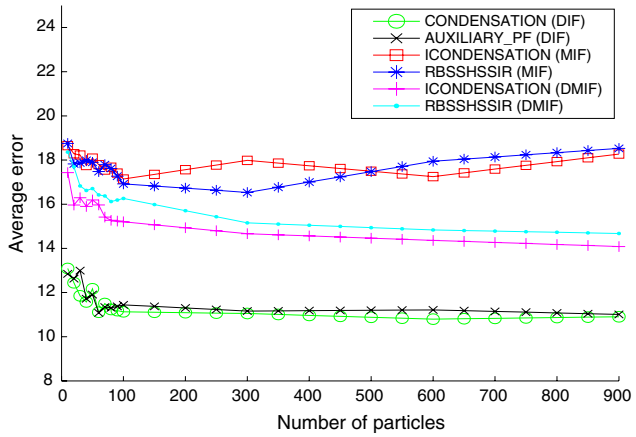


Fig. 19 Tracking errors versus number of particles on sequences involving cluttered background

updated online through a first-order discrete-time filtering process entailing the state estimate i.e.

$$h^c_{\text{ref},k} = (1-\kappa)\cdot h^c_{\text{ref},k-1} + \kappa \cdot h^c_{E[\mathbf{x}_k]}, \tag{14}$$

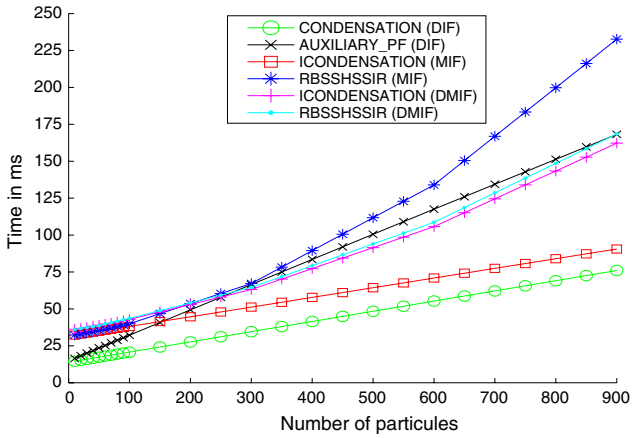


Fig. 20 Average time consumption versus number of particles on all sequences for several strategies

where κ weights the contribution of the mean state histogram $h^c_{E[\mathbf{x}_k]}$ to the target model $h^c_{\text{ref},k-1}$ and index p has been omitted for compactness reasons. Drift and possible subsequent target loss are experienced in any tracker which involves models updating. To avoid this, the particles weighting step considers the likelihood $S2C1$ which fuses, thanks to (9), color distributions cue but also shape cue relatively to the head silhouette (Fig. 4).

Given the H/R interaction distance and the evaluations results in Sect. 4.2, the common importance function of ICONDENSATION and RBSSHSSIR strategies is based on color blobs and face detectors, namely $FSBD$. These proposals permit automatic initialization when persons appear or re-appear in the scene and improve the recovery of deadlocks induced by target loss.

In nominal conditions, all the particle filtering strategies lead to a similar precision and failure rate. Experiments on sequences including appearance or illumination changes, such as the two runs reported in Fig. 21, also show similar results. Indeed, fusing shape and color cues in the

Table 5 Parameter values used in our face tracker

Symbol	Meaning	Value
$(\sigma_u, \sigma_v, \sigma_s, \sigma_\theta)$	Standard deviation of the random walk dynamics noise on the state vector $\mathbf{x_k} = (u_k, v_k, s_k, \theta_k)'$	(15, 6, 0.01, 0.3)
σ_s	Standard deviation in likelihood $S1M$ combining shape and motion cues	36
ρ	Penalty in Eq. (10)	0.12
N	Number of particles	150

Table 6 Parameter values used in our upper human body tracker

Symbol	Meaning	Value
(α, β)	Mixture coefficients in the importance function $q(\mathbf{x}_k \vert \mathbf{x}_{k-1}, z_k)$ along Eq. (2)	(0.3, 0.6)
$(\sigma_u, \sigma_v, \sigma_s)$	Standard deviation of the random walk dynamics noise on the state vector $\mathbf{x_k} = (u_k, v_k, s_k)'$	$(11, 6, \sqrt{0.1})$
$(\sigma_{u_i}, \sigma_{v_i})$	Standard deviation in importance function $\pi(\mathbf{x}_k \vert z^S)$ for FD-based detector	(6, 6)
$(\sigma_{u_i}, \sigma_{v_i})$	Standard deviation in importance function $\pi(\mathbf{x}_k \vert z^C)$ for SBD detector	(6, 6)
σ_s	Standard deviation in shape-based likelihood $p(z_k^S \vert \mathbf{x}_k)$	25
N_R	Number of patches in $p(z_k^C \vert \mathbf{x}_k)$	2
σ_c	Standard deviation in color-based likelihood $p(z_k^C \vert \mathbf{x}_k)$	0.03
N_{bi}	Number of color bins per channel involved in $p(z_k^C \vert \mathbf{x}_k)$	32
κ	Coefficients for reference histograms $h^c_{\text{ref},1}$, $h^c_{\text{ref},2}$ update in Eq. (14)	(0.1, 0.05)
N	Number of particles	150

Table 7 Parameter values used in our whole human body tracker

Symbol	Meaning	Value
(α, β)	Mixture coefficients in the importance function $q(\mathbf{x}_k \vert \mathbf{x}_{k-1}, z_k)$ along Eq. (2)	(0.3, 0.6)
$(\sigma_u, \sigma_v, \sigma_s)$	Standard deviation of the random walk dynamics noise on the state vector $\mathbf{x_k} = (u_k, v_k, s_k)'$	$(7, 5, \sqrt{0.1})$
ν	Threshold for importance function $\pi(\mathbf{x}_k \vert z_k^M)$	10
$(\sigma_{u_i}, \sigma_{v_i})$	Standard deviation in importance function $\pi(\mathbf{x}_k \vert z^M)$ for MD-based detector	(8, 8)
σ_m	Standard deviation in motion-based likelihood $p(z_k^M \vert \mathbf{x}_k)$	0.2
N'_{bi}	Number of motion bins involved in $p(z_k^M \vert \mathbf{x}_k)$	32
σ_c	Standard deviation in color-based likelihood $p(z_k^C \vert \mathbf{x}_k)$	0.03
N_{bi}	Number of color bins per channel involved in $p(z_k^C \vert \mathbf{x}_k)$	32
N_R	Number of patches in $p(z_k^C \vert \mathbf{x}_k)$	1
κ	Coefficient for reference histogram h^c_{ref} update in Eq. (14)	0.1
N	Number of particles	150

measurement function improves the discriminative power so that in these contexts, a robust tracking can be performed whatever the used particle filtering strategy.

Experiments on sequences including additional sporadic disappearances (due to the limits of the camera field of view) or jumps in the target dynamics highlight the efficiency of ICONDENSATION/RBSSHSSIR strategies in terms of failure rate (Fig. 22). In fact, these two strategies, by drawing some particles according to the output from detection modules, permit automatic initialization and aid recovery from transient tracking failures. In addition, the RBSSHSSIR filter leads to a slightly better precision than ICONDENSATION. This is a consequence of the more efficient association of subparticles sampled from the proposal with plausible predecessors thanks to the intermediate resampling stage (step 6 in Table 3).

Experiments on sequences including spurious detections due to the presence of another non-occluding person in the

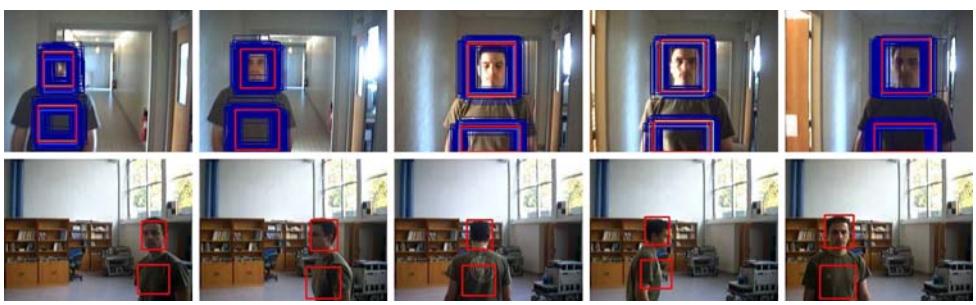


Fig. 21 Tracking scenario involving illumination (*top*) or appearance (*bottom*) changes with DMIF-ICONDENSATION. The *blue* (resp. *red*) *rectangles* depict the particles (resp. the MMSE estimate)

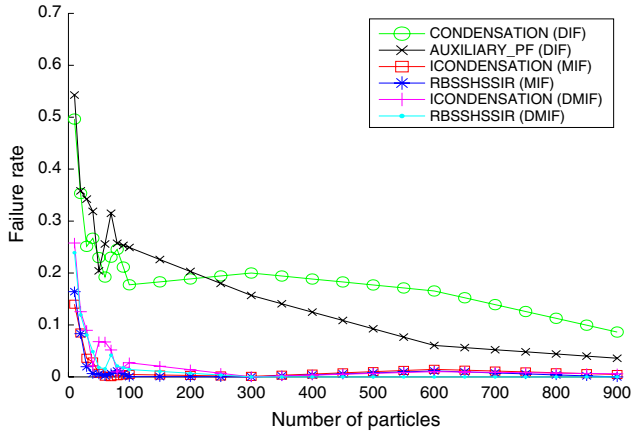


Fig. 22 Average failure rate versus number of particles on sequences including jumps in the target dynamics

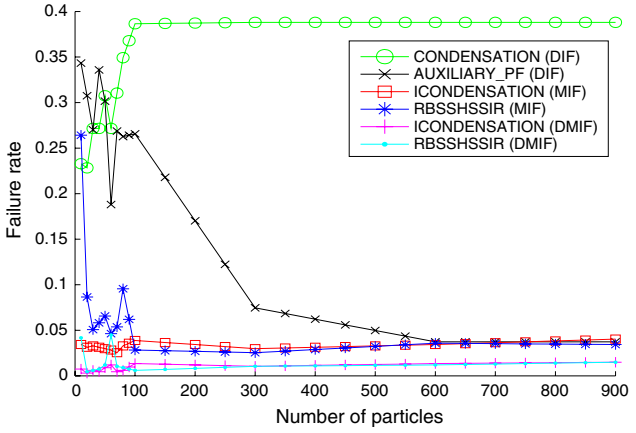


Fig. 25 Average failure rate versus number of particles on sequences including two people occluding each other

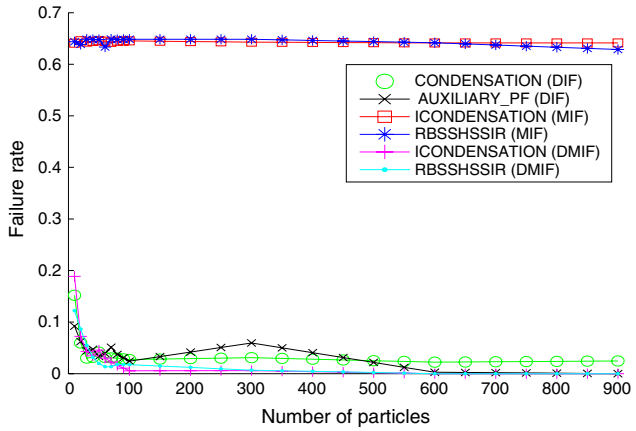


Fig. 23 Average failure rate versus number of particles on sequences including a spurious detection without occlusion

camera field of view, bring out that Measurement-based Importance Functions lead to a worse failure rate (Fig. 23). Conversely, the DMIF strategies ensure a proper tracking thanks to the sampling of some particles from the dynamics.

To illustrate these observations, Fig. 24 shows a tracking run including two persons for MIF-ICONDENSATION (Fig. 24, top) and DMIF-ICONDENSATION (Fig. 24, bottom). In the MIF-ICONDENSATION case, only the non-targeted person is detected so that the importance function draws all the particles on wrong regions, leading to a failure on and after the third frame.

Experiments on sequences which involve two people occluding each other highlight the efficiency of the ICONDENSATION/RBSSHSSIR strategies in terms of failure rate (Fig. 25).

DIF strategies lead to track the person on the foreground (Fig. 26, top), whereas ICONDENSATION/RBSSHSSIR strategies keep locking on the right target throughout the sequence (Fig. 26, bottom) thanks to the sampling of some particles according to the visual detectors outputs, and to the discriminative power of the measurement function.

The above experiments emphasize the necessity of taking into account both the dynamics and the measurements so that the tracking can be robust and efficient enough in all considered scenarios related to the guidance modality. Therefore, the DIF and MIF particle filtering strategies are excluded in this context. Even if DMIF-ICONDENSATION and DMIF-RBSSHSSIR are well suited and have similar time consumption (Fig. 27) for the used number N of particles (between 100 and 200), we finally adopt DMIF-RBSSHSSIR for this guidance modality because of its slightly better performances compared to DMIF-ICONDENSATION. The parameters reported in Table 6 are used in the likelihoods, proposal and state dynamics involved in our upper human body tracker.

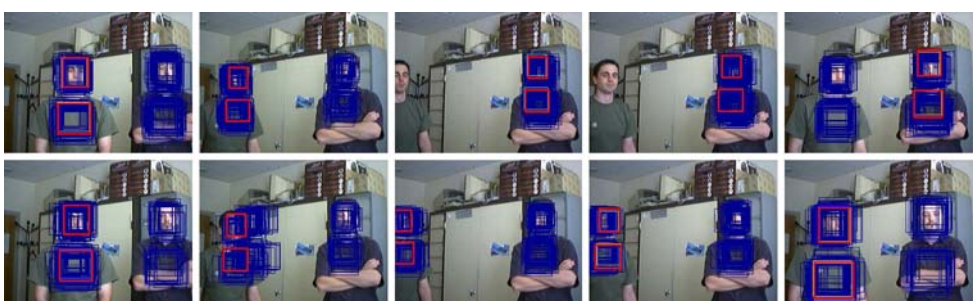


Fig. 24 Tracking scenario involving two people with MIF-ICONDENSATION (*top*) and DMIF-ICONDENSATION (*bottom*). On the *third top frame* the targeted person is not detected while the undesired person remains detected

Fig. 26 Tracking scenario involving occlusions with CONDENSATION (*top*) and DMIF-RBSSHSSIR (*bottom*)

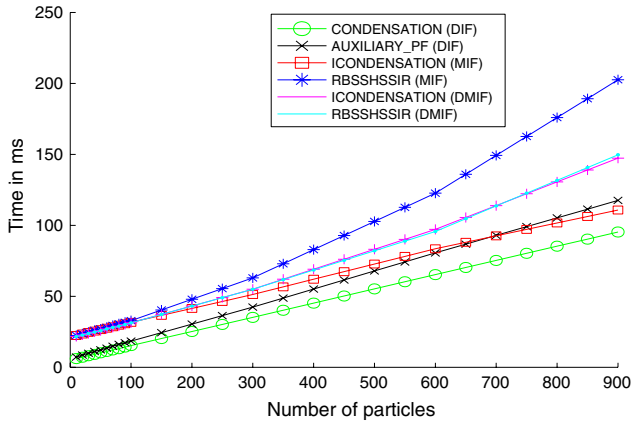


Fig. 27 Average time consumption versus number of particles on all sequences for several strategies

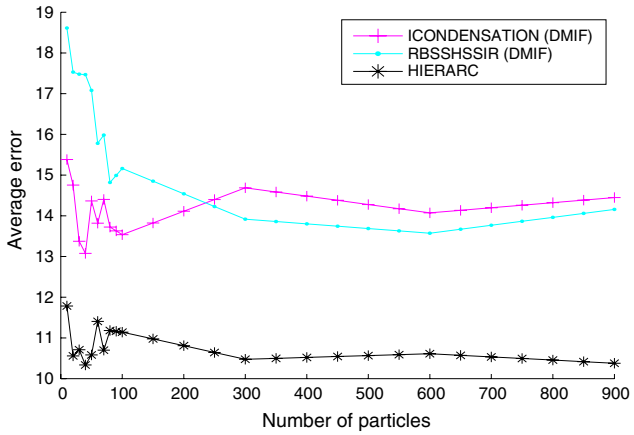


Fig. 28 Tracking errors versus number of particles in nominal conditions

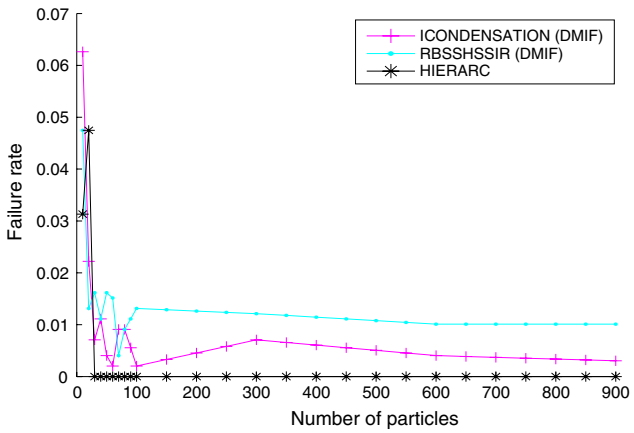


Fig. 29 Average failure rate versus number of particles in nominal conditions

5.2.3 Person tracker (search for interaction)

As was the case in the above section, the state vector has the form $\mathbf{x}_k = (u_k, v_k, s_k)'$. Color and motion cues are fused into the global measurement function (9) as the robot is supposed motionless. The selected importance function MD is defined from the motion detector output. Relying on the conclusions concerning the guidance modality, only DMIF-ICONDENSATION and DMIF-RBSSHSSIR are evaluated. As the Hierarchical Particle Filter (HIERARCHICAL_PF) defined in [33] constitutes an alternative to these strategies, it is also assessed. Thanks to its intermediate sampling (step 6 in Table 4), which enables the particles cloud to remain more focused on the target, it results in a significant decrease of the tracking error under nominal conditions, as illustrated in Fig. 28. In this helpful case, a slightly better failure rate is also observed as shown in Fig. 29.

Experiments on sequences including a full occlusion of the moving target by a static object (Fig. 32) highlight the efficiency of DMIF-ICONDENSATION/DMIF-RBSSHSSIR strategies in terms of of failure rate compared to the HIERARCHICAL_PF strategy (Fig. 30). Though some particles are sampled on the targeted person by the motion-based importance function (MD) as soon as he/she reappears after an occlusion, the HIERARCHICAL_PF strategy fails (Figs. 31, 32). Indeed, as these particles lie in the tails of the dynamics pdf, they are affected small weights and thus get eliminated during the first resampling step of the algorithm (step 6 in Table 4). Meanwhile, the other filtering strategies which rely on Dynamics and Measurement based importance functions can perform the tracking.

Similar conclusions hold when the target is motionless and subject to occlusion (Figs. 33, 34). This can again be explained by the action of its first resampling step which concentrates particles on high motion activity regions. Figures 35

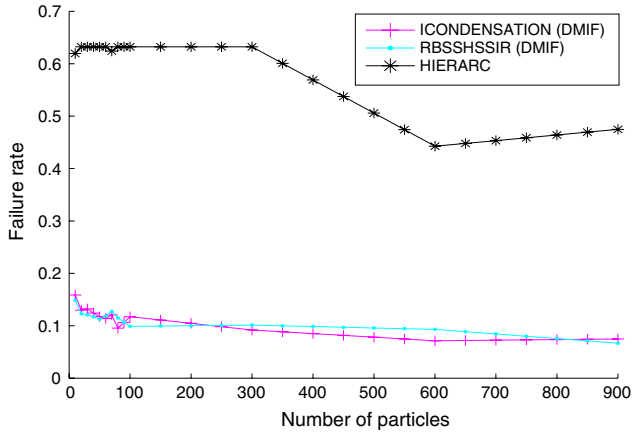


Fig. 30 Average failure rate versus number of particles on sequences including an occlusion by a static object

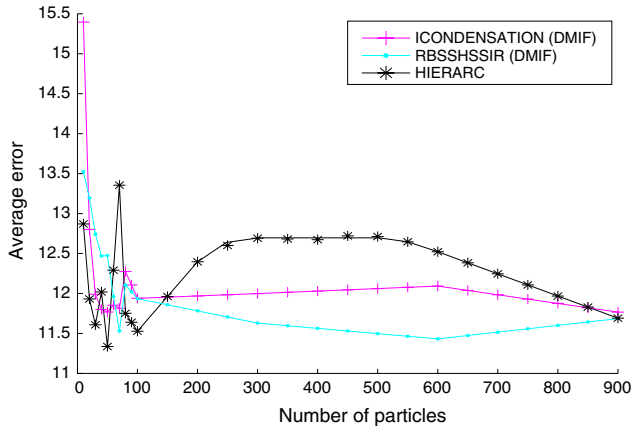


Fig. 31 Tracking errors versus number of particles on sequences including an occlusion by a static object

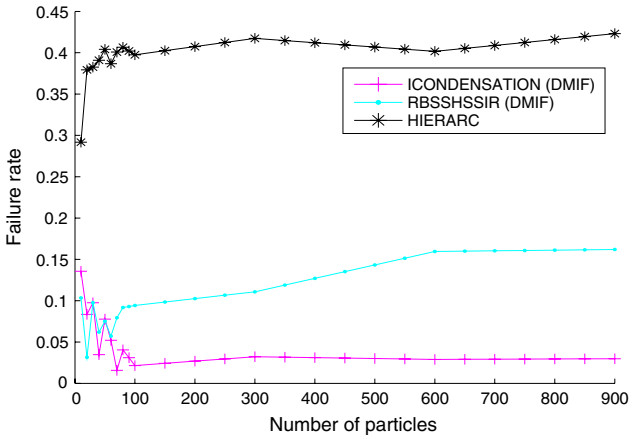


Fig. 33 Average failure rate versus number of particles on sequences including target occlusions

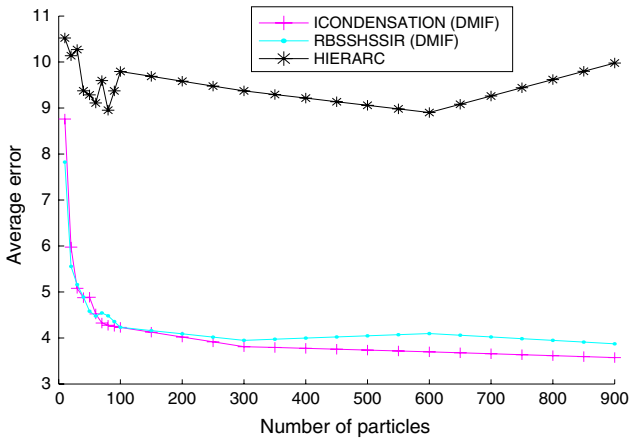


Fig. 34 Tracking errors versus number of particles on sequences including target occlusions

and 36 present two tracking scenarios involving several persons with mutual occlusions. In the first scenario, the HIERARCHICAL_PF tracker locks onto the undesired person, because only regions of high motion activity which are in the modes of the system dynamics pdf are explored by its first step. Regions corresponding to the target, even if they do comply with the color model, are thus discarded during the resampling procedure. In contrast, the RBSSHSSIR tracker which doesn't dissociate motion and color cues, keeps locking on the targeted person. The second scenario leads to the same observations and confirms the RBSSHSSIR efficiency. The two filters DMIF-ICONDENSATION/DMIF-RBSSHSSIR are well-suited to this modality. As robustness is preferred to precision for our application, we finally opt for the DMIF-RBSSHSSIR algorithm. The fixed parameters involved in the likelihoods, proposal and state dynamics of our human body tracker are given in Table 7.

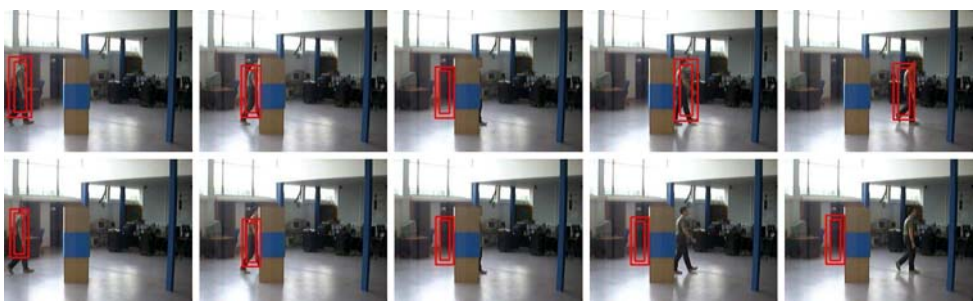


Fig. 32 Tracking scenario involving full occlusion by a static object with DMIF-RBSSHSSIR (*top*) and HIERARCHICAL_PF (*bottom*)

Fig. 35 Tracking scenario involving occlusion of the motionless target by another person crossing the field of view with DMIF-RBSSHSSIR (*top*) and HIERARCHICAL_PF (*bottom*)

Fig. 36 A scenario involving persistent occlusions due to persons. Tracker based on a DMIF into the RBSSHSSIR algorithm

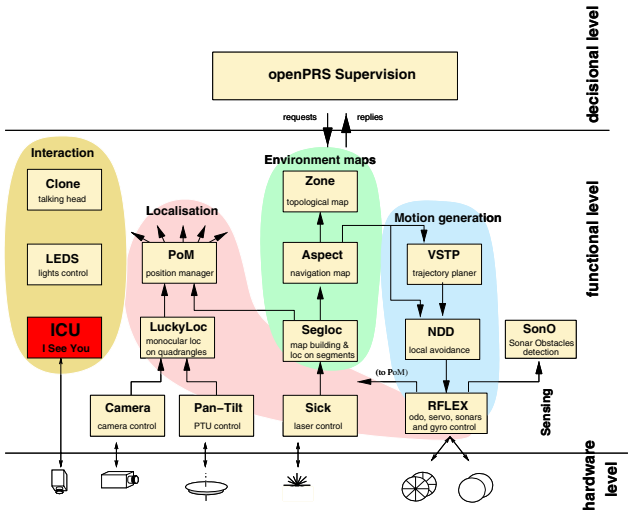


Fig. 37 Rackham's software architecture

6 Integration on Rackham robot

6.1 Outline of the overall software architecture

The above visual functions were embedded on the Rackham robot. To this aim, Rackham is fitted with the "LAAS" software architecture introduced in Fig. 37 and thoroughly presented in [1].

On the top of the hardware (sensors and effectors), the *functional level* encapsulates all the robot's action and perception capabilities into controllable communicating modules, operating at very strong temporal constraints. The *executive level* activates these modules, controls the embedded functions, and coordinates the services depending on the task high-level requirements. Finally, the upper *decision level* copes with task planning and supervision, while remaining reactive to events from the execution control level.

In addition to functional modules dedicated to exteroceptive sensors handling, e.g. cameras, laser and ultrasonic telemeters,…, low-level servo algorithms, elementary navigation functions, etc.[5], a module named ICU—for "I see you"—has been designed which encapsulates all the aforementioned tracking modalities. It is depicted below.

6.2 Considerations about the ICU software architecture

The C++ implementation of the module ICU is integrated in the "LAAS" architecture using a C/C++ interfacing scheme. It enjoys a high modularity thanks to C++ abstract classes and template implementations. This way, virtually any tracker can be implemented by selecting its components from predefined libraries related to particle filtering strategies, state evolution models, and measurement/importance functions. For more flexibility, specific components can be defined and integrated directly.

ICU sequentially invokes the tracking components through its processing pipe, as illustrated in Figure 38. So, the functions shared by several trackers running in parallel are processed only once.

Section 6.3 enumerates all the visual functions provided by the module ICU, which not limited to tracking. Section 6.4 details the way how they are entailed in the tour-guide scenario, and discusses the automatic switching between trackers.

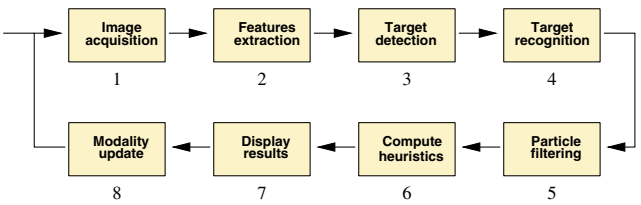


Fig. 38 Sequencing the module ICU

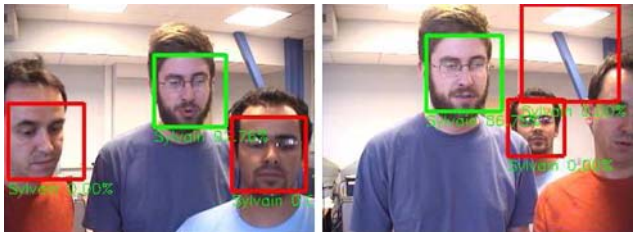


Fig. 39 Snapshots of detected (*red*)/recognized (*green*) faces with associated probabilities. The target is named Sylvain in this example

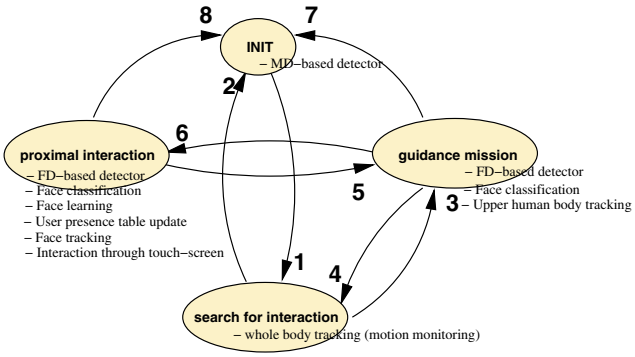


Fig. 40 Transitions between tracking modalities

6.3 Visual functions provided by the module ICU

These can be organized into three broad categories:

1. Functions related to human body/limbs detection: Independently from the tracking loop, FD-based or MD-based detectors (see Sect. 4.2) can be invoked depending on the current H/R distance and the scenario status.
2. Functions related to user face recognition: The face recognition process underlies the following functions
   - a *face learning function* based on the FD-based detector in order to train the classifier.
   - a *face classification function* based on these training samples and eigenfaces representation [38]. The face recognition probability associated with each detected face can be integrated both in the face and upper human body trackers. Some recognition snapshots are reported in Fig. 39. Further details can be found in [16].
   - a *user presence function* updates a presence table of the 30 previously learned robot users. The table update is similar to a FIFO stack, i.e. the oldest user added in the table is handled next.
3. Functions related to user tracking: These are
   - the *three tracking functions* characterized and evaluated in Sect. 5. Recall that they have been designed so as to best suit to the interaction modalities of Sect. 2.
   - an *estimator of the H/R distance* of the targeted person from the scale of the updated template during the tracking.

   The robot activates these functions depending on the current H/R distance, user identification and scenario status. The next subsection details the way how they are scheduled.

6.4 Heuristic-based switching between trackers

A finite-state automaton can be defined from the tour-guide scenario outlined in Sect. 2, as illustrated in Fig. 40. Its four states are respectively associated to the INIT mode and to the three aforementioned interaction modalities. Two heuristics relying on the current H/R distance and the face recognition status allow to characterize most of the transitions in the graph. Practically, outcomes from the face classifier and H/R distance functions are filtered on a sliding temporal window W_t of about 20 samples. The robot in INIT mode invokes the motion-based detector (MD), so that any visitor entering the exhibition initializes the whole body tracking (arrow 1). The robot assumes that the visitor is willing to interact when he/she has come closer and has got detected by the frontal face FD-detector over W_t. If so, the upper human body tracking mode is launched (arrow 3). If the user H/R distance keeps decreasing to less than $1m$ and his/her face remains detected/recognized, a "proximal interaction" begins, entailing the face tracker (arrow 6). The face learning function and the human presence table update function are possibly invoked if the user is unknown. When starting the "guidance mission", the robot switches to the upper human body tracker (arrow 5). Temporary target loss are notified when the face classifier fails for more than 70% of the 20 images composing W_t. Re-identification of the guided visitor in the next W_t is required in order to resume the ongoing mission. Finally, the robot returns in INIT mode when: (1) no moving blobs are detected, (2) the current user hasn't been recognized over W_t, (3) the end mission is signified by the robot (arrows 2, 7 and 8).

Thanks to its efficient modular implementation, all the ICU functions can be executed in real time on our robot. Experiments show their complementary and efficiency in cluttered scenes.

7 Conclusion

This paper has introduced mechanisms for visual data fusion/combination within particle filtering to develop people trackers from a single color camera mounted on a mobile robot. Most particle filtering techniques to single-target tracking

have been surveyed and tested. A first contribution concerns visual data fusion for the considered robotics scenarios, a context which has fairly seldom exploited particle filtering based solutions. The most persistent cues are used in the particles weighting stage. The others, logically intermittent, permit automatic initialization and aid recovery from transient tracking failures. Mixing these cues both into the importance and measurement functions of the underlying estimation scheme, can help trackers work under a wide range of conditions encountered by our Rackham robot during its displacements. A second contribution relates to the evaluation of dedicated particle filtering strategies in order to check which people trackers, regarding visual cues and algorithms associations, best fulfill the requirements of the considered scenarios. Let us point out that few studies comparing the efficiency of so many filtering strategies had been carried out in the literature before. A third contribution concerns the integration of all these trackers on a mobile platform, whose deployment in public areas has highlighted the relevance and the complementarity of our visual modalities. To our knowledge, quite few mature robotic systems enjoy such advanced capabilities of human perception.

Several directions are currently investigated. First, we study how to fuse other information such as laser or sound cues. The sound cue would not just contribute to the localization in the image plane, but will also endow the tracker with the ability to switch its focus between speakers. A next issue will concern the incorporation of appropriate degrees of adaptivity into our multiple cues based likelihood models depending on the target properties changes or the current viewing conditions [40]. In addition, our tracking modalities will be made much more active. Zooming will be used to actively adapt the focal length with respect to the H/R distance and to the current active visual modalities. Finally, the tracker will be enhanced so as to track multiple persons simultaneously [27,42].

Acknowledgments The work described in this paper was partially conducted within the EU Integrated Project COGNIRON (The Cognitive Companion) and the EU STREP Project CommRob (Advanced Behavior and High-Level Multimodal Communication with and among Robots), funded by the European Commission Division FP6-IST Future and Emerging Technologies under Contracts FP6-IST-002020 and FP6-IST-045441.

References

1. Alami, R., Chatila, R., Fleury, S., Ingrand, F.: An architecture for autonomy. Int. J. Robot. Res. **17**(4), 315–337 (1998)
2. Andrieu, C., Davy, M., Doucet, A.: Improved auxiliary particle filtering: Application to timevarying spectral analysis. In: IEEE Wk. on Statistical Signal Processing, pp. 309–312. Singapore (2001)
3. Arulampalam, S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. IEEE Trans. Signal Process. **50**(2), 174–188 (2002)
4. Avidan, S.: Support vector tracking. In: IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR'01), pp. 184–191. Kauai, HI (2001)
5. Bailly, G., Brèthes, L., Chatila, R., Clodic, A., Crowley, J., Danès, P., Elisei, F., Fleury, S., Herrb, M., Lerasle, F., Menezes, P., Alami, R.: HR+ : towards an interactive autonomous robot. In: Journées ROBEA, pp. 39–45. Montpellier, France (2005)
6. Bichot, E., Mascarilla, L., Courtellemont, P.: Particle filtering based on motion and color information. IEEE Trans. Information Sci. Appl. **2**, 220–227 (2005)
7. Brèthes, L., Menezes, P., Lerasle, F., Briot, M.: Face tracking and hand gesture recognition for human-robot interaction. In: IEEE Int. Conf. on Robotics and Automation (ICRA'04), pp. 1901–1906. New Orleans, LA (2004)
8. Bretzner, L., Laptev, I., Lindeberg, T.: Hand gesture using multi-scale colour features, hierarchical models and particle filtering. In: IEEE Int. Conf. on Automatic Face and Gesture Recognition (FGR'02), pp. 405–410. Washington D.C. (2002)
9. Brèthes, L.: Suivi visuel par filtrage particulaire. Application à l'interaction homme-robot. Ph.D. thesis, Université Paul Sabatier, LAAS-CNRS, Toulouse (2006)
10. Bullock, D., Zelek, J.: Real-time tracking for visual interface applications in cluttered and occluding situations. J. Vis. Image Comput. **22**, 1083–1091 (2004)
11. Chen, H., Liu, T.: Trust-region methods for real-time tracking. In: IEEE Int. Conf. on Computer Vision (ICCV'01), vol. 2, pp. 717–722. Vancouver, Canada (2001)
12. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. In: IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, pp. 564–575 (2003)
13. Doucet, A., De Freitas, N., Gordon, N.J.: Sequential Monte Carlo Methods in Practice. Series Statistics For Engineering and Information Science. Springer, New York (2001)
14. Doucet, A., Godsill, S.J., Andrieu, C.: On sequential monte carlo sampling methods for bayesian filtering. Stat. Comput. **10**(3), 197–208 (2000)
15. Gavrila, D.M.: The visual analysis of human movement : a survey. Comput. Vis. Image Underst. **1**, 82–98 (1999)
16. Germa, T., Brèthes, L., Lerasle, F., Simon, T.: Data fusion and eigenface based tracking dedicated to a tour-guide robot. In: Int. Conf. on Vision Systems (ICVS'07). Bielefeld, Germany (2007)
17. Giebel, J., Gavrila, D.M., Schnorr, C.: A bayesian framework for multi-cue 3D object. In: Eur. Conf. on Computer Vision (ECCV'04). Prague, Czech Republic (2004)
18. Haritaoglu, I., Harwood, D., Davis, L.: W4 : Real-time surveillance of people and their activities. IEEE Trans. Pattern Anal. Mach. Intell. **8**(22), 809–830 (2000)
19. Hu, W., Tan, T., Wang, L., Maybank, S.: A survey on visual surveillance of object motion and behaviors. IEEE Trans. Syst. Man Cybernet. **34**(3), 334–352 (2004)
20. Isard, M., Blake, A.: Condensation—conditional density propagation for visual tracking. Int. J. Comput. Vis. **29**(1), 5–28 (1998)
21. Isard, M., Blake, A.: Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In: Eur. Conf. On Computer Vision (ECCV'98), pp. 893–908. London, UK (1998)
22. Isard, M.A., Blake, A.: A mixed-state condensation tracker with automatic model-switching. In: IEEE Int. Conf. on Computer Vision (ICCV'98), pp. 107–112. Bombay, India (1998)
23. Jones, M., Rehg, J.: Color detection. Tech. rep., Compaq Cambridge Research Lab (1998)
24. Julier, S., Uhlmann, J.: A general method for approximating nonlinear transformations of probability distributions. Tech. rep., RRG, Dept. of Engineering Science, University of Oxford (1994)
25. Kitagawa, G.: Monte carlo filter and smoother for non-gaussian nonlinear state space models. J. Comput. Graph. Stat. **5**(1), 1–25 (1996)

26. Li, P., Zhang, T.: Visual contour based on sequential importance sampling/resampling algorithm. In: IEEE Int. Conf. on Pattern Recognition (ICPR'02), pp. 564–568. Quebec, Canada (2002)
27. MacCormick, J., Blake, A.: A probabilistic exclusion principle for tracking multiple objects. Int. J. Comput. Vis. **39**(1), 57–71 (2000)
28. MacCormick, J., Isard, M.: Partitioned sampling, articulated objects, and interface-quality hand tracking. In: Eur. Conf. on Computer Vision (ECCV'00), pp. 3–19. Springer, London, UK (2000)
29. Moeslund, T., Granum, E.: A survey on computer vision-based human motion capture. Comput. Vis. Image Underst. **81**, 231–268 (2001)
30. Nummiaro, K., Koller-Meier, E., Gool, L.V.: An adaptative color-based particle filter. J. Image Vis. Comput. **21**, 90–110 (2003)
31. Pitt, M., Shephard, N.: Filtering via simulation: auxiliary particle filters. J. Am. Stat. Assoc. **94**(446), 590–599 (1999)
32. Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-based probabilistic tracking. In: Eur. Conf. on Computer Vision (ECCV'02), pp. 661–675. Berlin, Germany (2002)
33. Pérez, P., Vermaak, J., Blake, A.: Data fusion for visual tracking with particles. Proc. IEEE **92**(3), 495–513 (2004)
34. Rui, Y., Chen, Y.: Better proposal distributions: Object tracking using unscented particle filter. In: IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR'01), pp. 786–793. Kauai, HI (2001)
35. Schwerdt, K., Crowley, J.L.: Robust face tracking using color. In: Int. Conf. on Face and Gesture Recognition (FGR'00), pp. 90–95. Grenoble, France (2000)
36. Thayananthan, A., Stenger, B., Torr, P., Cipolla, R.: Learning a kinematic prior for tree-based filtering. In: British Machine Vision Conf. (BMVC'03), vol. 2, pp. 589–598. Norwich, UK (2003)
37. Torma, P., Szepesvári, C.: Sequential importance sampling for visual tracking reconsidered. In: AI and Statistics, pp. 198–205 (2003)
38. Turk, M.A., Pentland, A.P.: Face recognition using eigenfaces. In: IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR'91), pp. 586–591. Maui, HI (1991)
39. Van Der Merwe, R., De Freitas, N., Doucet, A., Wan, E.: The unscented particle filter. In: Advances in Neural Information Processing Systems, vol. 13 (2001)
40. Vermaak, J., Andrieu, C., Doucet, A., Godsill, S.: Particle methods for bayesian modeling and enhancement of speech signals. IEEE Trans. Speech Audio Process. **10**(3), 173–185 (2002)
41. Vermaak, J., Blake, A.: A nonlinear filtering for speaker tracking in noisy and reverberant environments. In: IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'00). Istanbul, Turkey (2000)
42. Vermaak, J., Doucet, A., Pérez, P.: Maintaining multi-modality through mixture tracking. In: IEEE Int. Conf. on Computer Vision (ICCV'03). Nice, France (2003)
43. Vermaak, J., Pérez, P., Gangnet, M., Blake, A.: Towards improved observation models for visual tracking: Selective adaptation. In: Eur. Conf. on Computer Vision (ECCV'02), pp. 645–660. Berlin, Germany (2002)
44. Vezhnevets, V., Sazonov, V., Andreeva, A.: A survey on pixel-based skin color detection techniques. In: Proc. Graphicon-2003 pp. 85–92 (2003)
45. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR'01). Kauai, HI (2001)
46. Wachter, S., Nagel, H.: Tracking persons in monocular image sequences. Comput. Vis. Image Underst. **74**(3), 174–192 (1999)
47. Wu, Y., Huang, T.: A co-inference approach to robust visual tracking. In: IEEE Int. Conf. on Computer Vision (ICCV'01), vol. 2, pp. 26–33. Vancouver, Canada (2001)